ISBN 978-93-84422-80-6
7th International Conference on Advances in Science, Engineering, Technology and Natural Resources (ASETNR-17)
Dec. 27-28, 2017 Pattaya (Thailand)

Research Trends of Autonomous Control Network for Applying the Bio-Inspired Algorithm

Jungha Jin[1], Ye-Won Oh[2] and Keecheon Kim[1]

[1] Dept. of Computer Science & Engineering, Konkuk University
[2] Dept. of IT Convergence Information Security, Konkuk University

Abstract**:** *A variety of life forms that exist in the natural world, have been created evolving an efficient action rules for their own survival and the species of prosperity. And the attention from a variety of survival principle of these life forms, called Bio-Inspired Algorithm to be applied to create the algorithm so as to be able to be applied in an environment other than the natural world. Natural environment itself has been included in various uncertain change, by that effects of problems how to efficiently utilize the limited resources environment, these Bio-Inspired Algorithm, can provide a rapid adaptability to the conversion of the application environment, resources stably can provide scalability adaptability in constrained environment, can provide numerous advantages in terms of interoperability. When such a Bio-Inspired Algorithms try to apply in terms of the network, in the former case, indicates that it is possible to easily provide an autonomous network configuration, in the latter case, resources such as IoT environment it is possible to provide the interoperability of a constraint type of environment. Thus, it is studied to connect to the network this Bio-Inspired Algorithm, recent field of networking research issues and act in complementary, can provide a synergistic effect. Bio-Inspired Algorithm techniques that can be used by applying congestion phenomena and Synchronization of nature in a network environment, has a variety of existing, autonomous control at SDN(Software Defined Networking) through a research to take advantage of this or incorporated into the configuration of the network, it is possible to develop in a direction to provide a more efficient interoperability of the same resource constrained environments with IoT environment. For the implementation of these Bio-Inspired autonomous control network environment, it summarizes which established OpenFlow environment and newly emerged technology P4:programming protocol-independent packet processor, present the autonomous implementation the Bio-Inspired control network for future.*

Keywords: *Bio-Inspired Algorithm, Software Defined Networking, OpenFlow, P4: Programming Protocol-Independent Packet Processors*

1. Introduction

Bio-Inspired Algorithm is an algorithm that models behavioral rules of various living things in nature. It functions that Various organisms that are members of the natural world can survive through simple and efficient rules of behavior without artificial control, maintain the ecosystem of life, Environmental management, and survival and breeding of species though mutual synchronization.

Bio-Inspired Algorithm derived from nature have an autonomous algorithm functions in a horizontal structure rather than a centralized algorithm. Natural life means that it is a bottom-up type distributed algorithm that's not a top-down type that can represent a consistent form in the ecosystem in which it belongs, by executing the behavior rules defined by each individual. Bio-Inspired Algorithm derived from nature world are being studied not only in engineering but also in humanities and sociology. Especially in computer engineering, It is applied to networks to optimize and synchronize.

In this paper, we analyze qualitatively the Bio-Inspired Algorithm such as Ant Colony Optimization Algorithm, Bee Algorithm, Firefly Algorithm and Flocking Algorithm, which have been actively studied and examine the Bio-Inspired Algorithm as a solution to the main issue in constructing the autonomous control of the network. In addition, research trends of technology research for application of autonomous control network of Bio-Inspired Algorithm will be examined and research trends for efficient and interoperable adaptive networking technology will be examined.

2. Bio-Inspired Algorithm

Example of Bio-Inspired Algorithm include classical methods such as the Genetic Algorithm that model biological systems, the Ant Colony Optimization(ACO) Algorithm that model the process of ants using pheromone secretion to find optimal pathways, the Firefly Algorithm that describes the synchronization process of the life cycle of the organism, and the Bee Algorithm that models the behavior of food searching for life. and It is ordered movements by flock concentration, which can be easily observed in more sophisticated and complex forms of biotic communities such as flocking birds, wild animal flocks, fish flocks, and bacterial flocks. In other words, studies are being conducted to model self-ordering phenomena in Flocking Theory.

