TELKOMNIKA, Vol.17, No.2, April 2019, pp.676~682
ISSN: 1693-6930, accredited First Grade by Kemenristekdikti, Decree No: 21/E/KPT/2018
DOI: 10.12928/TELKOMNIKA.v17i2. 9621

A decentralized paradigm for resource-aware computing in wireless Ad hoc networks

Heerok Banerjee*[1], S Murugaanandam[2], V Ganapathy[3]
Department of Information Technology, School of Computing, SRM Institute of Science and Technology,
Mahatma Gandhi Rd, Potheri, SRM Nagar, Kattankulathur, Tamil Nadu 603203, India
*Corresponding author, e-mail: heerokbanerjee_ra@srmuniv.edu.in[1],
murugaanandam.s@ktr.srmuniv.ac.in[2], ganapathy.v@ktr.srmuniv.ac.in[3]

Abstract

A key factor limiting the democratisation of networked systems is the lack of trust, particularly in the wake of data-intensive applications that work on sensitive and private data, which requires providing strong network security guarantees via encryption and authentication algorithms, as well as rethinking algorithms to compute on the network peripheries without moving data. In many security and privacy-critical domains such as Home Automation IoT networks, AUV networks etc., the existence of a centralized privileged node leads to a vulnerability for leakage of sensitive information. In this paper, we have proposed a decentralized networking architecture that adopts collaborative processing techniques and operates within the tradeoff between network security and performance. We have investigated the design and sustainability of autonomous decentralized systems and evaluated the efficiency of the proposed scheme with the help of extensive simulation tools.

Keywords: *Ad hoc networks, decentralized computing, grid computing, peer-to-peer networking, wireless networks*

1. Introduction

The emergence of IoT and grid computing is a harbinger for the reinvestigation of network reliability measures and security policies in order to save such technologies from scrutiny and misuse. Moreover, the advent of such technologies has led manufacturers to produce cheap network-capable devices in large quantities. Refrigerators, DVRs, thermostats, and even minute utilities such as light-bulbs have been redesigned to be networking capable and to provide an extra convenience through remote controlling services. As observed, many of these devices have weak security and are considerably configured with easily penetrable security measures [1, 2]. To overcome these issues related to network reliability and security, we concentrate on decentralized network architectures that balances the trade-off between network security and network performance [2]. One of the key goals achieved by the decentralization of networked systems is increasing the variability in the distribution of failure nodes inside an autonomous system, which contributes in building a resilient network to sustain from critical network failures.

Over the years, the World Wide Web has rapidly transformed from an obscure platform for publishing content to a primary infrastructure of learning and commerce. But, the majority of its users are unaware that the internet has suffered substantial changes in its organization and infrastructure over the time of its expansion. The early web consisted of chaotic and complicated navigations, where organization and management of data were heavily distributed. The early rise of the internet was supported by distributed protocols which failed to provide consistency and an effective delivery. So, it was evident that the web needed to be consolidated into an uncompromised and curated service platform to reach the requirements of businesses and the general audience. This practice towards the consolidation of the internet architecture led to the introduction of two key activities, namely, routing mechanisms and self-organizing topologies. While several novel routing mechanisms and self-organizing architectures have been designed and implemented over the years, they have failed critically in terms of providing security and reliability [3]. Additionally, the dilemma of prioritizing performance over security has always been imperative [4] and is one of the key factors in determining the limitations of a

Received April 17, 2018; Revised December 18, 2018; Accepted January 26, 2019

networking architecture. The prioritization of performance and QoS over security certainly leads to poor integrity of data and network security. This trade-off between network security and performance has yielded many security-centric and performance-centric architectures that are operated in our day to day life. However, such architectures have not remained consistent throughout the period of their development. Several ramifications in the past have led to protocols pertaining to high performance and low reliability [5], whereas modern approaches consider security and user-privacy on top. In this paper, we propose the design of decentralized networks and discuss the performance of security-centric protocols which are equipped to deal with the growing necessity of digital data as represented in Table 1.

