Toward the Construction of a Virtual Ecosystem

By Evolving Virtual Creature's Behaviours

Nesrine OUANNES, Noureddine DJEDI
LESIA laboratory / Med Khider Biskra University
Department of Computer Science
Biskra, Algeria
{nesrine.ouannes, noureddine.djedi}@lesialab.net

Yves DUTHEN, Hervé LUGA
Institut de Recherche en Informatique de Toulouse,
Université de Toulouse - CNRS - UMR 5505
Toulouse, France
{yves.duthen, herve.luga}@univ-tlse1.fr

Abstract—**In this paper a virtual ecosystem environment with basic physical law and energy concept has been proposed, this ecosystem is populated with 3D virtual creatures that are living in this environment in order to forage food. Artificial behaviours are developed to control virtual creatures. A genetic algorithm with an artificial neural network were implemented together to guarantee some of these behaviours like searching food. Foods are presented in different locations in the virtual ecosystem. The evolutionary process uses the physical properties of the virtual creatures and an external fitness function that will conduct to the expected behaviours. The experiment evolving locomoting virtual creatures show that these virtual creatures try to obtain at least one of the food sources presented in its trajectory. Our best-evolved creatures are able to reach multiple food sources during the simulation time.**

Keywords; Artifical life; Genetic Algorithm; Recurrent Neural Netwok; 3D virtual creature and virtual ecosystem.

I. INTRODUCTION AND MOTIVATIONS

Researchers in Artificial life attempt to design and construct systems that exhibit some characteristics of living organisms. Among the great variety of biological systems that inspire and guide these researches and according to Bedau [1], three broad areas can be identified according to the basic of their elements: (a) At the microscopic scale, chemical, cellular and tissular systems; Wet ALife synthesizes living systems out of biochemical substances, (b) At the mesoscopic scale, organismal and architecture systems; or the Soft ALife that uses simulations or other purely digital constructions that exhibit lifelike behaviour, (c) At the macroscopic scale, collective and societal systems. In our software we try to blend at least (b) and (c) in the same simulation.

Inspired by the life and food chain process occurring in nature is our main objective to create a virtual ecosystem. Developing artificial behaviours for artificial creatures in such ecosystems by hand is almost an impossible task [2] that can be employed by means of Evolutionary Algorithms.

Evolving behaviours of artificial creatures in three dimensional, physical environments are subject of a variety of works, most of them based on Karl Sims one [3, 4]. Some of these models focus on developing morphologies and behaviours for artificial creatures [5-9]. Other models have attempted only to study the behaviours such as those of Sims creatures as [10]. Note also [11, 12] that examine how an artificial creature acquires adaptive swimming behaviours in a hydrodynamic environment, and [3, 9, 11, 13] for light following task.

Recent works was developed to study the evolution of more interesting behaviours in construction of a virtual ecosystem, like predator prey co-evolution [14, 15] and foraging behaviours [16]. In some approaches behaviours were studied in virtual ecosystems, like the 'Avida' system [17] inspired from 'Tierra' system [18], and the 'Life Drop' system [19] comparable to 'Gene Pool' system [20].

In this paper, we propose architecture to simulate a virtual ecosystem and present the first realized step towards it. This ecosystem is populated with 3D virtual creatures that have to gather food to survive. Artificial behaviours are developed in order to control virtual creatures. The virtual creatures living in the ecosystem are divided in four nodes: producers (plants), 2 kinds of consumers (herbivores and carnivores) and decomposers such as bacteria. Initially, we study the behaviour of herbivorous creatures, which feed on available resources in their environment. In this part of our ecosystem, we present a controller model, which controls physically simulated creatures in a biologically inspired manner: by evolving the neural connection weights of a neural network to obtain foraging behaviours.