In the field of communication and networking technologies, the need for Bio-Inspired Algorithms is explained as an example of a similar environment in biological systems and communication networks in nature and describe the classic Ant Colony Optimization algorithm, Bee Algorithm, Firefly Algorithm, Flocking Algorithm, and Hudding Penguins Algorithm, which are being studied in order to solve the problems in existing network and networking technologies.

2.1. Ant Colony Optimization Algorithm

Ant Colony Optimization Algorithm is a method of observing the habit of ants and imitate the behavior of ants transport the fastest route for the feed that far away from the ant's nest. When ants come out of their nest and look for feed, they secrete pheromone on their way, which allows ants without an eye to transport feed to ants along pheromone that remain on their way. In this way, when ants carry feed using diverse routes, a number of pheromone accumulate on the route with the shortest distance between the ants and feed for a certain period of time. At the result, the ants recognized the pheromone accumulation path as the shortest route and select the pheromone to carry the feed.

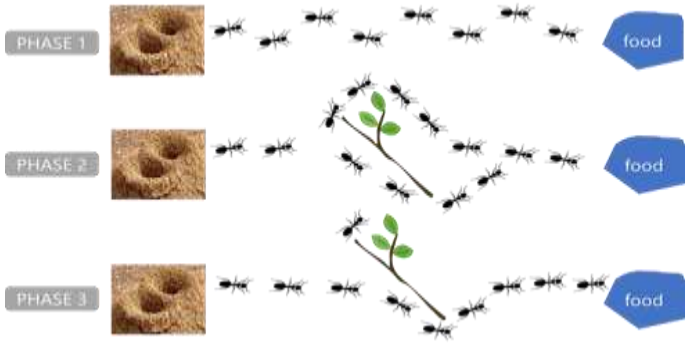

Fig. 1: Ant Colony Optimization Algorithm

The Ant Colony Optimization Algorithm has been applied to various network optimization problems such as scheduling problems in a network, mobile path search.[1][2]

2.2. Bee Algorithm

Bee Algorithm is a Bio-Inspired Algorithm technique inspired by the phenomenon of nature that bees collect honey from flowers. In the Bee Algorithm, the role of a bee is divided into a Scouter which plays a role in finding flowers with lots of honey around, a Forager which collects honey, and an Onlooker which wait in a honeycomb, all of bee carries out three roles alternately. When the scouter finds flowers, which have a lot of

honey, it changes roles from the scouter to the forager which collects honey. and the forager relays to the onlooker about flower's location throughout the specific waggle dance and collect the honey by flying in the direction of collecting continuously. the onlooker plays a role of the forager which collect the honey from flowers on the location where the forager was informed after being located in the honeycomb. When the collection of all the honey in the area is completed, the forager come back to the honeycomb and selects one of the roles between the scouter and the onlooker. The Bee Algorithm has been applied to various optimization techniques and discussed mainly in Statistical Quality Control, Job Scheduling, and Robotics.[3][4]

