Muscle-Based Skeletal Bipedal Locomotion using Neural Evolution *

Artyom, Topchyan
Technical University of Munich
Munich, Germany
e-mail: topchyan@in.tum.de

Tigran, Topchyan
Yerevan State University
Yerevan, Armenia
e-mail: tigran@topsoft.am

ABSTRACT
Developments in Robotics and Computer Animation have shown that the research of Bipedal locomotion is of great importance and necessity in both fields. The approach presented in this paper utilizes a human like muscle based control system and an evolutionary technique to evolve bipedal walkers. As a result a number of controllers have been evolved, which are able to control the walker for varied distances.

Keywords
artificial intelilgence, machine learning, neural evolution, bipedal locomotion, muscle simulation.

1. INTRODUCTION
Creating human like robots or human like motion has been a long standing goal in robotics and computer simulation. The human form gives a much greater degree of utility and task freedom to robots as well as a much greater sense of belief and immersion for animation, which contain believable human like characters.

In the bipedal walking research, most researches usually start by specifying the reference trajectories for the bipedal robots to follow and are obtained by observing the walking patterns of humans, which are gathered through case studies or analysis of motion capture data[1]. There are a num- ber of drawbacks to this approach such as a requirement for strict parametrization, which leads to non-stability of that the model to different morphological changes in the work space such as external forces. A number of techniques have been developed to simplify the control design process,but it still remains an open topic [2][3].
More recent developments in the field utilize machine learning techniques, which allow the controls to be learned for a number of different possible domains[4].Most notably bio-inspired approaches have arisen, which have shown to produce good results. One such approach is the utilization of evolutionary methods with neural networks. This approach has shown to produce goods results, which produce believable human like motion invariant to a number of morphological changes [5]. This is largely due to specific way these methods optimize the network to fit the destination space. The method used in this paper starts with a simple ran- dom population of networks which are gradually grown in the process of the search into more complex networks. This leads to deep networks with complex topologies, which are suitable for inferring complex non-linear models present in robot motion. Another key aspect of the approach in this paper is the utilization of a muscle system to control the model motion.

*In proceedings of CSIT 2013

2. THE SIMULATION

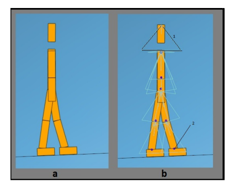

Figure 1: Figure showing the 2D Skeleton with joints hidden(a) and visible(b).Head Muscle Joint(1), Left foot Revolution joint(2)

For the sake of simplicity the simulated skeleton was designed to run in 2 dimensions, the main principles are believed to still apply to a 3D skeleton. This much simplifies the simulation, but doesn't change the essence of the learning method. A simplified box contour is used for the bodies in order to achieve simpler collision detection, but more complicated skeletons are also considered. The masses, sizes of body parts as well as imposed limits on the joint angle ranges are based on various published human measurements [6].

2.1 Skeleton Model
The developed Skeleton has 12 Body Parts, which are connected with 11 Revolution Joints with 2 DoFs(Degree of Freedom), and 11 specifically developed muscle joints(see Fig 1). This could be in essence viewed as a number of soft-springs, which are controlled by a PID control law.

2.2 Muscle joint
The Muscle joint was developed to try and mimic human muscle behaviors. The idea behind is that unlike more standard joints used in Robotics, where Torques are directly applied to Joints in order to achieve specific target positions/angles.The Muscle joint flexes or relaxes to achieve certain angles. The Muscle joint also adds a softness to the joints and as a result the movements of the Skeleton.

The rationale behind using this specific joint comes from observations on the subject of how a Human being actually perceives Motion. A specific motion is achieved by tensing and relaxing Muscles at a specific frequency depending on how fast a Human being wants to achieve a certain position with a certain body part. It is true that the end effectors are Torques which are generated in response to the Muscle tension,but when learning to walk or doing something humans do not perceive the process as a sequence of torques to be applied, but rather by the muscle tension. Neural Evolution is a biological bases technique for learning and closely resembles the evolutionary process of that of a human being, it is argued that by working through such a medium as opposed to direct Torque manipulation brings about a more Human like learning process and as a subset a more human like mo- tion. Meaning the neural network learned to carry out the motion would work through a medium, which simplifies the input and output space.

