

Incrementally Deployable Data Centric Network Architecture by Applying Particle Swarm Optimization

Bhushan S. Chaudhari, Swadipa S.Kulkarni and Sonalee Das

Department of Information Technology, Sandip Institute of Technology and Research Center, Nashik, India.

ARTICLE INFO

Article history:

Received: 9 May 2015;

Received in revised form:
14 April 2016;

Accepted: 20 April 2016;

Keywords

Ad-hoc network,
Particle swarm optimization,
Dynamic route guidance,
Future proof,
Incrementally deployable.

ABSTRACT

Uttermost of Internet traffic is coordinated of applications where users are engrossed in procuring the data from server, which they access from the host. In other words, today's Internet architecture is host-centric. This paper elaborates the need of data-centric architecture over host-centric architecture. But there are no possibilities to deploy pure data-centric architecture practically in one night. This is an incrementally deployable network architecture that successfully supports both services that is host-centric and data-centric network. And withal being data-centric and incrementally deployable, DCNA additionally fortifies mobility and multi-homing features effectively [12].

© 2016 Elixir All rights reserved.

Introduction

Data Centric Network Architecture (DCNA)

Current Internet architecture was designed in 1970s and it is host-centric. The hosts are generally named by their IP addresses. The current Internet is not data centric that means it does not have any mechanism that is directly having data i.e. file, stream, etc. and Services.

Data Centric Network Architecture (DCNA) has a shim layered service binding (SB) layer. DCNA is based on the shim layer. This shim layer is situated in between the transport layer and the application layer with suitable interfaces so that these layers can get connected. Instead they are associated with domain. For example, when we search any image file (Tajmahal.jpg) hosted by Google <http://www.google.com/img/Tajmahal.jpg>.

In this, <http://www.google.com> is the domain name and is used to locate the destination host and route the packet requesting the image to the destination host. This data is not persistent; it depends on where it locates and who manages it. For example, if image of Tajmahal is hosted by one another domain i.e. <http://www.baidu.com/figure/Tajmahal.jpg>.

The development of internet was tremendous in past few years. Majority of current Internet usage has changed for data retrieval and service access (i.e., being data-centric). Hence, growing internet should be designed around the data-centric model [1- 6]. In data-centric model, the user can directly obtain the desired data without knowing the location of data simply by inputting the persistent name. Having data-centric and also incrementally deployable features, PSO is also used for getting expected result as per Clients demand in short time. The Domain Name System hides the location of data when it maps the domain name to IP addresses. It is easy for user to obtain data or services in the current internet due to non persistent data names. When we enter URL to obtain desired datum, then the HTTP error 404 is encountered. Therefore cannot obtain desired data,

although the desired data may be on the same server, but it has been shifted from one dictionary to another dictionary.

There are many data-centric architecture have been proposed till now. But most applications, routers, transport layer protocols, and end hosts are host-centric. It is not possible to deploy the pure data-centric model in overnight. There are billions of routers, end hosts, servers across the world. It is not possible to upgrade all of them at once. Also It is not possible to replace the millions of routers because routers are extremely expensive. Some internet Service providers have transformed their networks from IPv4 to and it was standardized for more than 10 years. Therefore DCNA is both data-centric and host-centric. It is incrementally deployable and supports multi homing and mobility[12].

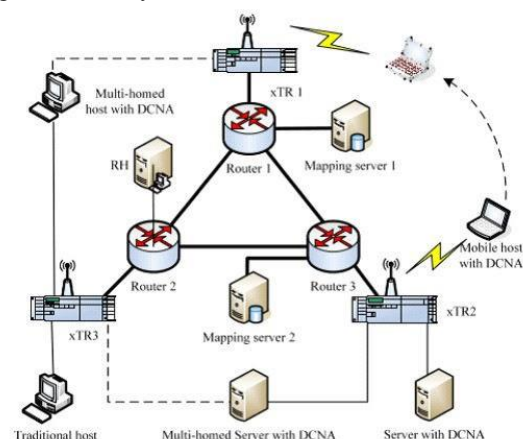


Fig 1. Illustration of DCNA for data-centric accommodations [15]