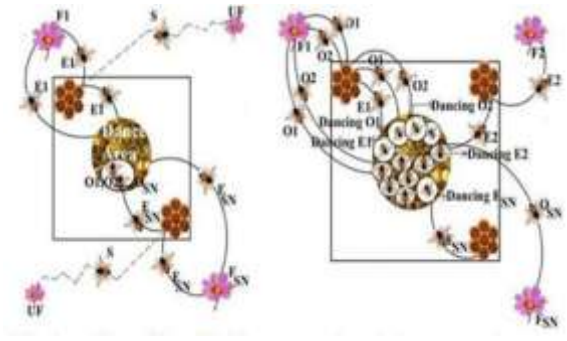

Fig. 3: Bee Algorithm

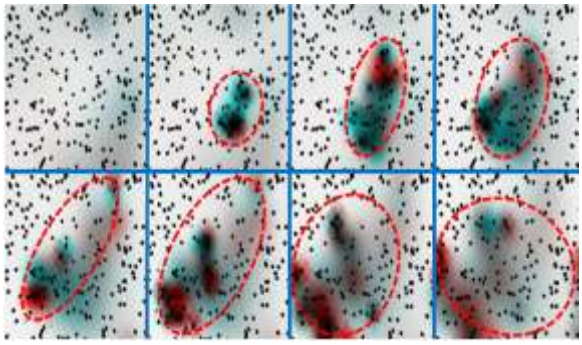

Fig. 2: Firefly Algorithm

2.3. Firefly Algorithm

Firefly Algorithm is collective synchronization algorithm that was developed with inspiration form firefly. When fireflies are shining, they shine according to the individual frequency of each object, but as time goes by, each object interacts with the others glitter, eventually shining like a giant firefly at the same time.[5]

The collective synchronization phenomenon can be understood through the Kumamoto model. In the Kumamoto model, the same oscillator are weakly connected to each other, and the synchronization phenomenon of the firefly is modeled assuming that the interaction of the oscillators depend on the sine function of the phase difference.[6] Each object in the initial state that has not been synchronized oscillates with its own frequency. but, as a result of the interaction between the individual entities, the intrinsic frequency is changed to increase the bond strength between the entities. When the numerical value exceeds the natural frequencies of the entities, synchronization occurs between the entities, resulting in the same phase or frequency. The Firefly Algorithm effectively models the collective synchronization phenomenon and has been applied to the network synchronization.

2.4. Flocking Algorithm

Biotic community phenomenon means a phenomenon in which disordered individuals in nature form themselves an ordered state without external artificial action. Three laws is important in the biotic community phenomenon. They are separation that each individual should keep regular intervals from each other, alignment that they control their direction of movement using an average value of the direction of movement of other individuals around, and cohesion that they decide their location by maintaining regular intervals from other individuals around. [7]

Fig. 4: Bird Flocking Algorithm

Fig. 5: Fish Flocking Algorithm

https://doi.org/10.17758/ERPUB.ER12174010

Flocking algorithm is an algorithm inspired by movements of various flocks in nature. Self-Ordered phenomenon that migratory birds fly in a flock maintaining specific shapes or that shoals of small fish form groups to protect themselves from external invaders is modeled and used in various fields such as resource management, scheduling management, node placement in networks. [8][9]

The Flocking model, which is also called the Cucker-Smale model, is defined by F. Cucker and S. Smale as being separated by three rules of simple population theory: Separation, Alignment, Cohesion. Processing method, and converges to a specific value after a predetermined time elapses. The Cucker-Smale model provides a theoretical basis for obtaining optimized solutions in areas requiring distributed processing. [10][11][12][13][14]

2.5. Huddling Penguins Algorithm

The Huddling Penguins Algorithm imitate the way a group of emperor penguins survive in a cold Antarctic environment to maintain a constant body temperature by sharing their body temperature with each other.[15][16] The emperor penguins keep their body temperature against each other in order to minimize their heat loss from the cold winds of the Antarctic, maintaining a constant body temperature by the movement of penguins at the center and boundaries of the flock. The penguins on the border move naturally in the direction of the wind and imitate the phenomenon that the penguins in the center are circulated to the boundary. The Huddling Penguins Algorithm is modeled on average to maintain the same energy consumption.