Centralized networks are a group of interconnected devices that are governed and monitored at a single node which acts as a communicating agent with neighboring networks. Essentially, the entire networking process which comprises of validation, encryption/decryption, computation, routing, interpretation and re-organization, is also centralized and is much expected to be a primary target for disrupting the entire network. On the other hand, decentralized networks do not propagate information through a single point. Devices are selected stochastically and are interconnected with each other forming many peer-to-peer networks that provide secure and reliable communication and hence serve concrete privacy and anonymity over the network [6]. In this paper, we have proposed a novel architecture accounting both, the performance and the security measures involved during the design of distributed ad hoc networks for accommodating multi-objective networking applications and also provided a blueprint to implement decentralized systems using contemporary simulation frameworks [7, 8].

2. Research Methodology

We conducted several quasi-experimental evaluations of widely accepted architectures [9], with promising historical evidences. Our evaluations inclined mostly towards the underlying security measures and methodological foundations of network analytics, based on which most of the design was framed.

Firstly, we attempted to identify the architectural constraints existent in traditional networked architecture and compared them extensively with major details [9, 10]. Consequently, we simulated the protocols designed for wireless ad hoc networks with typically large network participants and introduced various dynamic changes (load variations, topological updates, inbound congestions) to conduct quasi-experimental observations. We recorded some of the network performance metrics such as response time, module-to-module delay, packet drop ratio etc. Additionally, we conducted several benchmark tests and observed anomalies in these conventional models. Table 1 reproduces the characteristics of different network architectures and illustrates that the proposed community-based architecture promises reliable, secure and uniform connectivity amongst its community nodes.

Table 1. Comparison of Different Network Architectures

Model Property		Client-Server (Conventional)	Peer-Peer	Decentralized Community based
Control		Centralized	Centralized	Decentralized
Connection	Model	Address-based	Address-based	Service-based
	Request	One-one	One-Many	Cooperative (1→N)
	Reply	One-one	One-one	Collaborative (M→N)
	Feature	Unilateral	Unilateral	Multilateral
Nodes		Passive	Active	Active
Load		Server Congestion	Peer Congestion	NIL

3. Proposed Architecture

The proposed architecture namely Autonomous Collaborative Decentralized-Community Network (ACDCN) is a self-organizing logical topology for incorporating collaborative networking capabilities by distributing the workload to its neighboring collaborators. The member nodes are selected by considering the bilateral connectivity and availability of resources. A publisher node introduces a complex process, which is further disseminated in the community network by introducing pairs of collaborations and shared data pools [10, 11]. Figure 2 shows that the community nodes acknowledge at least three of their neighboring collaborators. In such decentralized architectures, a community node retains full freedom to connect with the community via any existent member node, and thus, decreases the threat of intrusion.

The objective of this architecture is to distribute data and reduce the computational workload amongst its community nodes. The underlying network consists of non-community member nodes that act as proxy members as shown in Figure 1, to transfer data from one community to another. These nodes are not aware of inter-community transactions and do not participate in sharing the workload of the network. The non-member nodes assure a feasible path to allow a bilateral communication between communities and exchange of data in a stochastic manner. On a large scale, community nodes also publish requests recursively for other community networks, only if the entire community is exhausted. The subscriber nodes receive triggered updates from the publisher and compute on the provided data [11]. Hence, the proposed architecture allows distribution of data-intensive operations and adds a feature to the QoS factor by the collaboration of computational power.

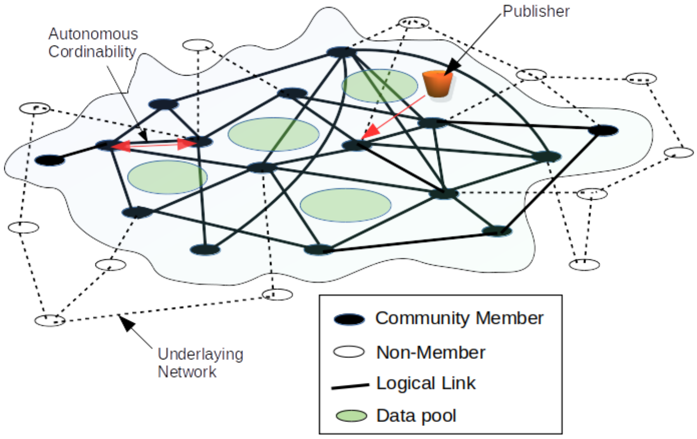


Figure 1. Overview of collaborative decentralized-community network architecture

The key features of the proposed architecture are discussed:

a. Fault-Tolerance and Stability

Conventional network models follow a centralized channel of communication and are easy to maintain. However, since there is only a single point of failure, this leads to an insecure and vulnerable threat to the entire system. Such central nodes are keen to network attacks and the instability and the overhead of the central node impact the entire performance of the network [12]. Centralized systems are highly unstable whereas distributed decentralized systems are very stable where singleton failure is negligible and does not cause much chaos as compared to centralized systems. However, distributed systems have a limitation in terms of reliable communication and security. A hybridized architecture, accounting decentralization of data-intensive distributed systems, yields a decentralized community-based architecture of networked systems, which is fault tolerant and highly stable.

b. Publisher-Subscriber Protocol

A community-based network architecture serves as a publish-subscribe platform where community nodes can publish their share of tasks and broadcast them to the entire community [13]. Community networks are time-stamped and can provide much more security. In the proposed architecture, a publisher-subscribe mechanism is implemented which offers an effective solution to distribute the load uniformly on demand. When a community node has insufficient computing resources to complete a task, the node publishes the request to all the community members [14], thereby relieving itself from huge computations and alleviating some memory to rejoining the community network.

c. Autonomy and Self-Organization

The dynamism exhibited by such large-scale networks requires some degree of autonomy and self-organization [15, 16]. Inspired by the cooperation in economic communities and the Autonomous Decentralized System (ADS) architecture [6], we propose the concept of Autonomous Collaborative Decentralized-Community Networks (ACDCN) which are intended to meet the rapid dynamism of the network requirements [17]. It categorizes and customizes tasks specific to collaborator specifications and is completely decentralized while selecting the collaborator, in the sense, each community node performs the same set of operations as subjected to the network but with random participants. In such decentralized schemes, conventional machine learning techniques are feasible to yield self-learning algorithms and meta-heuristic algorithms for large computations (hierarchical classification, path determination and content management) with respect to time-variant attributes [18].

4. Simulation Modelling and Performance Analysis

The simulation setup of the proposed architecture ACDCN was performed by using OMNet++ [7] and the data is visualized with matplotlib [8]. Initially, for setting an environment of a small-scale topology as shown in Figure 2, we have taken the number of nodes n=7 and the maximum number of collaborators m=5. We then introduce a composite and complex task f(node $_{[0]}$) solvable in polynomial time T(η) to the community network. Node $_{[0]}$ allocates memory to complete a subpart of f(node $_{[0]}$) and publishes the residual work $f^{(7,4)}$(node $_{[0]}$) as a subset of the initial task to the community network with four collaborators namely, node $_{[1]}$, node $_{[2]}$, node $_{[3]}$ and node $_{[4]}$. The collaborating members are notified by sharing a hash table. Each collaborator follows the same procedure, i.e., it randomly selects a subtask and adds itself to the community network by measuring two parameters, namely residual buffer size ε_k and propagation delay K_{const}. Figure 2 illustrates the simulation environment:

