Neuroevolution of Augmenting Topologies for Artificial Evolution: A Case Study of Kinesis of Creature in the Various Mediums

Sunil Kumar Jha
Department of Physics, Adani University, India
Sunil.Jha@aii.ac.in

Filip Josheski
University of Information Science and Technology "St. Paul the Apostle," North Macedonia
s8fijosh@stud.uni-saarland.de

Xiaorui Zhang
School of Computer and Software, Nanjing University of Information Science and Technology, China
002224@nuist.edu.cn

Zulfiqar Ahmad
Institute of Hydrobiology, Chinese Academy of Sciences, China
zulfiqar@whu.edu.cn

Abstract: *The motivation of the present study is to evolve virtual creatures in a diverse simulated 3D environment. The proposed scheme is based on the artificial evolution using the Neuro Evolution of Augmenting Topologies (NEAT) algorithm to educe a neural network that controls the muscle forces of the artificial creatures. The morphologies of the creatures are established using the Genetic Algorithm (GA) method based on the distance metrics fitness function. The concept of damaging crossover of neural networks and genetic language for the morphology of creatures has been considered in the morphologies of the artificial creature. Creatures with certain morphological traits consume a large time to optimize their kinetics, thus they are placed in a separate species to limit the search. The simulation results in the significant kinetics of artificial creatures (2-5 limbs) in virtual mediums with varying dynamic and static coefficients of friction (0.0-4.0). The motion of artificial creatures in the simulated medium was determined at different angles and demonstrated in the 3D space.*

Keywords: *Artificial evolution, kinetics, NEAT algorithm, artificial neural network, genetic algorithm*.

Received February 5, 2021; accepted October 31, 2021
https://doi.org/10.34028/iajit/19/6/5

1. Introduction

Artificial evolution attempts to restore the natural phenomenon of an organism in a virtual ambiance. Several approaches have been used in the creation of virtual creatures; the most common are

a. Software approach,
b. Hardware approach,
c. Wet approach [18], etc.

The Artificial Neural Network (ANN) was used as the control unit of the body parts of virtual creatures [14]. The present study is inspired by the 3D artificial life field [14, 18] and aims to evolve virtual creatures that can move efficiently in a virtual environment with varying dynamic and static friction. Genetic Algorithm (GA) is a widely used optimization technique in several applications [1, 3].

GA has been used to select the optimal combination of morphologies and control systems of virtual creatures, in addition, their search in the entire space comprising several hypothetical architectures. The possible solutions have been coded into genotypes analogous to the genetic description of the animals. Thereafter, genotypes have been improved to phenotypes that compete in a simulated 3D environment. The latter process follows the associated laws of Physics in order to make realistic models of creatures capable of moving effectively. The main motivation of the present study for evolving virtual creatures is due to their potential application in of modular robotics. The searching of the optimal combination of morphologies of virtual creatures through multidimensional spaces is tiresome and time-consuming; moreover, engineering solutions designed with optimization algorithms can be used for a more creative approach towards solving the earlier problems. Consequently, a method for automating the search process of the optimal morphology of virtual creatures has been proposed. Especially, the neuroevolution approach has been implemented as a control system for virtual creatures and for perceiving environmental responses inputs, while GA is executed in the creation and the evolution of the morphology of creatures. Neuroevolution has been proven to be a faster and more efficient technique for learning non-Markovian tasks because memory inside an ANN can be represented with recurrent connections [17]. It is basically a combination of ANN [7] and GA [1, 3]. The ANN provides a good control mechanism for evolving artificial creatures by using the sensors (to perceive the outside environment) as the input nodes and actuators (to move the limbs of a creature) as the output nodes. The GA was developed by importing the mechanisms of natural adaptation [1, 3] in subsequent steps:

a. Selecting an initial set of chromosomes, fitness function, crossover probability (p_c), and mutation probability (p_m),
b. Fitness evaluation of initial chromosomes to select the parents for the mating,
c. Applying the genetic operation to produce the new chromosomes,
d. Replacing the older chromosomes and checking the stopping criteria,
e. If stopping criteria is not achieved, then repeating steps (b)-(d).

Mainly, two concurrent approaches of neuroevolution have been used in past research reports

1) Traditional approach based on fixed topology (fixed nodes in the hidden layer of the ANN) [16], in which, the neuroevolution searches for the optimal values of the connection weights between the nodes while the topology remains unchanged
2) Topology and Weight Evolving Artificial Neural Network (TWEANN) [4, 5] since topology is as much important as connection weights.