Particle Swarm Optimization (PSO)

The Particle Swarm Optimization describes, a concept for the optimization of non linear functions using particle swarm methodology. Dynamic route guidance is the most important part of the intelligent transportation system. Particle Swarm Optimization (PSO) is a bionic algorithm that simulates the bird

flocking, fish schooling. It has the advantages like small individuals, simple calculation, and robustness. The effective route guidance can reduce the travel time of drivers, and it provides the route of destination in a very short period and avoid congested road segments, so it raises the road network efficiency. The PSO is not so expensive computationally in terms of both speed and memory requirements. A satisfying simulation is based upon: nearest-neighbor velocity matching and "craziness". A population of birds was eventually recognized arbitrarily with a spot for each on a torus pixel grid having X and Y velocities. At every iteration a loop in the program was resolute, for every agent (a more suitable term used in place of bird), the agent which was its proximate neighbor to that it, assigning agent's X and Y velocities to that agent in focus. Essentially this rule engendered a synchrony of kineticism. Haplessly, the flock quickly settled on unanimous, and in the direction which did not change.

Therefore this created a stochastic variable known as craziness which came into reality. Some changes were done and integrated to those randomly chosen agent's X and Y velocities at each iteration which introduced deviation which was enough to give the simulation an fascinating as well as "lifelike" look into the system. And the behavior of the agent population is now more like a swarm rather than that of a flock. The phrase swarm has a substructure in the literature. There are the five core principles of swarm intelligence.

1. The proximity principle in which the population of birds must carry out simple time and space computations.
2. The quality principle is that the bird's population in this environment must be able to react to quality aspects.
3. The third principle is diverse replication in which the agent population should not commit its activities along extremely narrow channels.
4. The fourth principle is the stability principle in which the agent population cannot not transmute its mode of behavior whenever the environment condition changes.
5. The fifth principle of swarm intelligence is the adaptability in that the agent population must be able to transmute its mode of behavior when it's worth the computational price.

The fourth and fifth principles are the antithesis sides of the same coin. The concept of particle swarm optimization (PSO) has all these five fundamental principles. Rudimentary to the paradigm are multi-dimensional space calculations that are carried out through a series of time steps. The agent population is reacting quickly to the quality aspects pbest and gbest. Allocating the replications between pbest and gbest ascertains the diversity of replication. Only when the gbest changes the agent population also changes adhering to the principle of stability which is fourth principle. The population is very adaptive because it changes when gbest changes. The term particle was chosen as a compromise. The agent population members are mass-free and volume-free, and thus could be called as "points," it is always that the velocities and accelerations are opportunely more applied to particles, even if each one is defined to have arbitrarily minuscule mass and diminutive volume. Further, Reeves discussed particle systems which consisted of clouds of rudimentary particles were the models of diffused objects such as clouds and smoke. Thus the label the authors had chosen to represent the optimization concept was particle swarm [13]. Kennedy and Eberhart introduced the Particle Swarm Optimization (PSO) [4]. It was motivated by the swarming behavior and nature of animals, social behavior of human, flocks of birds and herding development in vertebrates. PSO is a population optimization

algorithm that could be effectively applied very easily to solve different functions of problems due to optimization. As an algorithm, the main ability of PSO is its expeditious convergence. A swarm of particles is a population of many particles, in which every particle is in a motion state or is in a kineticism state which can pass through the search space and can be magnetized to the more desirable positions. PSO should have a fitness assessment function to analyze which positions are best, the function can take the position of the particle and can assign it a value of fitness. Then the aim is to optimize the fitness function and this fitness function is pre-defined and also dependent on the problem. Each particle has its own coordinate and velocity to change the direction of flying in the search space. Following the current optimum particles, all the other particles goes through the search space. Every particle contains a position vector „z“, through which the candidate solution can be represented to the problem, a velocity vector „v“, and a memory vector „pid“, which is more better candidate solution confronted by a particle.