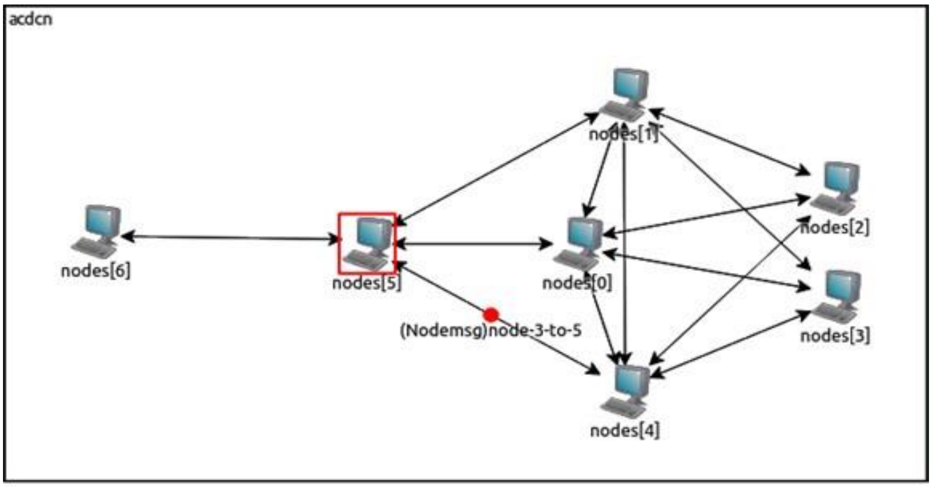


Figure 2. ACDCN simulation-topology and components (n=7, m=5)

mathematically, the completion time for a task can be formulated as follows:

$$T\left(f^{(n,m)}\left(node_{[i]}\right)\right) = \eta_{(i)} + \sigma \times T\left(\sum_{a=i+1}^{m} f^{(n,n-a)}\left(node_{[a]}\right) + \kappa_{const}\right) \quad (1)$$

Here, $\eta_{(i)}$ denotes the individual time taken by node $_{[i]}$ to compute its own task, σ is the collaboration index of the community network and K_{const} is the constant propagation delay, assuming that the channel operates at uniform delays. The optimal time to solve the task f(node $_{[i]}$) is T(η) = $\eta_{(0)} + \eta_{(1)} + \eta_{(2)} + \eta_{(3)} + \ldots + \eta_{(m)}$ **.** If the number of collaborating nodes tends to reach n-1, then we can put limit from {m} to {n-1} to determine the non-parametric equation as given:

$$T(\eta) = \eta_{(0)} + \sigma\left(\sum_{a=1}^{n-1} f^{(n,n-a)}\left(node_{[a]}\right) + (n-1) \times \kappa_{const}\right) \quad (2)$$

$$T(\eta) = \eta_{(0)} + \left(\frac{\sigma}{1-\sigma}\right) \times (n-1) \times \kappa_{const} \quad \Rightarrow 0 < \sigma < 1 \tag{3}$$

We used non-parametric estimation techniques to estimate suitable values for σ in order to achieve an optimal response time [19, 20]. After the simulation was carried out, we extracted the simulation data and using Qtiplot, we attempted to fit the data into the proposed linear mathematical model to determine the accuracy of the proposed model. The simulation was carried out by initiating the community nodes at random locations, each equipped with three types of modules namely, an application module, a routing module and three queuing modules. The application module generates the complex tasks and encapsulates the set of subtasks in a test program. This test program is embedded in a packet and then forwarded to the routing module. The routing module is responsible for determining the optimal route to a potential collaborator and forwarding the packet to the next hop. The data received by the collaborator is again interpreted by the application module and further decomposed to be executed collaboratively.