2. A Survey of Related Studies

The origin of the evolution of 3D virtual creatures was discussed in numerous research reports in the last few years in the published literature, like physically-based organism has been evolved and their struggle for mates and food has been demonstrated [22]; Komosinski and Ulatowwski [10] have created virtual creatures composed of sticks; a system for the interactive evolution of colored 3D creatures in a physical environment has been demonstrated [15]; L-systems have been used for generating encoding of both the morphology and control system of creatures [6]; Shim and Kim [19] have evolved flying creatures, like bats, birds, and butterflies, etc. with double wings based on body-brain co-evolution; Lipson and Pollack [13] have evolved 3D walking virtual robots; the Evolution of Robotic Organisms (ERO) has been tested extensively to encourage diversity and enhance the crossover [11]. In another related study by Lehman and Stanley [12], ERO was used with a novelty search that converges to multiple niches of creatures rather than fittest and optimized morphology. The artificial life approach was implemented in a simulation to search the interface between the population and evolutionary dynamics [8]. Metamorphosis based approach was implemented in the evolution of robots and their locomotion was validated on land and in aquatic surface [8]. A hierarchical NEAT (hNEAT) system is another latest development in the field of artificial evolution [9].

The historical markings in NEAT avoid damaged offspring. The NEAT algorithm has been adjusted and ported to run on the Unity game engine [21]. Each of the creatures was initialized in 3D with several limbs that are varying in a predefined range. The system proposed for artificial evolution in the present study uses the Unity 3D game engine, which has an integrated *Physx* (physics) engine [21]. The *Physx* engine was used to change the coefficients of friction to design different simulated environments. In most of the reviewed approaches, the issue of the production of damaged offspring is been not considered in the generation of virtual creatures. Besides, it is hard to notice the implementation of the combination of the NEAT and Unity 3D game engine for the generation of virtual creatures in a diverse simulated environment. The present study mainly focuses on the evolution of virtual creatures in a diverse simulated 3D environment (with varying dynamic and static friction and acceleration due to gravity) which is not discussed in the reviewed literature. Another difference compared to the previous methods is the implementation of the NEAT algorithm and Unity 3D game engine to avoid the production of damaged offspring using neural network and genetic algorithm in the present study. The main contributions of the study are as follows.

- All simulations were done using the Physx engine of the Unity 3D game.
- Firstly, the simulation of the virtual environments was done using the different values of static and dynamic friction.
- Specifically, three types of the virtual environment were created and used in the study of the kinesis of the virtual creature.
- Additionally, some novel virtual environment was created using the different values of the gravitational acceleration at the constant value of the coefficient of friction to monitor the kinesis of the virtual creatures.
- The detailed representation of the kinesis of the virtual creatures is demonstrated and discussed.

The present study and the reviewed literature are related to the generation of the virtual creatures and their kinesis study. This is the parallel between the present study and the reviewed literature. The basic difference between the present study and the reviewed literature is the approaches that have been used for the generation of the virtual creatures, the approaches that have been used for the generation of the simulated environment to study the behavior of the virtual creature and the future application of the generated virtual creature. In our opinion the proposed approach for the generation of the virtual creatures and simulated environment specifically the variation in the friction and gravity will be helpful in the study and application of the virtual creatures on other planets in the future.