$$Z_i = \{z_{i1}, z_{i2}, z_{in}\}$$

$$v_i = \{v_{i1}, v_{i2}, \dots, v_{in}\}, i=1,2,\dots,n$$

Where n is size of swarm. The previous experience of ith particle is expressed as:

$$pid_i = \{pid_i1, pid_i2, \dots, pid_in\}$$

„pgd“ is another memory vector which is the best candidate Solutions that is encountered by all the particles, is used. Then the particles are operated according to the following equations:

$$vid(t+1) = wvid(t) + j1_rand(pid_i - zid(t)) + j2_rand(pgd - zid(t)) \dots\dots\dots(1)$$

$$zid(t+1) = zid(t) + vid(t+1) \dots\dots\dots(2)$$

Where, -w an inertia weight, which is used to control the trade-off between the global and local exploration abilities of the swarm of particles. A sizably voluminous inertia weight facilitates ecumenical exploration, whereas a small one tends to facilitate local exploration. In order to get a better global exploration, w can be gradually diminished to get a better solution. wj1 and j2 - two positive constants rand – uniformly engendered desultory number.

The Eq. (1) represent that in calculating the next velocity, the previous velocity, the best location in the neighborhood about the particle, and the global best location, these all contribute some influence to the next velocity. The velocities of the particles in every dimension can reach to a maximum level of velocity v max, which can be defined to the category of the search space in every dimension. Position of each particle of the swarm is upgraded by Eq. (2)[16].

Algorithm steps

1. Initialize each particle with desultory position and velocity.
2. Fitness for each particle of the swarm is calculated (individual best).
3. Record of all the individual's highest position is kept (global best).
4. The velocity predicated based on the values of each particle's best and global best position are modified by Eq. (1).
5. particle's position is updated by Eq. (2).
6. Terminate when the condition is met [16].

An Analysis on the Advantages of the Basic Particle Swarm Optimization Algorithm

(1) PSO is stand on the intelligence. The PSO can be applied for technology use and also in scientific research.

(2) In PSO calculation is very simple and easy. If it is compared with other complicated calculations, it has more astronomically immense optimization capability and it can be easily completed.

(3) The authentic number code can be adopted by the PSO very easily which is decided directly by the candidate solution. And the number of the dimension is equipollent to the constant of the solution.

(4) As PSO has no overlapping and complex calculation. Therefore, the result and the search is completed very quickly. Only the most optimist particle of the swarm can transmit information to the other particles during the development of several generations, and the researching speed is very expeditious [13].

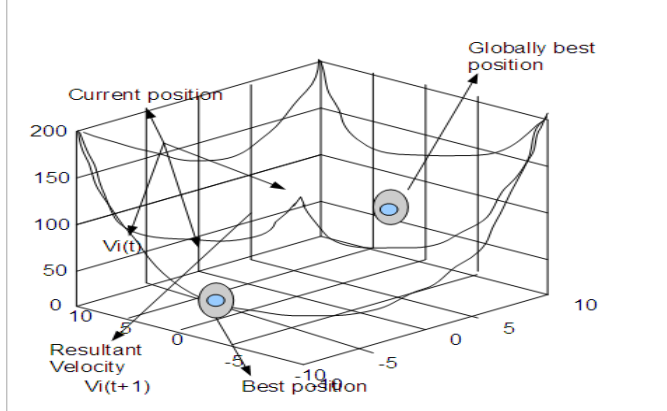


Fig 3. Illustrating the velocity upgrading scheme of basic PSO [16]

Mobility

At the large amount of hosts are connected to the Internet that expected to be mobile, the new network architecture must efficiently support mobility. There are two types of Mobility i.e host mobility and service mobility. In the Host mobility, it sanctions to move a device between IP subnets, while perpetuating to be reachable for incoming requests and maintaining sessions across different subnets. Opposite in the service mobility, it allows users to maintain access to their accommodations while moving or transmuting contrivances and network accommodation providers [12].

Multi-homing

The DCNA must support the multi-homing. The users access the several networks and they provide respective benefits to users. They provide the benefits like ubiquitous connection, reliability, and cost effectiveness. Most of the devices in present internet, they have more network interfaces that assigned various IP addresses. It is difficult to move the ongoing sessions between interfaces. This is possible using the Multi-homing with this new architecture very effectively.