The structure of the muscles can be seen in Figure 2

2.3 Sensory Data

The Skeleton has an access to various sensory data. Such as Body part positions, Center of Masses, Joint angles and others. The main parameters that are used as inputs for the evolutionary algorithm are flags, which indicate if the corresponding foot is in contact with the ground. The flag is set to 1 if the specific foot is in contact with the ground and -1 otherwise. The sensory data is used as the input for the neural network regression model in both the direct and indirect sense. The direct sense is the 1 and -1 input for the network and in the indirect sense of gathering data about the simulation space in order to stop the training or simulation or assess the validity of the produced movement.

3. CONTROLLER

The approach used to evolve the neural networks responsible for controlling the Skeleton is Neuroevoloution by Augmenting Topologies(NEAT) [8]. As the results of this technique a neural network with varying topologies and weighted connections is trained and optimized, which is then used to control the skeleton. This section briefly describes how the outputted network is used for control and how it functions as a brain for the skeleton.

3.1 Neural Networks

As mentioned above a neural network is used as a controller. The network is a a set of neurons,with each neuron having a weighted connection to other neurons and an activation value, which determines the scalar output of the neuron. The neurons are updated once per time step by summing incoming transition weights multiplied by the incoming activation values and feeding that value into the activation function f(x),thus determining the activation value for the next time step. NEAT uses the steepen sigmoid the activation function:

$$f(x) = \frac{1}{1 + e^{-ax}}$$

Figure 2: The parameter "a" in the sigmoid function determines how steep it is. The larger "a", the steeper it is.

As a result of using simple sigmoid neurons the network will have a complicated topology of connections to achieve different outputs depending on the time step. Unlike common gradient based loss optimization for neural networks, the main goal of NEAT is optimizing the structure of the network,breeding more complex topologies,which have a strong link to the performance of the network in general.

To control the skeleton the neural network is treated as a black box controller,with 2 inputs and 11 outputs. The 2 inputs are the flags determining if the left and right feet are touching the ground or not. The 11 inputs values are target angles which need to be achieved for a walking motion.

4. EVOLUTION

NEAT is used to evolve the network topology and also the corresponding connection weights.This section contains details on the method used to evolve walkers.

4.1 Initial population

The algorithm is initialized with a random population of networks grouped into species based on a K-Means clustering. The initial population is a collection of random topologies,which ensures topological diversity from the start. NEAT starts with a minimal population,which then is gradually complexified, this is done in order to get a minimal solution in the outcome[8].

The initial population corresponds to the population at generation 0.

4.2 Evaluation

Having initialized and already containing an initial population from the previous generation, the algorithm needs to evaluate the current genome population and assign a fitness score which will be the indication of its innovation. It will directly impact the evolution in the future generations. The Evaluation is done by allowing the individual networks to control the Skeleton for a specified duration[1].

In general the longer the simulation time the longer the training takes, but in most cases the better the result.

4.2.1 Fitness function

To measure how successful the current genome was at controlling the Skeleton some kind of objective function must be used.In evolutionary methods that function is referred to as the fitness function and it measures the fitness of the current genome expressing how successful it was at carrying out the objective. A choice of a suitable fitness is a very important task and in most cases drastically influences the outcome of the evolutionary process. In the case of bipedal walking a fitness function is not trivial to choose, and there are different suggestions on the subject of which function to use. Some suggest using minimal energy use[7], or the similarity of the achieved motion to the recorded human motion capture data, but for the case of pure bipedal locomotion a function that biases the search space towards longer,stable walks was chosen.

The fitness assigned to a specific genome is calculated by the distance in the positive direction the skeleton was able to walk under the control of that specific controller. The fitness function is expressed by this simple formula:

$$f_{fitness} = max(max(L_x, R_x), \epsilon)$$

Figure 3: Where L_x is the x coordinate of the left foot and R_x for the right foot.ϵ is a small positive number(as fitness cannot be negative)

[1] *For most of the conducted experiments the duration was set to 10000 simulation time steps which amounts to 3 minutes.*

4.2.2 Early Termination Constraints

As not all networks(genomes) are viable for the simulation, not only in the sense, that they do not lead to positive increase in distance traveled, but they might also fall out of the range of acceptable angles for the specific joint. This means, that as the network outputs are directly mapped to the target angles they might not lead to undesired morphological changes in the body. To address this issue a num- ber of early termination constraints were put in place to discard networks that are obviously not leading to positive changes in the movement. For most of the experiments three constraints were used.