3. NEAT Algorithm

The Neuroevolution of Augmenting Topologies (NEAT) algorithm is different than most of the traditional approaches towards neuroevolution, since, it evolves weights while keeping a fixed architecture and the topology of the neural networks [2, 20]. The NEAT algorithm has some characteristic features including the historical markings which avoid the production of damaged offspring, speciation for a suitable division of genomes, and low dimensionality of search space, etc., [6, 7, 8]. A direct encoding method was used in NEAT to search each node and its connections of phenotype [5]. The NEAT genome functions of node genes and connections genes contain comprehensive information about the nodes, their weights, and networks. A historical markings approach was used in the NEAT to inhibit the creation of damaged offspring by tracking the genes of similar traits [5]. Specifically, an innovation number was assigned to each of the emerging genes for similarity checkup which solves the competing conventions problem in less time. The innovation number of genes was used to decide the crossover operation and to categorize them into matching, excess, and disjoint genes. The disjoint genes are randomly selected from the parents during the crossover. The structural mutations in NEAT (adding a new connection and a node) have been shown in Donate *et al*. [5]. In the first type, a new connection gene was placed at the end of the genome and the subsequent existing innovation number was assigned while in the second type, the connection was disabled before splitting, and two new connection genes were added to the end of the genome. The new node was created in between the two new connections in the resulting network phenotype. The topological improvements in the neural network structure have been insured with the speciation by grouping analogous neural networks in niches. The neural networks compete only with each other, but not with the whole population in each niche. The similarity of the neural network was measured with the compatibility distance [5]. The compatibility distance (d) was used to speciate the genomes using a threshold value (d_t). Each species have been characterized by a random genome from the previous generation (an elite genome), and the genomes from the new population are speciated, according to the compatibility distance metric with the elite genomes from the previous population [5]. The NEAT algorithm starts with minimal topologies and upgrades, the novel structures regularly, which significantly decrease the dimensionality of the search space. The new structures have been introduced by the structural mutations and survive only if they compete with the other in the niche.

4. Evolution of Artificial Creatures

The main array of limbs of creature comprises a set of limbs array in which each of the individual limb arrays contains a segment array (information about the segment, like position, scale, and a limit for the movement in 3D). The angular limit along any one of the axis can be varied while it is fixed for the rest two axes in the Unity game engine. The information about the root of the limb, such as color, height, length, and width was stored in a distinct variable than the limb array. The location of the limb segment outside the surface of the root was determined by the position variable. If the creature morphologies were spawned or initialized, into the game world for simulating their morphology, it was established in the sequence. More precisely, the initialization loop first goes through the first limb in the limb array, and then it goes through the first limb segment in the limb segment array up to the last segment. The loop then initializes the second limb in the limb array and so on. The creature that results from such an encoding scheme has been shown in Figure 1. While creating the initial population of morphologies, there is an option to specify the minimum and the maximum size of the limb segments and the root, the maximum number of limbs and limb segments within a limb, and the area of creatures. A random value of the position of the limbs relative to the root was assumed at the beginning of the simulation. A choice is available to replace the limb and the root components easily with other shapes, like spheres or cylinders.

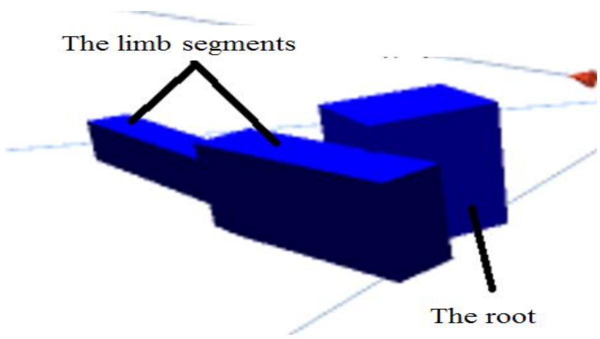

Figure 1. An artificial creature resulting from the morphological encoding.

Each creature has one central control system which is an artificial neural network in NEAT. In the present study, an ANN (9×7×3) was used for this purpose. An initial population of NEAT genomes was generated at the beginning of the simulation. At the same time, the genomes of initial creatures were created with a population size equal to the number of genomes. A distinctive identification number was assigned to each of the genomes in the initial population, after that the NEAT genome was decoded into a neural network (control system of the creature). The ANN loops through each limb segment and read out the properties concerning the position of the joint and its rotation around each axis, and output an actuator movement for each axis of the joint, currently processing. The NEAT loops in linear time through the limb hierarchy, to read the sensor input, and to move the limbs, the ANN has a constant number of inputs which results in less computation. The ANN loops through the joints many times per second and moves them slightly every time, which causes rotation between joints without delay. A flag number was assumed for each of the limb segments during the simulation which changes till the last limb segment was reached which results in limited inputs of the ANN according to the locomotion of the creature. The kinesis was defined as the movement of particular joints by a certain amount of degrees around three axes, at particular periods, in the presence of angular constraints of the joints. Each of the input nodes of the ANN represents a specific sensor. The joint angle sensor measures each degree of freedom of each of the joints of the creature. The timer sensor decides the motion of a particular joint within the interval. The position sensor identifies the joint in the present main loop which establishes a connection between the current time and position of the loop in the joint array. The touch sensor fires if a limb touches the ground and increments the touch counter. Maximizing the contact with the ground supports the kinesis. The actuators enable the creature to move in the virtual environment by applying torques and forces on the joints. Three outputs of ANN control the three degrees of freedom of each joint. The straightforward application of the controller output to the creature joints results in shaking and jittering of the creatures and it threatens the stability of the simulation. Because of this reason, the output of the controller was transformed before applying it to the joints. The ANN outputs were normalized (0-1) and multiplied by 180 (degree of freedom for any joint). After that, rotation quaternion of the joints was computed to stabilize the locomotion of the creature. Control systems (neural networks) of the creature crossover except for their morphology according to some predefined rules in NEAT to pass the advantageous genetic information to the offspring neural network leading to an enhanced fitness score. It is tough to decide which morphological properties of the creature's genomes will mutate in a single control system. The creature with more joints requires additional time to optimize its movement through crossover and mutation. The NEAT speciation algorithm considers the number of creature joints into account when dividing the neural networks into species. The neural networks speciated in this way reflect the differences in the number of joints between the creatures. Similarly, the morphologies and the creature genomes were also speciated.