The DCNA has a shim layer or SB layer between application layer and transport layer. Use the appropriate interface to connect these layers efficiently. We build DCNA implementation using Linux environment and satisfies the design goals.

We can describe this using example shows in fig.4[12], the communication initiate between source and destination using SID. If the destination initiates the communication with source uses the IP address which is same. In this case, assume that destination is multi-homed to two subnets and it has two IP addresses named as IP dst 1 and IP dst 2. Assume that, the destination required for obtaining a service named as SID and hosted by source IP src. IP dst1 established a connection to the source firstly, if IP dst 1 is not available then established the connection with IP dst2.

The best use of multi-homing is the redundancy. Multi-homing is also used for load-balancing. In this case the destination established simultaneously two connections to the source. And obtain the data in percent as per given and rest of

data obtain from another source. In both the cases the service is assigned at the source, the destination does not change.

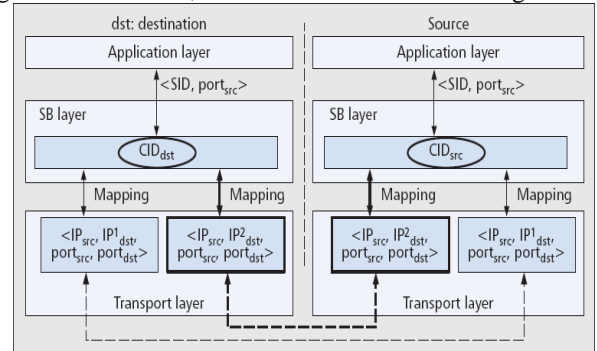


Fig 4. Illustration of multi-homing support in DCNA when the source and the destinations initiate a communication using the SID

Incrementally deployable

The DCNA is the incrementally deployable, as mentioned above. The fig.5 shows communication between the old node and the new node. The source cannot give them to the SID. The destination node uses the IP address of the source and initiates the communication. The fields of IP src, IP dst, port1, and port2 in the SA interface of the destination of application and sends a request to the SB layer of the destination. The SB layer assigns an unused CID1 to the service and request sends to the transport layer in the similar way that as an application sends a request to the transport layer in current Internet. A request receives by source node of the application layer, in the similar way that as a request is sent from an application in current Internet. In this way the source and the destination are communicate with each other. When service obtains from old nodes then, a new node has also the benefit of data-centric. The new node creates the multiple connections to multiple old nodes and these connections maps onto that same CID assigned for the datum. In the case of failure of connection, the new node can obtain service from the remaining old nodes.

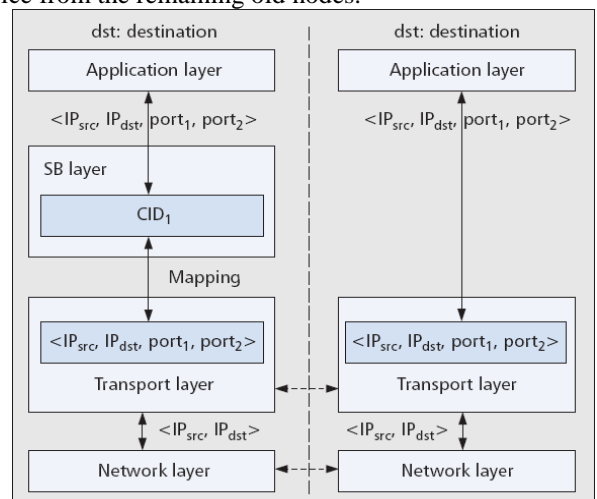


Fig 5. Illustration of communications between a node with the current Internet architecture and a node with DCNA Data Centric

DCNA supports the data-centric service. This illustrate using example shown In fig. 6, suppose a destination with IPdst needs to create a service with SID hosted by two sources, i.e. Source 1 with IPsrc1 and Source 2 with IPsrc2. For obtaining service named as SID, the application layer of the destination has to send a request to the SB layer of the destination. That is old node; there is communication between old node and new node. Since, SID is not supported by the source node. Therefore the destination node can only establish the communication.