1. If both flags signal that feet aren't in contact with the ground,meaning either the skeleton has fallen over and is on the ground trying to get up, or is trying to take flight.

2. If both L_x and R_x are negative meaning that no positive movement is achieved.

3. If one of the target angles is not in the limits imposed by the Muscle Joint.

Such a method is used to fully utilize the sensory capabilities that the simulation and humans have.

4.3 Complexificiation and Mutations

After a genome population has been evaluated and fitness values have been assigned, the next step is to further complexify the networks in the population and do another evaluation round. The complexificiation is achieved by random mutations applied to the current population.
Neat contains two mutation variations,structural and non-structural.

- Non-structural mutations are mutations applied to the weights of the connection between neurons.

- Structural mutations on the other hand directly influence the network topology.

NEAT has some number of key features,that set it apart from most other evolutionary based methods and allow NEAT to achieve both greater time based performance and com- plexity [8].

5. RESULTS

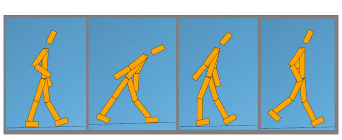

Figure 4: Walking sequence for the controller corresponding to the network in Figure 8

Experimentation has proved that the approach is viable for evolving networks,which are able to control the walker for variable distances. The evolved controllers provide surprisingly human like behavior in both achieving a walking pattern and not achieving it.

An example of a successful walking pattern is displayed in Figures 7 and 8. Because the walker is biased to walk the longest distance possible and a step length was not factored in, the walker takes long steps and this leads to a non balance state, which in the end leads to the stop of movement or the fall of the walker. This is an example of a human like behavior in trying to walk as far as possible in a more stable walking pattern. Another example of a walker is displayed in Figures 9 and 10. The walker presented here initially presents a human like walking pattern with a small constant step, but at some point a non balance point is reached the walker, striving to travel a greater distance,starts jumping forward to get as far as possible.

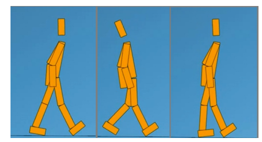

Figure 5: Walking sequence for the controller corresponding to the network in Figure 9

5.1 Networks

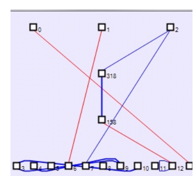

Figure 6: Network evolved after 450 generations of mutations, with an initial random population of 150 networks

The corresponding evolved network for the walker in Figure 4 is presented in Figure 6. This network was evolved after a fairly large amount of generations, specifically 450 in this case. Because more generations have passed the general topology of the network is quite complex and has connections and cyclic connections for all the inputs and outputs with hidden nodes being added in. This leads to the more efficient walker in Figure 4, which takes control of all the parts of the body and tries to walk as far as possible. In contrast the network for the walker in Figure 5 is presented in Figure 7.It is evident that this network has a far simpler topology and this is because it was evolved in just 2 generations. But the network is still able to control the walker, this is because the evolutionary technique created connection for just one foot and one arm and this leads to the walker "dragging" its body forward by creating counter balance with the arm and doing a step with just one foot. This is a rather interesting observation, that despite the simple network evolved it still illustrates how factoring a general aim to travel as far as possible into the technique and using the specific joint structure which actually allows for the "dragging" of the body. This can be viewed as a very human like behavior in some cases.

5.2 Population

A strong relation between the initial population of networks and the output topology of the networks has been noted. An example of this can be seen from Figure 8. This networks had an initial population of 466 networks and achieved complexity comparable to the network in Figure 8 in only 100

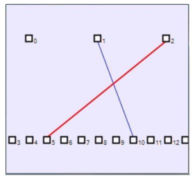

Figure 7: Network evolved after 2 generations of mutations, with an initial random population of 150 networks

mutation cycles. This can be explained by the fact, that having a higher initial population size there are more networks and thus, more species and greater probability of successful breeding within the species, which will produce healthy, successful offspring.