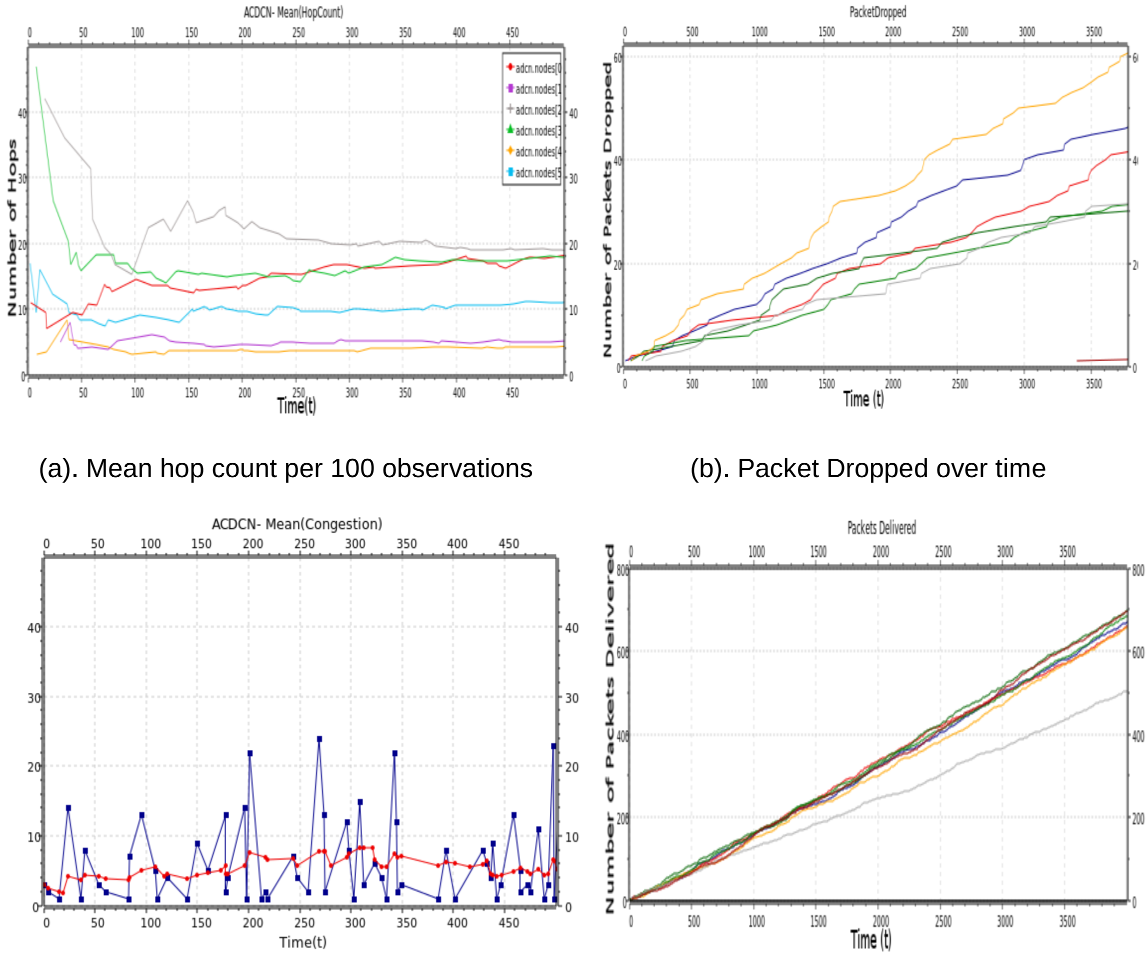
5. Results and Discussion

After performing the simulation of the proposed system, we analyzed the performance of the system based on the mathematical model given in equation (3). We plotted the graphs of conventional QoS metrics used to determine the efficiency of a network architecture. The metrics along with the obtained graphs are given below in relevance to the performance of the proposed system.

The number of hops encountered by the packet essentially refers to the number of recursive calls to the test program. We observed that the initial hops are variant and gradually synchronizes to a consistent value which remains uniform for the rest of the simulation period. Initially, m collaborators of fixed buffer size are unoccupied and can distribute sub-parts of the test program amongst themselves. Hence, the community nodes follow a non-linear curve as shown in Figure 3(a) As the residual memory available to the community members decreases as a function of time, a lesser number of complex queries could be served collaboratively. As a result, the collaborator then operates with an average congestion, which was observed to be 11 kbps per collaborator, as depicted from Figure 3(c). For optimal response time, we specify different collaborative indices σ [equation (3)] for a subset of collaborators. By varying the value of σ, we limit the number of collaborations for a set of given tasks depending upon the workload shared by the collaborator.