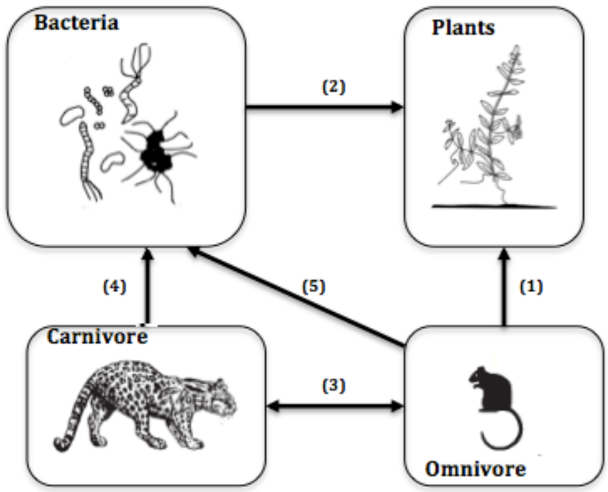

Figure 1. Our proposed Virtual Ecosystem Model. (1) The foraging behavior. (2) The Bacteria that produce nutrient to plants. (3) Predator-prey behaviors. (4 and 5) Are the waste rejected by the animals.

We model a virtual environment with a population of such creatures to study the evolution of some behaviours (walking, following) constrained by their energy consumption in a physically plausible 3D environment. The creatures must survive while maintaining their energy level that is computed according to their metabolism. The energy is gained from eating and decreases by a value that is defined by a cost for motion. This metabolic model makes it favourable for a creature to stabilize its use of energy and to reach as many energy foods as possible until its lifetime expire.

The rest of this paper is organized as follows. We begin with describing the virtual creature model that is simulated using simple Newtonian physics. Then we will show how we adopt a recurrent neural network to control the creature's motions. In the final section two experiments will be presented. In one only single food source was provided. In a more complex problem, a lot of food sources were present. We end with a discussion of the results obtained from the both experiments of evolved behaviours.

II. Models And Methodes

A. *Virtual Creature's Model.*

We create the virtual creature by connecting rigid 3D body parts with joints and actuators. The kinematical model is then generated from, the geometrical data of the virtual creature (link lengths, type and position of joints etc.). We employ open dynamics engine (ODE)[1] as a simulation platform for the virtual creature. ODE is a free, industrial quality software library for simulating articulated rigid body dynamics. It is fast, flexible and robust, and it has built-in collision detection. Therefore, ODE is suitable for a realistic simulation of the physics of an entire creature.

Figure.2 shows virtual creature model, this creature consists of three rectangular boxes jointed with two Ball and socket joints. The virtual creature model presented in this paper is equipped with two identical food sensors like the only information about foods.

We use the recurrent neural network that controls the motions of the virtual creature (locomotion) and the genetic algorithm to evolve more complex behaviours (following or looking for a food). Actuators are controlled by outputs of the recurrent artificial neural network (RNN).

The world in which the creatures live is a three-dimensional, physically simulated environment where energy resources are continuously absorbed by the virtual creatures and transformed to energy.

We use a physically based model of an articulated creature that embodies the nonlinear relationships between the forces and the moments acting at each joint and the appendages etc., and the position, velocity and acceleration of each joint angle.

TABLE I. Body parameters of the simulated creature.

Body part	Geometry	Dimension (m)
Torso	Rectangular box	0.50×0.50×0.50
Right Appendage	Rectangular box	0.90×0.40×0.15
Right Appendage	Rectangular box	0.90×0.40×0.15

[1] ODE: available at: http://ode.org/

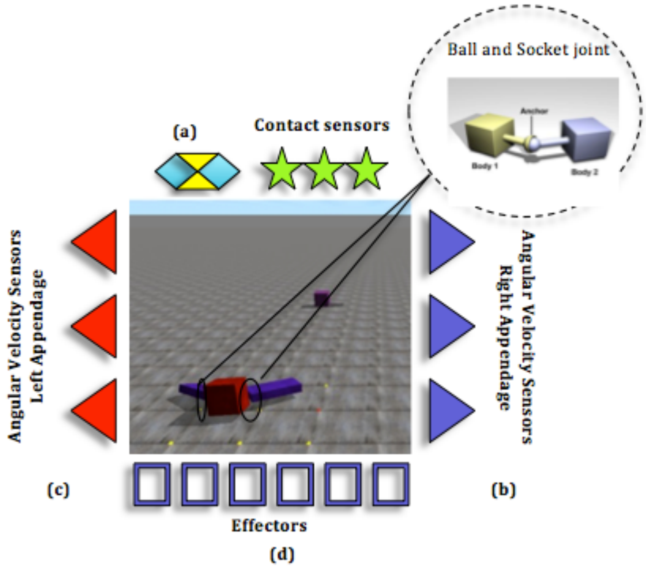

Figure 2. A simple 3-D Artificial Creature model used in our simulation, with a torso and two appendages. This model has (a) two food sensors, a direction sensor and three contact sensors for each rigid body in contact with the ground. The appendages have sensors attached to them, (b) three angular velocity sensors at the right one, and (c) three angular velocity sensors at the left one. It has also (d) six effectors in its joints.