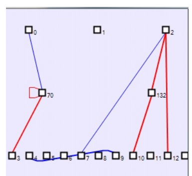

Figure 8: Network evolved after 100 generations of mutations, with an initial random population of 466 networks

6. FUTURE WORK

There are a number of techniques, which can be used to improve the performance of the evolved controllers.

6.1 Fitness function variation

It has already been discussed in previous sections,but the usage of more complex fitness functions, which have more input parameters, such as step order, step length or a better distance metric may all be used to achieve greater performance.

6.2 Evolutionary method choice

Another way is to use an alternative Evolutionary technique. One such technique is Novelty Search. Novelty search is based on NEAT, but unlike NEAT which uses fitness to judge innovation and performance Novelty Search rewards Novelty in the evolved networks, and it has been shown that for deceptive problems such as Bipedal walking, it might lead to far better results than NEAT [9].

6.3 More advanced models

Some work has already been done on a more complex model for not only walking, but other human like behavior. The skeleton presented below was developed in order to leverage a more complete and complex system.

This skeleton was created by scanning a model of a human skeleton and creating an approximate physical representation in the simulation. The skeleton also sports the full array of human muscles used for lateral movement. The approximate positions of the muscle were taken from extensive kinesiology research. In addition to the more advanced skeleton,a much more complicated sensory system has been implemented, which takes into account a vision like system for the skeleton. Considering this vastly more complex system the initial approach is being extended to also include behaviors, such as standing. Using this behavioral network in conjunction with the standard walking strategy leads to

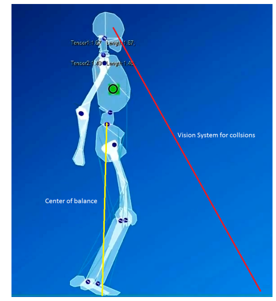

Figure 9: An advanced skeleton with center of balance projection as well as vision system for collision estimation

interesting results of much more stable walks.The system is still currently being developed, but it is evident that such a biological approach is a very interesting research field.

7. CONCLUSION

This work presents a method for creating bipedal walker controllers using neural evolution and muscle based control. The method allows to create human like motion in artificial agents using only the model itself as an input.

Despite the fact that a smooth, fluid bipedal walker was not as of yet produced, the work shows that the specific skeletal structure and the evolutionary approach have shown to be a perspective field for further research as even very simple evolved models exhibited human like bipedal motion and at the same time involved very little parametrization.

REFERENCES

[1] KangKang Yin, Kevin Loken, Michiel van de Panne *"Simple Biped Locomotion Control,"* 2007.

[2] Jerry Pratt Chee-Meng Chew Ann Torres Peter Dilworth Gill Pratt *"Virtual Model Control: An Intuitive Approach for Bipedal Locomotion, "* 2001.

[3] Jungho Lee, Jun Ho Oh *"Biped Walking Pattern Generation Using Reinforcement Learning,"* 2007.

[4] Brian Allen, Petros Faloutsos *"Complex Networks of Simple Neurons for Bipedal Locomotion,"* IROS, 2009.

[5] S. Nolfi and D. Floreano *Evolutionary robotics: The biology, in- telligence, and technology of self-organizing machines,* MIT Press, 2000

[6] *"Department of Defense, United States of America, Anthropometry of U.S. Military Personnel (DOD-HDBK-743A). Department of Defense,"* United States of America, 1991.

[7] Jerry Pratt Chee-Meng Chew Ann Torres Peter Dilworth Gill Pratt *"Virtual Model Control: An Intuitive Approach for Bipedal Locomotion,"* 2001.

[8] Kenneth O. Stanley and Risto Miikkulainen *"Evolving Neural Networks Through Augmenting Topologies,"* Evolutionary Computation 2002.

[9] Joel Lehman and Kenneth O. Stanley *" Abandoning Objectives: Evolution through the Search for Novelty Alone, "* Evolutionary Computation 2002.