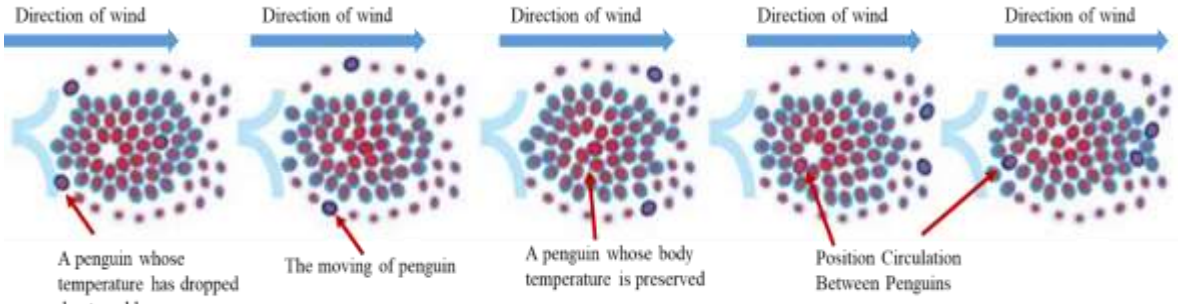

Fig. 4: Huddling Penguins Algorithm

3. Autonomous Control Network

New emerging autonomous control network services require new network policies and protocols to meet these requirements as applications become more diverse and complex. However, Since the internet environment has a wide range of influence based on a large number of users, there are many difficulties and limitations in changing the physical structure of the internet itself due to introducing a new protocol or improving performance. In addition, when abnormal traffic occurs, or net events occur in the network, the network administrator establishes a new policy in order to solve the problem, and converts it into a form that can be apply in the network. In this case, an inefficient processing occurs. The SDN(Software Defined Network) is one of the future Internet technologies that can be efficiently accommodate these changes. SDN is an open platform that can be directly programmed by separating the part of packet forwarding and the network control function and making the design and operation process easier through the network abstraction.[17] In the initial SDN, a centralized environment is adopted. Recently, SDN using multiple controllers is emerging for control autonomously the problem that occur in various environments.[18] This chapter discusses the necessity of applying the Bio-Inspired Algorithm mentioned above to the main problems in the network.

3.1. Extreme Gain of Networking Environments

The size of the future Internet is expected to increase geometrically in terms of the number of constituent nodes.[19] In the case of configuring a Mobile Ad-Hoc Network(MANET) and implementing a Sensor Network, it is necessary to consider the maximum number of nodes form dozens to hundreds of nodes for future application services. The increase in the number of nodes connected to the network causes a very heavy load on

the network. The traffic load will exceed the capacity of the network, which leads to congestion on the transmission path and packet loss due to collision. It will seriously damage the reliability of communication. If we try to solve this problem by choosing the appropriate number of nodes, the period of control information for guaranteeing reliability by increased of the complexity of the system increases with the size of the network. At the same times, the routing table for path determination and the overhead in exchanging control information for table update increase proportionally because as the network size increases, the number of connectable paths increases, and the search for the optimal path increase geometrically. In the existing network technology, since a large number of nodes are installed in a large-scale network environment such as the MANET and the sensor network and effective communication and maintenance are impossible, networking algorithms should be scalable to the network size change autonomously and adaptive operation. It is necessary to apply the Bio-Inspired Algorithm that can be performed.

3.2. Autonomous Network Environment

The existing network environment should dynamically configure the communication system composed of the static communication channel or the network structure to come in the future that includes node mobility, vast amount of traffic, bandwidth requirement to guarantee QoS, channel state, and network structure. In the MANET, when the node moves, the size of the network, the transmission distance, the channel status, and the communication link are continuously updated[20]. Also, in the case of the tracking system of the sensor network, the amount of traffic generated by the sensor in the corresponding area is rapidly increased and then decreased according to the movement of the node, in this case, the existing static network management scheme causes inefficiency in network capacity because it cannot cope with the dynamic load that occurs. Therefore, In order to construct an autonomous network, It is necessary to implement a Bio-Inspired algorithm, especially and Artificial Immune System of a biological system, such as algorithm which can adapt to environmental changes or an Activator-inhibitor System, adaptive Bio-Inspired algorithms are applied to detect changes efficiently and adapt to predicted system patterns.