In addition to the geometric data, a physically realistic model requires kinematic data and some physical properties as mass, gravity centre and inertia matrix, for each link and joint, max/min motor torques and joint velocities which are difficult to simulate. To simulate interaction with the environment detection and handling of collisions as well as suitable models of appendages- ground contacts are required. In the context of this simulation we uses the open source Open Dynamics Engine (ODE) library that can handle collision detection for several geometric primitives.

B. *Recurrent Neural Network (RNN)*

An RNN is an artificial neural network [21] that is a well-known brain model. It consists of a set of neurons (units) and set of synapses (arcs) including self-coupling of individual neurons. The artificial creature presented in this paper uses Elman [22] Recurrent Neural Network for its biological plausibility and powerful memory capabilities.

The topology of our network is made up of 4 layers: an input layer, a hidden layer, an output layer and a context layer. The number of neurons contained in the input and output layer depends on the artificial creature's morphology (i.e. the number of sensors and actuators). Each inter-neuron connection within the RNN is assigned a weight (see Figure 3).

The RNN network can operate either in discrete time as common in feed-forward networks, or in continuous time. In the latter case, using a simple neuron model, the dynamical behaviour of the i^{th} node in the network is governed by the equation (1).

$$\tau\iota+\gamma\iota = \sigma(\beta\iota + \sum\omega\iota\varphi \sum\gamma\iota + \sum \omega\iota\varphi I \ \ I\varphi).\ \iota=1,..\nu. \qquad (1)$$

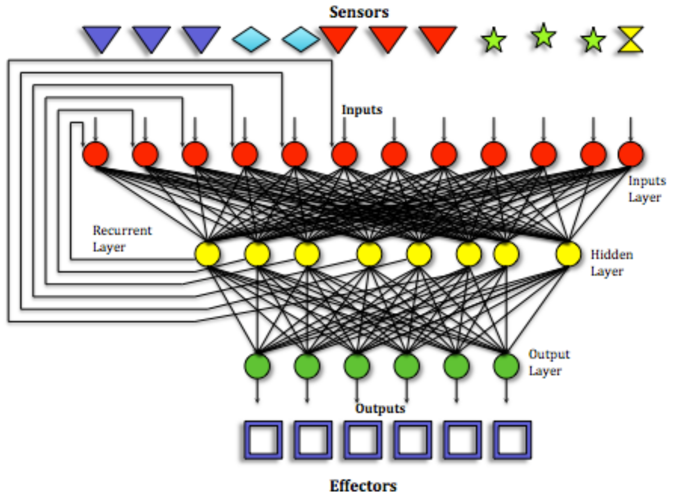

Figure 3. The Recurrent Neural Network Structure. It consists of four layers; input (with 12 neurons), hidden (8 neurons), output (6 neurons) and context layer (6 neurons). There are arcs (inter-neuron connections with weights) and the Time response of this RNN is 5 Time Step.

Where ν is the number of neurons in the network, τ_i are time constants, γ_i is the output (activity) of node φ, $\omega_{\iota\varphi}$ is the (synaptic) weight connecting node φ to node ι, $\omega_{\iota\varphi}I$ is the weight connecting input node φ to node ι, I_φ is the φ^{th} external input to node ι, and β_ι is the bias term, which determines the output of the neuron in the absence of inputs. $\sigma(\,)$ is a sigmoid function whose main purpose is to restrict the activity of the neurons to a given range. The hidden, context, and output layers of the RNN all use the same bipolar sigmoid activation function (see equation 2).

$$\sigma(\chi) = 2/(1 + \varepsilon - \alpha\chi) - 1. \qquad (2)$$

The obvious advantage a RNN has over the traditional feed forward network is "memory." The use of feedback connections allows the RNN to have a "memory" of past events. Thus, pattern presentation to the RNN will take into consideration what moment in time the pattern occurs. Biological neural networks process information in a similar fashion to the RNN.