5. Experiments and Analysis Outcomes

The *Physx* engine included in the Unity 3D game was used in the simulation of the kinetics of virtual creatures. Both the static and dynamic friction was assumed to be responsible for the kinesis of evolved virtual creatures. Consequently, different coefficients of static and dynamic friction were assumed for the simulation. Specifically, by changing the coefficients of static and dynamic friction, three different virtual environments were generated, e.g.,

1) A low static and dynamic friction,
2) An intermediate static and dynamic friction,
3) A high static and dynamic friction medium.

The details about the three different simulating environments have been summarized in Table 1. An initial population of NEAT genomes was decoded into neural networks at the beginning of each of the simulations. Thereafter, the population of the creature's genomes, with the same unique IDs as that of NEAT genomes, was decoded into morphologies and was initialized into the virtual environment. The neural networks control the creatures, those initialized from the identical genome ID. A fitness score was computed for creatures/neural networks based on their performance in the simulated virtual environment. Precisely, the maximum distance traveled from the point of origin in the virtual environment by the creature was assumed as its fitness measure. The creature capable of traveling larger distances, in a specified time interval and in environments with different friction values, was rewarded and has a better probability of passing their genes to the subsequent generation. The rationale for selecting the distance metrics as a fitness function is that creatures with the highest speed and linear movement will travel the greatest distance from the point of origin. Some creatures had better speeds, but did not travel the maximum distance due to their circular motion. An argument was made that the creatures which traveled in circles have their structure optimized by crossover, but at this point, the maximum distance traveled approach was assumed, because it enables the algorithm to produce stable architectures within fewer iterations by using a simple fitness function $(f) = k \times d(x)$, where k is a proportionality constant, and d(x) is the distance function. The concept of damaging crossover of two neural networks has been explored by using a morphological crossover. The damaging crossover produces an offspring where two or more limb segments connect to the root at the same position which delays the movement of creatures. The limb order inconsistency and the change of the connection point of the limb segment with the root also result in a damaging crossover. The movement ability of the creature was affected by limb order inconsistency since the earlier depends on the order in which the limbs have traversed. The change of the connection point of the limb segment also interrupts the movement ability of the creature. Also, because ANN does not control the order of the limbs in the main array, or the position of the limb segments relative to the root, it is up to the crossover to avoid the earlier mentioned issues. A morphological crossover was implemented to avoid the damaging crossover by introducing breakpoints, in which the position of the limb segments that connect to the root was fixed and the order of the limbs in the main limb array remains constant. The morphological crossover algorithm has the following two assumptions:

1. Child is a duplicate of the parent of higher fitness with similar root and limb arrangement,

2. Selection of a random number of arbitrary limb segments (position, size, and joint limits) for the creature of lower fitness to replace the limb segments of the creature of higher fitness.

In the earlier process, the limbs of the creature of higher fitness connected to the root are not subject to any change in their properties, with the exception of the joint limit. The offspring of the NEAT genome crossover and morphological crossover have similar IDs.

Table 1. Parameters of simulated environment.