3.3. Resource-Constrained Environment

Changes in the network environment are repidly proceeding in the form of a plurality of nodes connected to the network. As the network technology develops, it is required to guarantee the service quality level such as the service amount and the bandwidth, but the connection on the physical network shows the limit. It should be able to utilize resources efficiently by utilizing resource configuration environment. the Bio-Inspired Algorithm is an ecosystem-based algorithm that can only use resources efficiently in a limited environment. Therefore, it is possible to provide a more intelligent resource allocation scheme in a resource-constrained environment, which is a problem of a developing network. For example, in the case of the Ant Colony Optimization algorithm, ant can calculate the optimal route between feed and ant species in an efficient manner, which can be applied to route estimation in a cost-effective manner in a network environment.

3.4. Interoperability

The goal of the Internet of Things is to connect individual object to the network and extend them to large-scale wireless networks. However, there is a problem of solving the problem that the nodes constituting the internet of objects will be composed of various heterogeneous types is a real problem. For the heterogeneous network connection, the capability of the communication terminal and the communication protocol between different types should be modeled and effectively managed. In the naturel word, which is the basis of the Bio-Inspired Algorithm, various kind of connection and management of heterogeneity are observed, and it can be largely divided into a bio-system and a constituent part of a living organism. In the case of living system, many organisms can maintain a table internal state even in disturbance from the outside, which is possible by maintaining homestasis based on mutual cooperation between heterogeneous systems such as Nervous system, Endocrine system, and Immune system. The Bio-Inspired Algorithm based on living system can be applied to the development of networking technologies with heterogeneous connection structures. On the other hand, living

organisms such as group of insects are operating with different responses to external environmental stimuli. the Bio-Inspired Algorithm based on organism's composition can be combined with networking techniques to optimize task assignment and process selection through collective intelligence.

4. Bio-Inspired Algorithm Autonomic Control Network Implementation Technology Trend

SDN is a network architecture or a net paradigm, and OpenFlow is one of the sub-interface technologies for SDN. If SDN is a conceptual architecture, OpenFlow is a protocol for communication. In order to apply the Bio-Inspired Algorithm to the autonomous control network, we will look at the big concept of SDN, the sub-interface OpenFlow technology, and the newly studies P4: Programming Protocol-Independent Packet Processors technology.

4.1. SDN(Software Defined Networking)

SDN is a networking technology that can flexibly handle network routing and control and complex operation management with software programming. The SDN separates the data plane and the control plane of the network so that the network operator can control the communication functions in the data plane in various ways through the programming of the control palne with various situation in mind. It is a network control and management technology that enables flexible control and effective network management based on the separation of the network control function from the switch and centralization by securing the visibility of network resources. SDN is based on the two basic principles.

First, SDN should be capable of Software Defined Forwarding, which means that the data forwarding function that is processed by hardware-based switches and routers must be controlled through open interfaces and software. And in SDN, the development of advanced network management tools should be made possible through abstraction. These abstraction tools include the ability to monitor events across the network and control events or network elements such as topology changes and new flow inputs. It is aimed at possible Global Management Abstraction. The conceptual structure of SDN is represented by three hierarchial structures which are an infrastructure layer, a control plane, and an application layer. The infrastructure layer is located at a data transfer device that performs L0 to L3 switching functions, the control plane is a network controller that controls network operations using a global view of the overall network state. and the application layer is located in an application operating in the upper layer. Southbound interfaces and Northbound interfaces exist for seamless interoperation between layers.[21]