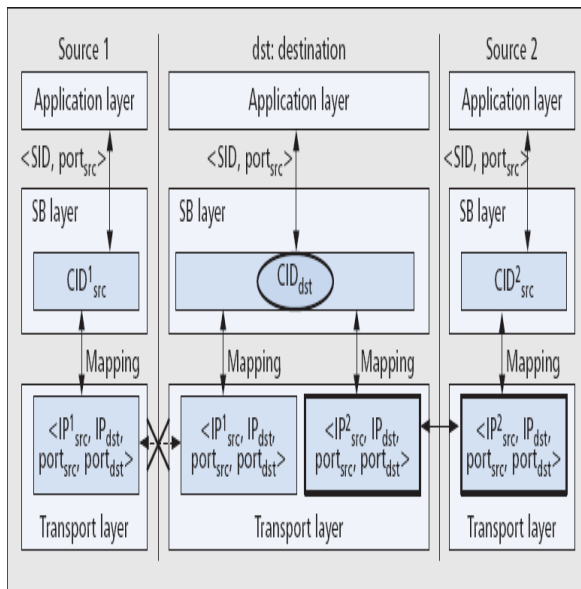


Fig 6. Illustration of DCNA for data-centric services

Literature Survey

To support data based services many data-centric network architectures have been suggested in recent years: content centric networking (CCN) ,network of information (Net-Inf) ,data-oriented network architecture (DONA), and publish-subscribe Internet routing paradigm(PSIRP).

The principles behind Content Centric Networks (CCN) were first explained in the original 17 rules of Ted Nelson’s Project Xanadu in 1979. CCN (Content -Centric Network) which transforms host -centric services to data-centric services. In this network, contents of file are distributed into many numbers of chunks and each one of them allocating unique name that follows hierarchical structure. CCN has some flaws. The first problem is that it needs changes in the basic network operation, which is a big obstacle. The second problem is scalability.

In year 2006, at UC Berkeley and ICSI the DONA project suggested an information centric network architecture, which enhanced TRIAD by authenticity and persistence in the architecture. PARC Research Fellow Van Jacobson gave a talk on August 30, 2006, titled “A new way to look at Networking” at Google. DONA (Data Oriented Network Architecture) is little similar to CCN i.e it allocates a unique name for each and every service with public-private key pair. Whereas the content names in CCN are hierarchical,and data names in DONA are flat and self-certifying. A node which is involved in the data that sends FIND packets to the resolution infrastructure, which routes FIND packets to find the data closer to the requesting node. Once the desired data is found, then it is sent to the requesting node using standard IP routing and also using forwarding techniques. DONA’s limitation is that it does not supports data-centric network approach but not host-centric network method and it is not a incrementally deployable network. The PSIRP (Public- Subscribe Internet routing paradigm) project got started on January 2008 which is expected to end in September 2010. This architecture eliminated the current Internet architecture defects such as inefficient mobility support, distributed denial of service (DDoS) attacks, spam mails[12].

Mathematical Model

The Particle Swarm Optimization (PSO) Algorithm

The first persons who recommended the Particle Swarm Optimization (PSO) algorithm were Eberhart and Kennedy. As shown in Fig. 6, it commences by desultorily initializing a population of birds where each bird is called a “particle”. Each

particle recollects the best solution which is found by the entire flock of birds along the search trajectory. The velocities and also the positions of these particles are upgraded by the following equations:

$$V_{id}^{New}(t+1) = w \times V_{id}^{old}(t) + c_1 \times rand() \times (P_{id}(t) - x_{id}^{old}(t)) + c_2 \times rand() \times (P_{gd}(t) - x_{id}^{old}(t)), \tag{3.1}$$

$$x_{id}^{New}(t+1) = x_{id}^{old}(t) + V_{id}^{New}(t+1) \times \Delta t, \tag{3.2}$$

In the following equations there are two acceleration constants called as the social parameter and the cognitive parameter which are commonly set to 2.0. The function rand () engenders a evenly arbitrary value in the same range. Inertia weight is the parameter here. Eberhart and Shi [15] proposed that.