Physical Parameters	Simulating environment		
	Low friction	Intermediate friction	High friction
Static friction	0.01	0.53	4.00
Dynamic friction	0.00	0.36	3.00
Gravitational acceleration	9.81	9.81	9.81
No. of limbs	2.00	5.00	2.00
Max. limb segments	1.00	2.00	2.00

The morphological mutation of the creature's genome occurs if the NEAT genome with similar IDs mutates. The creature's genome resulting from the morphological mutation has the same ID as that of the NEAT genome. The value of the mutation factor was assumed to be equal to 0.17 during the simulation. The mutation operator was able to change any of the properties associated with the limb segments. Figures 2 and 3 represent the motion outcomes of virtual creatures in three different simulating environments (Table 1). To enhance the visual clarity, the root of the creature has been highlighted with a different color to that of the limbs. From the visual inspection of movement results, it is obvious that some creatures have used the virtual physical environment more effectively than others. The creature evolved for high friction medium has a high exposed surface area on the limbs and tries to lower the friction with the ground, to facilitate better movement (Figure 3). If a creature is completely in contact with the ground of the medium, the torque forces immediately pull it up. The creatures moving in the environment with low friction coefficients have developed an interesting gliding procedure. They possess a low surface area of contact with the ground to facilitate gliding and use the rear leg to push it to the surface and the front leg for stabilizing the steering effect (Figure 3). It is important to note that damaged offspring have been successfully avoided and none of the limb segments connect to the root at the same point. All the simulation (maximum 50 creatures) was done by using 4 GB RAM, Intel i5-2430M 2.40GHz computer. Future work would be to define the additional settings of damaged offspring from the morphological crossover and mutation, also to experiment with the mutation factor by selecting only specific morphological elements to mutate, or mutate all of them (the limb position included) with a strategy to prevent damaged offspring and to conduct simulations in more diverse environments. In Table 1, the gravitational acceleration has been assumed to be constant and equal to its value on the earth's surface. Besides, two other simulating environments were created by assuming constant values of the coefficients of static and dynamic coefficients and changing the values of gravitational acceleration (Table 2).

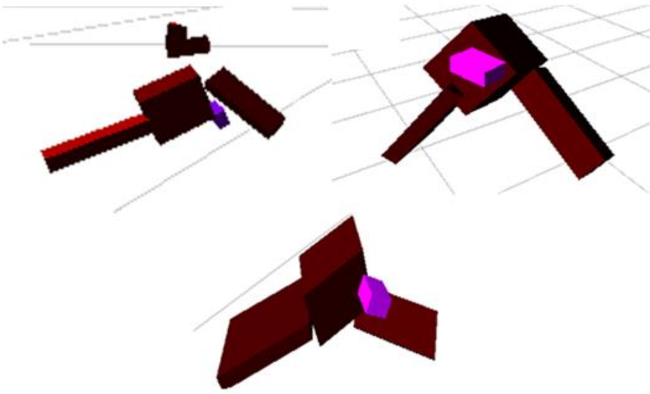

Figure 2. The motion of virtual creatures in the lower friction (0.01) simulated medium observed at three different angles 10°, 80°, and 160°.

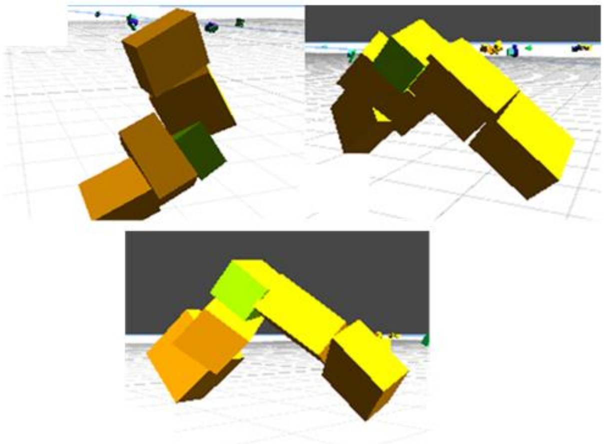

Figure 3. The movement of virtual creatures in high friction (3-4) simulated medium observed at three different angles 10°, 80°, and 160°.

Table 2. Another simulating environments.