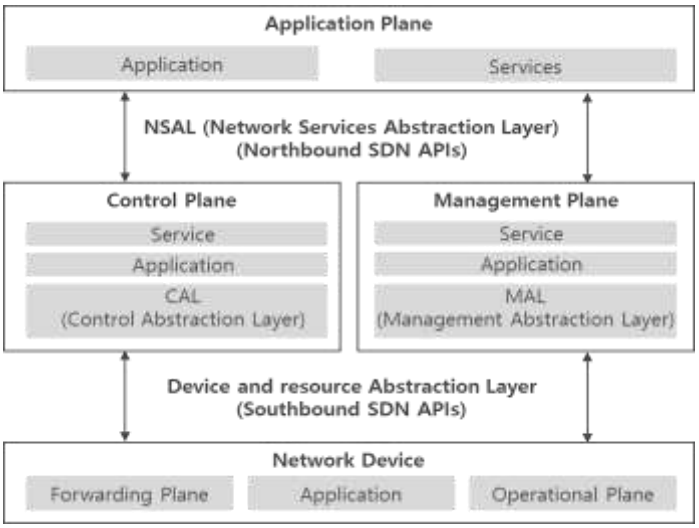

Fig 5: SDN concept structure

The network management function is concentrated in the SDN controller, so that it manages the network in a comprehensive manner and network design and operation can be configured in a simple form of network. Finally, the network manager does not need to manually manage the configuration information distributed in a plurality of devices by simply configuring the abstracted network as a program, and is easily controllable.

4.2. OpenFlow

Since OpenFlow was standardized by ONF(OpenFlow Network Foundation) in 2009, network venders such as CISCO and HP are constantly launching SDN switches. Open source-based controllers that control SDN network also include Ryu, FloodLight, and OpenDayLight. In a network implemented with SDN, the nodes are connected to the OpenFlow switch, and the OpenFlow switches are connected to the OpenFlow controller. It is possible to monitor the status of OpenFlow Controller which is directly connected, indirectly connected nodes with OpenFlow switches, and the SDN network itself in centralized form. Based on centralized monitoring, the OpenFlow Controller can provide services while managing resources within the SDN network, and these services are provided in the form of application service programs. Since the application can control the SDN network by participating in the controller by Northbound using the API, the programmable environment becomes possible, and open source controllers support various services and protocols considering this point.

Using these technologies, new protocols and services can be designed and implemented in the SDN network, and the addresses of nodes in the SDN network can be collected and managed. In the SDN-based network, not only the nodes of the existing network, but also the management of the mobility nodes through the OpenFlow Controller can be performed without going through the home agent and the foreign agent, so that the routing mechanism in the SDN network can be managed through the OpenFlow Controller.

4.3. P4: Programming Protocol-Independent Packet Processors

Since SDN places only the data plane on the existing Internet routers and puts the control plane on the upper controller, the distributed control and management of the routing is handled by the single controller, so SDN can set and monitor network routing paths. However, the packet handling method still has constraints that cannot be programmed and changed. Among them, P4: Programming Protocol-Independent Packet Processors is emerging as a representative method.