In equation (3.1), the particle’s precedent velocity (), its best position (), and also the global best position () helps in determining the incipient velocity of a particle. It is mandatory that a maximum limit value on the velocity should be imposed. If the threshold is exceeded by computed incipient velocity, then it is made to avert this particular particle from flying over the desired solutions in each iteration. Equation (3.2) designates how each particle’s position is upgraded in the search space predicated on their kineticism over a time interval, which is conventionally set to 1.

The PSO algorithm is influenced to find the global solution rather than to find the local one, but amendments on its precision and also the speed of convergence are very much desired. A person named, Zahara suggested a hybrid NM-PSO method in which the PSO method and the NM method are combined, and this method is applied to two study cases having exemplary results [6]. This paper specifies a remodeled version of the method presented by Zahara, the M- NM-PSO method, which produced even better outcomes.

Comparison of existing system and proposed system

Sr.no.	Method	Technique	Characteristics	Efficiency
1	CCN	PIT(hierarchical)	PKC with name	50%
2	DONA	Data-driven	PKC with real world	60%
3	PSIRP	Rendervous	Locate publication and scope	70%
4	DCNA	PSO	Fast access, incrementally deployable, Host centric, Multi homing ,Mobility	90%

Proposed Work

The main motive is to plan a new architecture of a network that is both data centric as well as incrementally deployable .Thus the end hosts which are implementing the new network architecture have to be suitable with the existing hosts in order to be incrementally deployable. From the perspective of the existing and new end hosts, it is sensible that we only need to make possibly minimum changes to the host stack.

Therefore we need to design a network architecture which is a efficient search technique, saves time and is also customer friendly[12]. It should provide efficient working of network and also provide fully automated network services. Today all the search engines are host centric. So the proposed system can be implemented with any search engine.

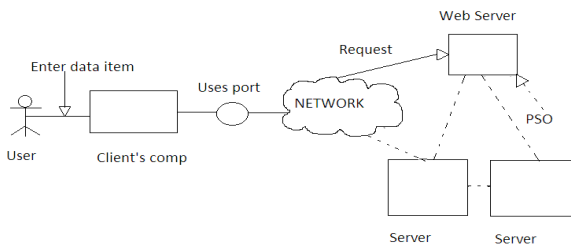


Fig 7. Architecture of DCNA

Result Analysis

This is a graph based on result analysis of time versus hop count which shows relationship between time and hop count. This graph shows the analysis of four different entities and how they are fetched. This result analysis shows that in less time more number of hop counts are covered which means covering more distance in less time and the image is fetched in less time. For example, in graph the paper.txt is fetched in 1s crossing 4 hop counts. Therefore this analysis shows that in less time period the information can be retrieved faster using DCNA.

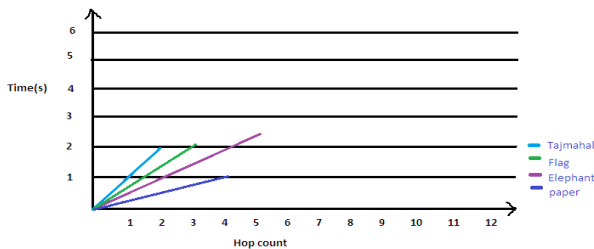


Fig 8. Graph based result analysis of PSO for DCNA

Data items	Hop count	Time(sec)
Tajmahal.jpg	2	2
Elephant.jpg	5	2.5
Paper.txt	4	1
Flag.jpg	3	2

Fig 9. Table based result analysis of PSO for DCNA

Conclusion

We recommended an incrementally deployable data-centric network architecture (DCNA) utilizing Particle Swarm Optimization for the Internet. DCNA utilizing PSO is able to fortify both data-centric and host-centric accommodation because of a shim layer (i.e. the service binding layer) which is integrated between the application and transport layers. In this paper we have specified DCNA's framework and outlined a scope of benefits suggested to be DCNA. We built a proof-of-concept prototype implementation and evaluated the feasibility of DCNA. Based on experiments carried out on the prototype, we have verified that DCNA is able to efficiently support service mobility and multi-homing and in addition, is incrementally deployable. While DCNA is also able to support host mobility, the handover delay is long and cannot satisfy the requirement of delay sensitive applications. However, this could be addressed by network layer mechanisms such as LISP, which has also been implemented in the prototype. We note that in spite of our proof-of-concept implementation, there are still many more open contentions, including sizably voluminous-scale experiments, and we apprehend that this work will motivate further research activities on cognated to future Internet architectures .