Physical Parameters	Simulating environment	
	Low gravity	High gravity
Gravitational acceleration	1.6	24.79
Static friction	0.6	0.6
Dynamic friction	0.5	0.5
No. of limbs	1.0	4.0
Max. limb segments	2.0	2.0

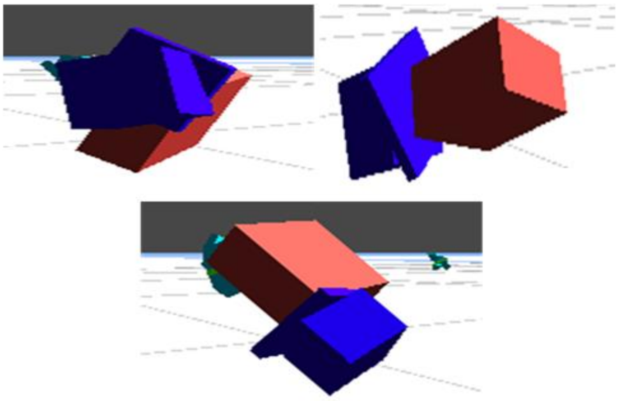

Figure 4. The Motion of virtual creatures in low gravity (1.6) medium at three different angles 10°, 80°, and 160°.

The graphical representation of the motion of virtual creatures in the earlier two environments has been demonstrated in Figures 4 and 5, respectively. The first simulating environment has assumed a value of gravitational acceleration less than that of the earth's surface, and the second has higher than the earth's surface. Subsequently, the earlier two simulating environments signify the possibilities of an effective movement of virtual creatures on another planet.

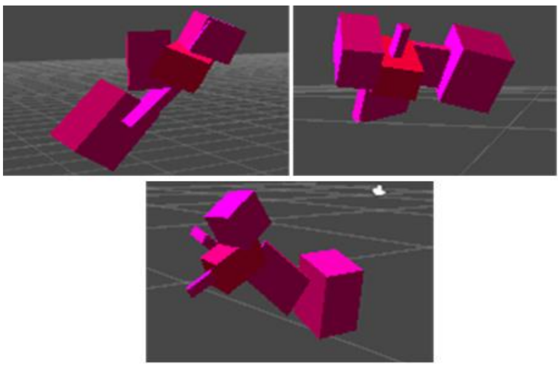

Figure 5. The Motion of virtual creatures in high gravity (24.79) medium at three different angles 10°, 80°, and 160°.

6. Conclusions

The NEAT approach of virtual creature evolution and a neural network as their central control algorithm has been applied successfully. The limitations of damaged morphologies and a crossover method were addressed. Further, a method of speciation has been suggested that considers the creature and the neural network genomes into account. The speciation, the mutation, the crossover, and the simulations were performed by linking the genomes NEAT and creatures with unique IDs (a simple and fast way of their association). The Unity 3D game engine has been selected as an environment to perform the simulations, which simplifies the task of altering the physical constants for creating a new virtual environment. Satisfactory results were achieved with a population number of up to 50 creatures at a time. Creatures have visibly adjusted their structure in the virtual environment for static friction between 0.01 to 4.0, dynamic friction between 0 to 3, number of limbs 2 to 5, and limb segment 1 to 2. Besides, the creatures also exhibited a stable structure for the coefficient of gravity 1.6 to 24.79, the number of limbs 1 to 4 at a constant value of static friction 0.6, dynamic friction 0.5, and limb segment equal to 2.

References

[1] Al-Janabi M. and Ismail M., "Improved Intrusion Detection Algorithm Based on TLBO and GA Algorithms," *The International Arab Journal of Information Technology*, vol. 18, no. 2, pp. 170-179, 2021.

[2] Banzhaf W., Beslon G., Christensen S., Foster J., Képès F., Lefort V., Miller J., Radman M., and Ramsden J., "Guidelines: From Artificial Evolution to Computational Evolution: a Research Agenda," *Nature Reviews Genetics*, vol. 7, no. 9, pp. 729-735, 2006.

[3] Dasgupta D. and Michalewicz Z., *Evolutionary Algorithms in Engineering Applications*, Springer, 2013.

[4] Ding S., Li H., Su C., Yu J., and Jin F., "Evolutionary Artificial Neural Networks: a Review," *Artificial Intelligence Review*, vol. 39, no. 3, pp. 251-260, 2013.

[5] Donate J., Li X., Sánchez G., and de Miguel A., "Time Series Forecasting by Evolving Artificial Neural Networks with Genetic Algorithms, Differential Evolution and Estimation of Distribution Algorithm," *Neural Computing and Applications*, vol. 22, no. 1, pp. 11-20, 2013.