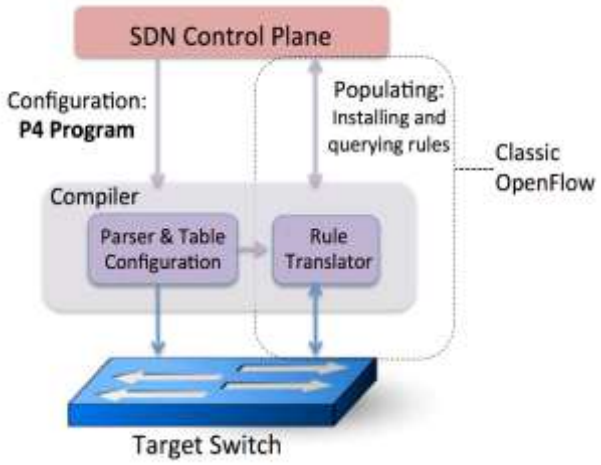

Fig. 7: P4: Programming Protocol-Independent Packet Processors concept structure

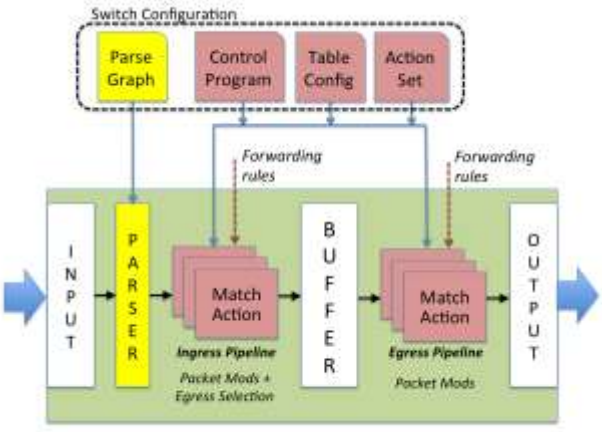

Fig. 6 : P4 Packet processing delivery model concept

P4 is a programming language that makes it possible to program packet processing methods on programmable switches. P4 represents the relationship with the API, and a forwarding table is designed to reside by fixing it as a function of the switch. In contrast to C and Python, P4 is a language specially designed for network data transfer and is designed to enable packet transfer data plan programming. P4 should provide reconfigurability so that the controller can immediately redefine the syntax and processing of the packet. Switches are protocol-independent so as not to be dependent on a specified packet format, and have an application-independent goal so that they are not system dependent. Since P4 is designed to be protocol-independent, it is interpreted by a program compiled specifying the protocol header format and field name for the program and the target device, and it is processed in order.

P4 programming packets are designed to be compilable for a variety of general-purpose CPUs, FPGA systems, network processors, ASICs, and execution systems, and are designed to be independent of system implementation. Using P4 on each system, it must be implemented in an executable format and include a compiler that can convert P4 programming packets into source code.[22]

The packet delivery model through P4 consists of two components: configuration and implementation. The parser of the operation program sets the order of the matching work steps and designates a specific header field to be processed in each step. The configuration information determines how protocols are supported and how the switch handles packets. The behavior of the implementation adds or deletes entries in matching tasks that are specific to the configuration. Implementation implies that the policy is determined by the packet under any circumstances.

Using P4 technology, SDN network can provide more flexible and efficient network infrastructure control and management than existing SDN, and provide more advanced interoperability. Inaddition, by implementing the Bio-Inspired Algorithm to implement the network, it can provide a lot of utility in the implementation of the adaptive autonomous control network.

5. Conclusion

In this paper, we have studied a method to solve the problems of the current network by using Bio-Inspired Algorithm. Future network development is under the theme of autonomous control network environment and various connections between different types are being studied, in order to use it, it is necessary to apply various Bio-Inspired Algorithm to networks. In order to implement autonomous control network environment by applying Bio-Inspired Algorithm, SDN environment which is the base technology is examined, and OpenFlow technology is summarized to implement SDN. However, due to the constraints of OpenFlow technology, it is difficult to support heterogeneous interoperability or adaptive autonomous control network configuration. Thus, using the net P4: Programming Protocol-Independent Packet Processors technology, we can solve the constraints in the OpenFlow environment with programmable packets. Domestic researches related to implementation technology using intelligent packet in autonomous control network environment combining with Bio-Inspired Algorithm are lacking.

Through the study of the above technique, programmable packets in an autonomous network are processed, thereby ensuring interoperability in heterogeneous systems and networking operations in a more efficient manner in a resource-constrained network environment.

6. Acknowledgements

This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIT) (2014-0-00547, Development of Core Technology for Autonomous Network Control and Management) and Next-Generation Information Computing Development Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science and ICT (No. NRF-2017M3C4A7083678).

7. References

[1] Dorigo, M., Birattari, M., Stutzle, T., “Ant colony optimization”, IEEE Computational Intelligence Magazine, vol. 1, no. 4, pp. 28-39, Nov. 2006.

[2] M. Dorigo, C. Blum, “Ant colony optimization theory:A survey”, Theoretical Computer Science, vol. 344, pp.243-278, Nov. 2005.

[3] Karaboga.D, “An idea based on honey bee swarm for numerical optimization.”, Technical Report TR06, Erciyes University, Engineering Faculty, Computer Engineering Department, 2005.

[4] Pham DT, Ghanbarzadeh A, Koc E, Otri S, Rahim S and Zaidi M. “The Bees Algorithm,” Technical Note, Manufacturing Engineering Centre, Cardiff University, UK, 2005.