1) The model of RNN

The virtual creature uses a set of sensors to collect data from the environment and feeds it to the RNN. Sensors that monitor the internal state of the virtual creature, such as joint angles are referred to as proprioceptive sensors. In this setting, the current joint angles of the previous time step of the simulation are used by the evolved controller to compute the next set of motor signals for the artificial creature.

Simulating a virtual creature in a realistic environment most likely requires feedback loops between the artificial creatures control system and its body, as well as between the control system and the environment.

The set of external sensors, which links the creature to its environment. Those sensors can measure values such as the creature acceleration or till relative to a fixed coordinate frame, external forces applied to the bodies, etc.

This artificial creature has:

- Two food sensors at the torso, that return the angle and the distance between the visible[2] food source and themselves,
- Three contact sensors; one in each body part. The contact sensors indicate when a body make contact with the ground plane;
- The Ball and socket joints contain angular velocity sensors that feed the rate of angular change back to the RNN.
- Finally, a direction sensor provides the ANN with a virtual compass.
- We control actuators sets implemented with the artificial creature that are three actuators in each appendage of the artificial creature. There are 12 sensors and 6 actuators in total.

In a control system, such as that used by the artificial creature, specifying the correct outputs for each possible input combination and state is practically impossible. In these situations, optimal behaviour must be learned by exploration.

Genetic algorithms can be used as an optimization process to evolve neural networks that prove to be robust solutions to difficult virtual-world learning tasks.

C. *Genetic algorithm*

In this work, the purpose of the genetic algorithm is to optimize the weights of the neural network, which controls a virtual creature. A synergistic relationship exists between the GA and the RNN. The GA optimizes the RNN, and the RNN produces behaviour via virtual creatures that is then scored. In the case of this work, a good genotype is a set of parameters that causes a creature to perform the desired behaviour, which is to feed more food sources as possible.

1) Chromosome

We use the Genetic Algorithm with a real number encoding. GA optimizes weights of a neural network to acquire adaptive behaviours of the simulated artificial creature. The weights are stored within an artificial chromosome that can undergo different modifications (crossover and mutation). The figure 4 illustrates the chromosome structure.

The evolutionary algorithm is based on a population of 100 genotypes, which are randomly generated. This population of genotypes encodes the connection weights of 100 neural controllers. The evolutionary runs are performed using a fixed length genome that consists of 638 genes, with 1 gene per RNN weight. The number of connections in a neural network represents the number of genes in the chromosome.

1) Fitness Evaluation

In genetic algorithms, the Fitness Function is of extreme importance: it is the operator that, during the evolution, evaluates all the individuals (their phenotypes) of the current generation. In our work to evaluate the fitness of a virtual creature, it is placed in the environment, with randomly placed food particles.

[2] In our experiments the creature's brain at time (t) sees only one food source.

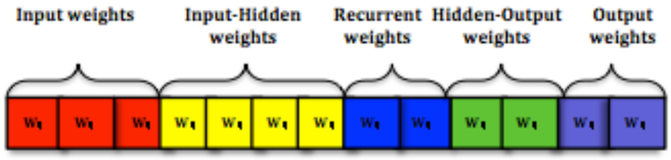

Figure 4. The Chromosome Structure. This gene encodes a recurrent neural network weights, it has five regions which present each one the interconnections between the four layers of the network. The weights encoded in this genome are organized like: input weights, input-hidden, recurrent, hidden-output and output weights.

The evaluation is formulated so as to minimize the cumulated distance between the virtual creature's current place and the nearest food source during the virtual creature's motion and also to minimize the consumed energy E, which is measured by cumulated torques generated by actuators. The equation (3) presents the fitness function we use to evaluate all the individuals.

$$\phi_{\text{fitness}} = \sum_{\tau=0}^{T} |\Psi_{\phi} - \Psi_{\tau}| - \sum_{\tau=0}^{T} \sum_{\varphi=0}^{A} |E_{\varphi}| \qquad (3)$$

Where T is the period of simulation, which is 1000 Time steps that correspond to (10 seconds), $\Psi\phi$ is the food source position and $\Psi\tau$ is the position of the virtual creature at time τ. A is the number of actuators in the virtual creature and $E\varphi$ is the energy spending in each actuator.