[6] Hornby G. and Pollack J., "Body-brain co-Evolution Using L-Systems as a Generative Encoding," *in Proceedings of the 3rd Annual Conference on Genetic and Evolutionary Computation*, California, pp. 868-875, 2001.

[7] Hu Y. and Hwang J., *Handbook of Neural Network Signal Processing*, CRC Press, 2001.

[8] Ito T., Pilat M., Suzuki R., and Arita T., "Population and Evolutionary Dynamics Based on Predator-Prey Relationships in a 3d Physical Simulation," *Artificial Life*, vol. 22, no. 2, pp. 226-240, 2016.

[9] Joachimczak M., Suzuki R., and Arita T., "Artificial Metamorphosis: Evolutionary Design of Transforming," *Artificial Life*, vol. 22, no. 3, pp. 271-298, 2016.

[10] Komosiński M. and Ulatowski S., "Framsticks: Towards a Simulation of a Nature-like World, Creatures and Evolution," *in Proceeding of the European Conference on Artificial Life*, Berlin, pp. 261-265, 1999.

[11] Krcah P., "Evolving Virtual Creatures Revisited," *in Proceeding of the GECCO*, London, pp. 341-341, 2007.

[12] Lehman J. and Stanley K., "Evolving a Diversity

of Virtual Creatures through Novelty Search and Local Competition," *in Proceeding of the 13th Annual Conference on Genetic and Evolutionary Computation*, Dublin, pp. 211-218, 2011.

[13] Lipson H. and Pollack J., "Evolving Physical Creatures," *in Proceedings of the 7th International Conference on Artificial Life*, Bloomington, pp. 282-287, 2006.

[14] Nakamura K., "Emergence of Adaptive Behaviors for Artificial Creature Using a Combined Artificial Neural Network," *Artificial Life and Robotics*, vol. 21, no.1, pp. 91-97, 2016.

[15] Ray T., "Aesthetically Evolved Virtual Pets," *Leonardo*, vol. 34, no. 4, pp. 313-316, 2001.

[16] Rempis C. and Pasemann F., "An Interactively Constrained Neuro-evolution Approaches for Behavior Control of Complex Robots," *Variants of Evolutionary Algorithms for Real-world Applications*, pp. 305-341, 2012.

[17] Richards N., Moriarty D., and Miikkulainen R., "Evolving Neural Networks to Play Go," *Applied Intelligence*, vol. 8, no. 1, pp. 85-96, 1998.

[18] Rieffel J., Knox D., Smith S., and Trimmer B., "Growing and Evolving Soft Robots," *Artificial Life*, vol. 20, no. 1, pp. 143-162, 2014.

[19] Shim Y. and Kim C., "Generating Flying Creatures Using Body-brain Co-Evolution," *in Proceedings ACM SIGGRAPH*, Goslar, pp. 276-285, 2003.

[20] Thompson A., "Notes on Design Through Artificial Evolution: Opportunities and Algorithms," *in Proceeding of the Adaptive Computing in Design and Manufacture V*, London, pp. 17-26, 2002.

[21] Unity., http://unity3d.com, Last Visited, 2020.

[22] Ventrella J., "Attractiveness vs. Efficiency (How Mate Preference Affects Location in the Evolution of Artificial Swimming Organisms)," *in Proceedings of the 6th International Conference on Artificial Life*, Madison, pp. 178-186, 1998.

Sunil Kumar Jha received the B.Sc. and M.Sc. degrees in Physics from VBS Purvanchal University, India, in 2003 and 2005, and Ph.D. in Physics from Banaras Hindu University, India, in 2012. His research interests include data mining and pattern recognition applications.

Filip Josheski completed B.Sc. from Faculty of Machine Intelligence and Robotics, University of Information Science and Technology "St. Paul the Apostle, North Macedonia in 2015. His research interest includes data mining, machine learning and virtual creature.

Xiaorui Zhang received Ph.D.at Southeast University, China, in 2010. She is a professor in Nanjing University of Information Science and Technology. She has published more than 50 papers in reputed journals. Her research interests include digital forensics, image processing the virtual reality and human-computer interaction.

Zulfiqar Ahmad worked as Research Associate in Department of Environmental Sciences, University of California Riverside, USA. His research interested is in Machine learning and data mining application. He has published 33, SCI articles in well reputed international peer reviewed journals.