The TTL parameter plays an important role in managing congestion and balancing packet drop ratio. Every packet header contains a lifespan of the packet which is referred to as the Time To Lease (TTL). If a certain amount of time expires, then the packet is discarded and the buffer is released. Hence it is significant to re-evaluate routing and resource provisioning algorithms to attain maximum delivery ratio and minimize the number of discarded packets [21]. The simulation results show that the number of packets dropped is multivariate and exhibit linearity in Figure 3(b). The average number of packets lost is calculated to be approximately 1528, which is relatively low as compared to other application layer protocols. Such losses can be avoided by adopting efficient routing mechanisms in the network, which proactively determine the optimal path to the destination with lesser cost [22, 23].

When the test program is completely executed, the results of each successive recursion are sent to the publisher. The subscribers then re-organize the data to generate the response to the publisher as shown in Figure 2. Figure 3(d) plots the number of packets successfully delivered to the destination node over time. The plot shows that the proposed system resembles a linear curve in delivering the packets with minimal variations. Hence, our proposed model can be considered as a linear model. Since the value of the collaborative indices is set to arbitrary values, we observe different response time from different collaborators. This subroutine scheduling of execution of subtasks helps in determining the set of most suitable collaborators with sharable resources. Also, the time taken for data aggregation and reorganization is reduced as the completed set of subtasks are priority organized by the publisher whenever a response is received. The application layer is also responsible for aggregating data and schedules the task i.e. to either notify the potential collaborators or organize incoming responses.



(a). Mean hop count per 100 observations

(b). Packet Dropped over time

(c). Mean Congestion

(d). Packets Delivered over time

Figure 3. Recorded QoS measures of the proposed model

6. Conclusion

In this paper, we have proposed a scalable decentralized community-based network architecture that primarily encounters security bottlenecks and introduces collaborative operations mostly suitable for ad hoc networks [20, 24]. Some intriguing fields in which such decentralized schemes can be adopted are Home IoT networks, VANETs, UW-WSN etc., [9]. The results obtained with the help of extensive simulation studies illustrate that Autonomous Decentralized Systems (ADS) are more effective when collaborative computing mechanisms are adopted [23]. However, one of the main challenges in such a decentralized scheme is increasing the efficiency of resource management and the routing mechanism [22, 25, 26]. In future, we plan to investigate the supportability of meta-heuristic approaches for solving path optimization problems and consolidate the routing module by adopting fuzzy based ant-colony optimization systems. Moreover, the collaborating index of the community network is measured specifically for homogeneous networks. We plan to measure the performance of heterogeneous community networks with time-variant propagation delays [18] and estimate a mathematical model for determining the response time using the same procedure as described in this paper. We also plan to determine the accuracy of Bayesian models for such decentralized schemes. After the validation of these experiments, we can come to a comprehensive understanding of whether such Autonomous Decentralized Systems exhibit congruent behaviors in varied environments and are scalable to accommodate the requirements of large-scale distributed networks [23, 24].

References

[1] M Razzaq, M Qureshi, S Gill, S Ullah. Security Issues in the Internet of Things (IoT): A Comprehensive Study. *International Journal of Advanced Computer Science and Applications (IJACSA).* 2017; 8(5): 383-388.

[2] L M Taraouco, L M Bertholdo, L Z Granville, L M R Arbiza, F Carbone, M Marotta, J J C De Santanna. *Internet of things in healthcare: Interoperability and security issues*. IEEE International Conference on Communications (ICC). 2012: 6121-6125.

[3] S Subhashini, V kavitha. A Survey on security issues in service delivery models of cloud computing. *Journal of Network and Computer Applications*. 2011; 34(1): 1-11.

[4] K Ragab, T Ono, N Kaji, K Mori. An Efficient Communication Technology for Autonomous Decentralized Community Information System. *IEEE ISADS*. 2003;

[5] M Rajesh, JM Gnanasekar. Congestion control scheme for Heterogeneous Wireless Ad Hoc Networks using self-adjust hybrid model. *International Journal of Pure and Applied Mathematics*. 2017; 116(21): 519-536.

[6] K Mori. *Autonomous Decentralized Systems: Concept, Data Field Architecture and Future Trends*. 1st Int. Sym. On ADS, (ISADS '93), IEEE. 1993: 28-34,

[7] OMNet++: https://www.omnetpp.org/; Accesed 13th December 2017.