2) Genetic Operators

The GA used here makes use of the standard single-point crossover operator. After the crossover operation, the gene has a probability of being mutated. The crossover operation chooses a couple of chromosomes with the crossover probability $P_{Crossover}$, the weights of the first chromosome selected are swapped to these of the other chromosome. The mutation operation chooses a chromosome (a set of weights) and then chose a gene (a weight) from it with the mutation rate $P_{Mutation}$ and perturbs the weight of the corresponding gene; on applying the Gaussian perturbation.

The population size was kept constant and the most parents are replaced with their offspring, for the other parents we use the Elitism; at each generation, the 20% of the population that are seen like best individuals (i.e., the elite) are retained in the subsequent generation. The parameters of the Genetic Algorithm used in our experiments are presented in the table 3.

III. Experiments

We carry out experiments to examine how the simulated virtual creature can acquire following behaviours towards some food sources by walking towards them, and analyse the acquired walking behaviours.

The experimental environment is an open and continuous 3D space without obstacles. It is filled with two types of objects: virtual creatures and food sources. When a virtual creature moves over a food source, its life energy is replenished by a certain amount.

This article describes two experimental setups:

- The case where one single food source is provided: where the fitness is given by measuring the energy level of a virtual creature and the distance travelled at the end of its lifetime.
- With more than one food source, a food source has a given maximum energy capacity which defines its initial energy content. If an artificial creature is in contact with a food source, a certain amount of energy is transferred from the source to the virtual creature and thereby consumed. The best creatures are those who have more energy quantities because after catching one food source they can go to catch another one until its lifetime expire.

We use in our experiments the energy consumption like one of the measures to select individuals in the evolution process, this concept is used to found the best creatures that can reach a lot of food sources by minimizing the loss of energy when it is walking.

The metabolic model makes use of energy as follow:

- The energy is gained from eating and decreases by a value that is defined by a cost for motion. This value is measured by cumulated torques generated by actuators.
- E_i describes an artificial creature's energy content, all creatures in the virtual ecosystems have the same initial value at birth,
- In order to increase the level of energy to avoid death, the artificial creature has to eat food sources that are placed in the virtual ecosystem. A virtual creature with energy equal to 0 dies (it stop motions because there is no more energy to use in its actuators) and after the expiration of its lifetime it will be removed from the population. Each food source provide with the same energy amount.

A. *Physics parameters*

The environment in which evolution occurs is extremely important to the final results. The simulated environment imposes similar constraints on the virtual creature as the natural environment would on a real creature of similar proportions. Different parameters have been used in our virtual creature's model; the table 2 summarize the effect of some parameters that are important to the generation of a realistic motion.

TABLE II. Physics Parameters Used by the Dynamic Engine.

Parameters	Effects
Bodies' mass dimensions and COM (Centre of mass) offset	Affect the maximum forces applicable to the joints
The physics engine errors caused by greater forces and torques	Implicate the violation of the movement's limits of a joint, and thus incurring, unrealistic motions
Forces applicable and masses	Affect the body's reaction during the motion
Using too few values can hamper the agent to achieve its goal	Because the character doesn't have enough degrees of freedom to walk
Several ground specifications	Can affect the creature motion

I. Results And Discussion

In this section, we present the results in evolutionary training of walking virtual creatures in the food following task, these results are obtained from the successful experiments described in the previous sections. First we begin our results representation with some results of food following task in the case of one food existing in the environment, after we present the evolving following behaviours for different food sources

A. *Following One Food Source*

In first case, the virtual creature acquires effective motions to reach the goal (food source) as efficiently as possible. The result motions of the artificial creatures that tries to catch foods; looks like walking behaviour on the ground and they are obtained after more than 1700 generations. Some of these results are presented in the Movie[2] where the artificial creatures in the latest generations walk faster than in the earliest ones. The evolutionary process in this work was able to successfully produce a stable walking movement that allow the virtual creature to move towards their goals.