[5] G. Werner-Allen, G. Tewari, A. Patel, R.Nagpal, and M. Welsh,“Firefly-Inspired Sensor Network Synchronicity with Realistic Radio Effects.”InSenSys, 2005.

[6] J.A. Acebron, L.L. Bonilla, C.J. Perez-Vicente, F.Ritort, R. Spigler, The Kuramoto model: “A simple paradigm for synchronization phenomena”, Rev. Mod. Phys., vol. 77, pp. 137~185, 2005.

[7] Craig W. Reynolds, “Flocks, herds, and schools: A distributed behavioral model”, ACM Computer Graphics, vol. 21, no. 4, pp. 25-34, 1987.

[8] Olfati-Saber, R., “Flocking for multi-agent dynamic systems: algorithms and theory,” IEEE Transactions on Automatic Control, vol.51, no.3, pp. 401- 420, March 2006.**)**

[9] J. A. Carrillo, M. Fornasier, J. Rosado, and G. Toscani,“Asymptotic flocking dynamics for thekinetic cucker-smale model,” SIAM Journal on Mathematical Analysis, 2010.

[10] Hui Yu, Ji-Gui Jian, “Flocking motion control of mobile agents based on distance-dependent adjacency matrix”, International Conference on Wavelet Analysis and Pattern Recognition, vol. 1, pp. 17-22, Aug. 2008.

[11] Seung-Yeal Ha, Taeyoung Ha, Jong-Ho Kim,“Emergent Behavior of a Cucker-Smale Type Particle Model With Nonlinear Velocity Couplings”, IEEE Transactions on Automatic Control, vol. 55, no. 7, pp. 1679-1683, July 2010.

[12] Demetriou, M.A., “Distributed parameter methods for moving sensor networks in unison”, American Control Conference, 2008, pp. 273-278, June 2008.

[13] Schwager, M., Slotine, J.-J., Rus, D., “Consensus learning for distributed coverage control”, IEEE International Conference on Robotics and Automation, pp. 1042-1048, May 2008.

[14] Bin Xu, Kurdila, A.J., Stilwell, D.J., “Geometric ergodicity of the distributional consensus problem in vehicle network control”, IEEE Conference on Decision and Control (CDC), pp. 7499-7506, Dec. 2010

[15] D.P.Zitterbart, B.Wienecke, J.P.Butler, B.Fabry,“Coordinated Movements Prevent Jamming in an Emperor Penguin Huddle,” PLoS one 6(6), e20260, 2011.

[16] A.Waters, F.Blanchette, A.D.Kim, “Modeling Huddling Penguins,” PLoS one 7, e50277, 2012.

[17] Hyun Sik Yoon, Kyoung-Soon Kang, Hak-Suh Kim, Hea-Sook Park, “Considerations for Data Path Design in SDN”, The Journal of The Korean Institute of Communication Sciences 30(3), 2013.2, 22-28 (7 pages)

[18] Sangheon Pack, Insun Jang, Dongeun Suh, Jong-Hwa Yi, “New paradigm of future network About SDN / NFV” The Journal of The Korean Institute of Communication Sciences 32(7), 2015.06, 82-92 (11 pages)

[19] “OUTLOOK: Visions and research directions for the Wireless World,” WWRF(World Wide Radio Forum), no.4, July, 2009.

[20] I. Chlamtac, M. Conti, J.J. Liu, “Mobile ad hoc networking:imperatives and challenges,” Elsevier Ad Hoc Networks, vol. 1, no. 1, pp. 13-64, 2003.

[21] ONF White Paper, “Software-Defined Networking: The New Norm for Networks,” Apr. 2012.

[22] P. Bosshart, D. Daly, G. Gibb, M. Izzard, N. McKeown, J. Rexford, C. Schlesinger, D. Talayco, A. Vahdat, G. Varghese, and D. Walker. P4: Programming protocol-independent packet processors. SIGCOMM Comput. Commun. Rev., 44(3):87–95, July 2014.