Acknowledgement

We would like to thank our Head of Department Prof. Amol Potgantwar and our project guide Prof. Bhushan Chaudhari for their guidance and reference at every step. Therefore we would like to extend our sincere gratitude towards them. We will be always remain grateful for provision of expertise, and technical support extended by our guide, for the successful completion of our quality base paper. We also thank to Prof. Vijay Sonawane for their valuable comments.

References

The books and reference materials used during the pre-development stages of the project include:

- [1] H. Balakrishnan et al., "A Layered Naming Architecture for the Internet," ACM SIGCOMM Computer Commun. Review, vol. 34, no. 4, Oct. 2004, pp. 343–52.
- [2] D. Trossen, M. Sarela, and K. Sollins, "Arguments for an Information-Centric Internetworking Architecture," ACM SIGCOMM Computer Communication Review, vol. 40, no. 2, April 2010, pp. 2733.
- [3] V. Jacobson et al., "Networking Named Content," Proc. ACM CoNext'09, Dec. 2009, Rome, Italy. IEEE Network, July/August 2014
- [4] B. Ahlgren et al., "Design Considerations for a Network of Information," Proc. ACM ReArch'08, Madrid, Spain, Dec. 2008.
- [5] T. Koponen et al., "A Data-Oriented (and Beyond) Network Architecture," ACM SIGCOMM Computer Commun. Review, vol. 37, no. 4, Oct. 2007, pp. 181–92.
- [6] P. Jokela et al., "LIPSIN: Line Speed Publish/Subscribe Inter-Networking," ACM SIGCOMM Computer Commun. Review, vol. 39, no. 4, Oct. 2009, pp. 195–206.
- [7] P. Nikander, A. Gurtov, and T. R. Henderson, "Host Identity Protocol (HIP): Connectivity, Mobility, Multi-Homing, Security, and Privacy over Ipv4 and Ipv6 Networks," IEEE Commun. Surveys & Tutorials, vol. 12, no. 2, Second Quarter 2010, pp. 186–204.
- [8] D. Meyer, "The Locator/Id Separation Protocol (LISP)," The Internet Protocol J., vol. 11, no. 1, Mar. 2008, pp. 23–36.
- [9] R. Shacham et al., "Session Initiation Protocol (SIP) Session Mobility," IETF RFC 5631, Oct. 2009.
- [10] T. Dreiholz et al., "Stream Control Transmission Protocol: Past, Current, and Future Standardization Activities," IEEE Commun. Mag., vol. 49, no. 4, Apr. 2011, pp. 82–88.
- [11] T. Wolf, "In-Network Service for Customization in Next Generation Networks," IEEE Network, vol. 24, no. 4, Aug. 2010, pp. 6–12.
- [12] Hong bin Luo, Hongke Zhang, Moshe Zukerman, and Chunming Qiao, "An incrementally deployable network architecture to support both data-centric and host-centric services", IEEE Network, July/August 2014
- [13] James Kennedy and Russell Eberhart, "Particle swarm optimization", IEEE Trans. on Systems, Man, and Cybernetics, vol. 29, no. 3, June 1999, pp. 1774–1779.
- [14] Qinghai Bai, "Analysis of particle swarm optimization algorithm", Vol. 3, No. 1 February, 2010
- [15] Prof. Bhushan S. Chaudhari, Miss. Poonam B. Sasale and Miss. Pooja V. Yeole, "An Incrementally Deployable Data Centric Architecture Using Particle Swarm Optimization Algorithm Authors", IJETST- Vol. 10, Issue 10, Pages 1668-1674, December, ISSN 2348-9480.
- [16] G. Komarasamy and Amitabh Wahi, "Improving the cluster performance by combining PSO and K-MEANS algorithm", ICTACT journal based on soft computing, April 2011, Volume: 01, Issue: 04