Figures 5 (the first from the top) shows some actuators outputs of the left and right normalize appendages of the virtual creature during locomotion relative to a simulation period of 10 seconds. When the virtual creature moves like walking on the ground, actuators generate phases with different level values between the left and the right joint but booth joints advance with a symmetrical way; one time on the positive direction and other on the negative one and with pushing the torso which is always in contact with the ground.

Figure 5 (the second one) shows the best and the average distances calculated by the artificial creature on walking toward the food source. The distance graph look like a straight that decrease over generations, thing that means that the creature is nearing its food source, in generation 800 the distance value is close to zero.

Figure 5 (the third one) shows the speed values of locomotion realized by the evolved creatures. We note that the values of speed increase in generations, which mean that the creatures walk faster, this allows them to catch a lot of food sources as possible and therefore, they recover energy that was lost during locomotion. The latter concept as presented in Figure 5 (the last 'down' one) or the energy level is inversely decreases than the speed, except for the case where the creature grabs a source and that the amount of energy from this source will be transferred to the creature.

To better present the results obtained from the behaviour of foraging, we add another graph (see Figure 6) that shows the path of positions initiated by the creature heading towards its first food source; realised in 10 seconds of simulation.

B. *Following More than One Food Source*

Figure 7 summarizes the results obtained for the case where a lot of food sources are presented in the environment. In this graph the virtual creatures try to follow more than one-food source in order to accumulate more energy quantities then the others.

[2] Video of the evolved creatures can be viewed at: http://lesialab.net/Videos/VirtualCreature.mov

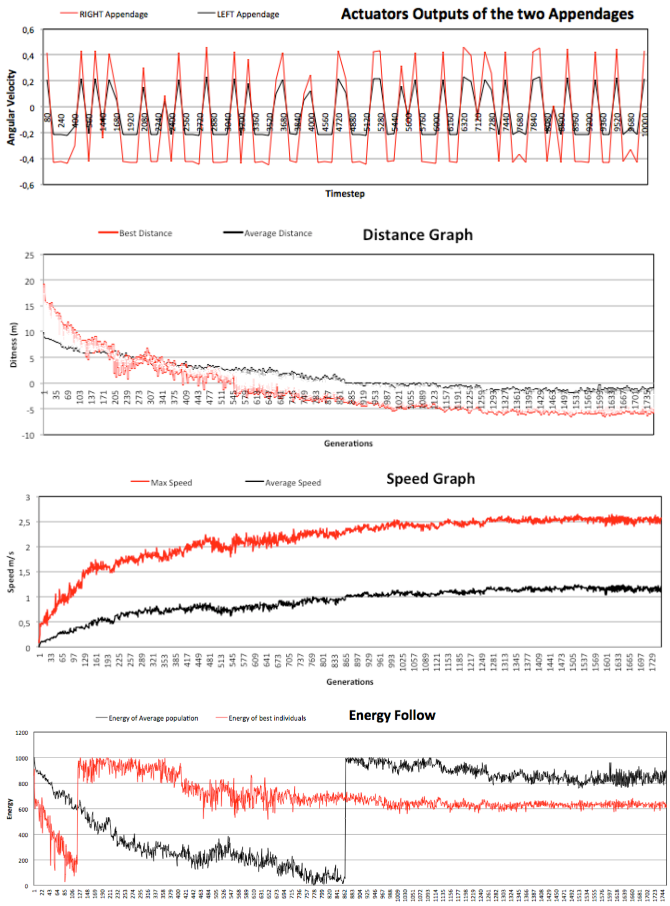

Figure 5. (First from top) The Output Actuator of the artificial creature. (Second) The Distance Graph of the artificial creature over generations. (Third) The Speed Graph of the artificial creature over generations. (Down) The Energy Follow of the artificial creature.