[8] MatplotLib: https://matplotlib.org/; Accesed 12th April 2018.

[9] Paul D H Hines, Seth Blumsack, Markus Schlapfer. Centralized vs Decentralized Infrastructure Networks; *arXiv preprint; 1510.08792*; 2015.

[10] Zhuang SQ, Zhao BY, Joseph AD, Katz RH, Kubiatowicz JD. *Bayeux: An architecture for scalable and fault-tolerant wide-area data dissemination*. 11th International Workshop on Network and operating systems support for digital audio and video, ACM. 2001: 11-20.

[11] Ingham, David B, Fabio Panzieri, and Santosh K Shrivastava. Constructing dependable Web services. *Advances in Distributed Systems*, Springer. 2000: 277-294.

[12] Jung, Jaeyeon, Balachander Krishnamurthy, Michael Rabinovich. *Flash crowds and denial of service attacks: Characterization and implications for CDNs and web sites*. 11th international conference on World Wide Web, ACM. 2002: 293-304.

[13] Costa, Paolo. Socially-aware routing for publish-subscribe in delay-tolerant mobile ad hoc networks. *IEEE Journal on selected areas in communications.* 2008; 26(5): 784-760.

[14] Paul D H Hines, Seth Blumsack, Markus Schlapfer. *When are decentralized infrastructure networks preferable to centralized ones.* 50th Hawaii International Conference on System Science. 2017: 3241-3250.

[15] S Chan, R Donner, S Lämmer. Urban road networks–spatial networks with universal geometric features ?. *The European Physical Journal B.* 2011; 84(4): 563–577.

[16] Bashan, Y Berezin, S V Buldyrev, S Havlin. The extreme vulnerability of interdependent spatially embedded networks. *Nature Physics.* 2013; 9(10): 667-672.

[17] L G Roberts. Beyond Moore's law: Internet growth trend. IEEE Computer. 2000; 33(1): 117-119

[18] Selivanov, A, Fradkov, A, Fridman, E. Passification-based decentralized adaptive synchronization of dynamical networks with time-varying delays. *Journal of the Franklin Institute*; 2015; *352*(1): 52-72.

[19] Mehmeti, F, Spyropoulos, T Performance analysis of mobile data offloading in heterogeneous networks. *IEEE Transactions on Mobile Computing*; 2017; *16*(2): 482-497.

[20] Mori, K Trend of autonomous decentralized systems. *Distributed Computing Systems, 10th IEEE International Workshop on Future Trends; 2004;* 213-216.

[21] R Subramanyan, J M Alonso, J B Fortes. *A scalable SNMP-based distributed monitoring system for heterogeneous network computing*. ACM/IEEE Conference on Supercomputing; 2000; 14.

[22] Jegan Govindasamy, Samundiswamry Punniakody. A comparative study of Reactive, Proactive and Hybrid Routing Protocol in Wireless Sensor Network under wormhole attack. *Journal of Electrical Systems and Information Technology.* 2017.

[23] M E Coimbra, M Selimi, A P Francisco, F Freitag, L Veiga. Gelly-scheduling: *Distributed Graph processing for service placement in community networks*. 33rd Annual ACM Symposium on Applied Computing; 2018; 151-160.

[24] Hao Yin, Dongchao Guo, Kai Wang. Hyperconnected Network: A Decentralized Trust Computing and Networking Paradigm. *IEEE Network*; 32(1): 112-117.

[25] H Banerjee, S Murugaanandam, V Ganapathy. Low-Energy Aware Routing Mechanism for Wireless Sensor Networks*. International Journal of Engineering Research in Computer Science & Engineering (IJERCSE)*; 2018; 5(1): 112-117.

[26] S Murugaanandam, K Sundaran, V Ganapathy. Comparison of cluster based Routing Protocols in Wireless Sensor Network: A Recent Survey. *International Journal of Pure and applied Mathematics*; 2016; 114(7): 559-571.