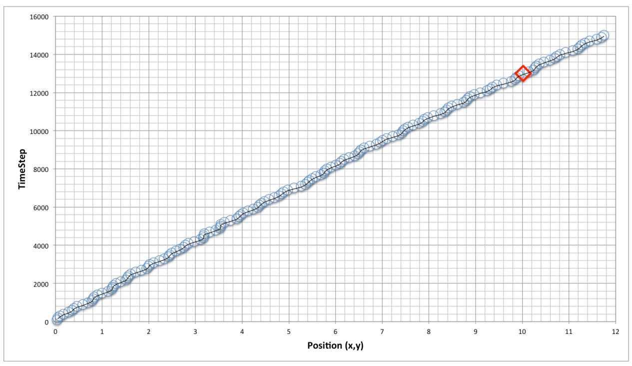

Figure 6. Series of positions of the best individual that catch its first food source.

This graph represents the average and best distances values over generations that decrease to get the first food and the second and so. From the generations 1200 the values are more stable because there is no more time to catch more food sources.

Figure 8 shows some snapshots of the virtual creatures trying to follow more then one-food source.

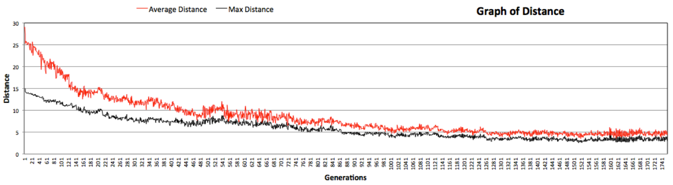

Figure 7. The Distance graph for the best and the average of the population of creatures that catch several food source.

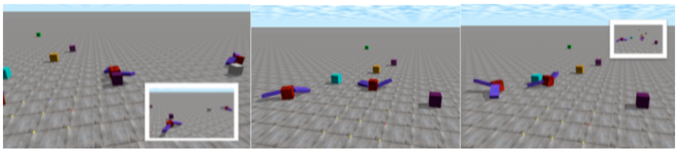

Figure 8. Traces of motions of the virtual creatures in the virtual ecosystem corresponding to food-following behaviours. The left snapshot shows two creatures that catch their first energy source. The midlle one shows the same two creatures going to follow another food source. The right one shows that one of the creatures catch its second energy source.

II. CONCLUSION AND FUTUR WORK

This article emphasizes on the first produced controllers that aim at finding the best creature's behaviour foraging for food sources. The study presented in this paper is the first step realised to the construction of a simulated virtual ecosystem.

Our future studies will show the links between all the classes of our virtual ecosystem, which have to interact inside it. The next step we aim at is the study of a predator-prey model (with co-evolution of two populations). We will study the dynamics of a predator prey system in a virtual ecosystem where the creatures have opposing goals as in [14] but in our proposed system we aim to use a single chromosome that encodes both behaviours of the predator and prey as [23].

The third part will be to study decomposers behaviours; this task can be realized by modelling bacteria behaviours, in this study we propose a model that uses the artificial chemistry to simulate the bacterium chemotaxis and to use a chromosome that evolve and emerge more complex behaviours of bacteria. Finally the Plants growth is the fourth behaviour to study which is a fascinate work to do.

ACKNOWLEDGMENT

The authors thank Nicolas Chaumont for his help to correct this paper and his detailed comments on it.

REFERENCES

[1] M. A. Bedau, "Artificial life: Organization, adaptation and complexity from the bottom up2003. Trends in Cognitive Sciences, 7(11), pp. 505 - 512.

[2] C. W. Reynolds, "Evolution of corridor following behavior in a noisy world," In SAB94: Proceedings of the third international conference on Simulation of adaptive behavior : from animals to animats 3, pp. 402 - 410, MIT Press, Cambridge, 1994.

[3] K. Sims, "Evolving virtual creatures. In SIGGRAPH 1994: Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques, pp. 15-22, ACM, New York.

[4] K. Sims, "Evolving 3d morphology and behavior by competition," in Artificial Life IV: Proceedings of the Fourth International Workshop on the Synthesis and Simulation of Living Systems, pp. 28–39, Cambridge. MIT Press.

[5] T. Miconi, "Evosphere: Evolutionary Dynamics in a Population of Fighting Virtual Creatures", in CEC2008: Proceedings of the IEEE Congres on Evolutionary Computation, pp. 3066-3073, 2008.

[6] J. E. Auerbach and J. C. Bongard, "Dynamic resolution in the coevolution of morphology and control," in Proceedings of Artificial Life XII, H. Fellermann et al., Ed., 2010, pp. 451-458.

[7] N. Chaumont, R. Egli, and C. Adami, "Evolving virtual creatures and catapults," Artificial Life, vol. 13, pp. 139-157, 2007.

[8] N. Lassabe, H. Luga, and Y. Duthen, "A new step for evolving creatures," In IEEE-ALife'07, pp. 243-251.

[9] M. L. Pilat and C. Jacob, "Evolution of vision capabilities in embodied virtual creatures," in Proceedings of the 12th Annual conference on Genetic and Evolutionary Computation, ser. GECCO '10. New York, NY, USA: ACM Press, 2010, pp. 95-102.

[10] N. Ouannes, N. Djedi, Y. Duthen and H. Luga, "Gait evolution for humanoid robot in a physically simulated environment ," in: Dimitri Plemenos, Georgios Miaouis (Eds.), SCI 374: Intelligent Computer Graphics 2011, pp.157-173. Springer Verlag-Berlin Heidelberg, 2012.

[11] K. Nakamura, I. Suzuki, M. Yamamoto, and M. Furukawa, "Acquisition of swimming behavior on artificial creature in virtual water environment," in ECAL'09 Proceedings of the 10th European conference on Advances in Artificial Life: Darwin meets von Neumann, G. Kampis, I. Karsai, and E. Szathm´ary, pp. 99-106, Eds. Heidelberg, Germany: Springer Verlag, 2011.

[12] K. Nakamura, I. Suzuki, M. Yamamoto, and M. Furukawa, "Virtual fluid environment on behavior ability for artificial creature" in ECAL 2011 Proceedings of the eleventh European Conference on the Synthesis and Simulation of Living Systems, T. Lenarets, M. Giacobini, H. Bersini, P. Bourgine, M. Dorigo, and R. Doursat, pp. 581-588.

[13] K. Iwadate, I. Suzuki, M. Yamamoto, and M. Furukawa, "Evolving amphibian behavior in complex environment," in ECAL'09 Proceedings of the 10th European conference on Advances in Artificial Life: Darwin meets von Neumann, G. Kampis, I. Karsai, and E. Szathm´ary, pp. 107-114, Eds. Heidelberg, Germany: Springer Verlag, 2011.

[14] D. Cliff, G. F. Miller, "Co-evolution of pursuit and evasion II: Simulation methods and results", In: Maes, P., Mataric, M.J., Meyer, J.-A., Pollack, J. B., Wilson, S. W. (eds.) From animals to animats, pp. 506-515, MIT Press, Cambridge, 1996.

[15] T. Francisco, G. Reis, "Evolving predator and prey behaviours with co-evolution using genetic programming and decision trees", in Proceedings of the 10th Annual conference on Genetic and Evolutionary Computation, ser GECCO'08, 2008, Atlanta, Georgia, USA, pp. 1893-1899.

[16] M. Joachimczak and B. Wrobel, "Evolving gene regulatory networks for real time control of foraging behaviours" in Proc. of the Alife XII Conference, pp. 348-355, Odense, Denmark, 2010.

[17] C. Adami and C. T. Brown, "Evolutionary learning in the 2d artificial life system 'avida' ", In Artificial Life IV, pp. 377-381, 1994.

[18] T. S. Ray, "An approach to the synthesis of life", In Artificial Life II, pp. 371- 408, 1991.

[19] M. Métivier, C. Lattaud, and J.C. Heudin, "A stress based speciation model in lifedrop", In Proceedings of the 8th Int Conf. On Artificial Life, pp. 121-126. MIT Press, 2002.

[20] J. Ventrella, "Designing emergence in animated artificial life worlds", in Virtual Worlds. Springer, 1998, pp. 143-155.

[21] S. Haykin, "Neural networks: A comprehensive foundation", In Upper Saddle River, Prentice Hall, 2nd ed., New Jersey, 1999..

[22] J. L. Elman, "Finding structure in time", In Cognitive Science 14, pp. 179-211, 1990.

[23] R. Gras, D. Devaurs, A. Wozniak & A. Aspinall, "An individual-based evolving predator-prey ecosystem simulation using a Fuzzy Cognitive Map model of behavior", Artificial Life, Vol. 15, n°4, pp. 423–463, 2009.