WALKING LIKE A TWO YEAR OLD: A ROBOT SIMULATION

Richard Jones

MEng Computer Science and Cybernetics, siu99rnj@reading.ac.uk

ABSTRACT

This paper shows the use of a central pattern generator type controller to govern biped locomotion. Various types of locomotion are examined including walking and running. To obtain the parameters for the central pattern generator various optimisation techniques are used. Their success and convergence speeds are shown in this paper when trying to reach the desired locomotion. A model is developed that determines the outcome of the experimented parameters for the locomotion.

1. INTRODUCTION

Walking and running is a common means of locomotion for animals. It is believed that an animal's ability to navigate its environment is an important factor in a species evolution. However, the apparent ease to which human can move hides a very complicated process that is still not understood fully. The ability to move is important when looking at robots and it is the premise of this paper that some of the problems that are encountered will have solutions grounded in the biological world.

This paper describes the use of central pattern generators (CPG) to control biped locomotion. It is believed that they are used extensively in animals to create cyclic motions to which a walking locomotion is one [1]. They are also used for other cyclic motions such as chewing and swimming movements [2]. The use of CPGs in biped walkers has been studied previously in [3].

There are various other strategies that have been used to develop bipeds both in simulation and physically including standard control techniques. Central pattern generator type structures have also been used as controllers.[3] These controllers however have used central pattern generators to control the angles of the joints without any regard to the stiffnesses of the joints or the links between them. It is believed that considering these stiffnesses is entirely biologically plausible. The geometry of the human leg (in particular the addition of the knee joint) allows for differing stiffness. For example when humans land after jumping they can bend their legs so that geometrically they are not that stiff. In contrast when the heel of a foot touches the ground when walking, the leg is straight and much stiffer. Hence, this project looks at controlling the joint angles and also considers the stiffnesses within the bipedal structure.

Central pattern generators are known to exist in animals. Making simulations and robots biologically based has had some success in the past [4][5][6][7]. This biologically realistic approach was hoped to have a high success rate.

This paper presents a simulation of a simple biped model that learns to move using central pattern generators. The various optimisation techniques that enable the central pattern generator to learn the correct parameters are presented as well as the results of the simulation undertaking various walking strategies. Optimisation techniques range from simple mutation to a more complex procedure that is based on the simplex optimisation technique.

2. CENTRAL PATTERN GENERATORS

Central Pattern Generators (CPG) have been found in many creatures. They produce a rhythmic output, which is believed to control several of the rhythmic actions that the body performs [2][8][9] both conscious and unconscious. Examples of such actions include walking and chewing. Collections of CPGs have been found at the bottom of the spinal cord and have been shown to produce a rhythmic motion [1] that controls muscles which induces locomotion. Even when detached from the brain and without any input they can be forced into starting this rhythmic output. Tactile contact by the feet has been known to start the rhythmic output of the CPG. Input will affect the oscillations though. It has been noted that without any input the frequency of a locust wing beat is 50% lower than with an input. This may attribute to why creatures can tend to walk without appearing to "think". Locomotion may not be dependent on control signals from the brain but on these rhythm producing structures in conjunction with controlling signals from the brain. It is believed that there is a CPG type structure controlling each limb in the body with several additional inter-neurons used for the coordination of movement. [10].

There have been several models developed to represent central pattern generators most of which make use of reciprocal inhibition. This is the pattern of connectivity in which two neurons inhibit each other. It has been used before in many neural structures and is essential when modelling central pattern generators. The first and most simple is the half centre model presented in Fig. 1.

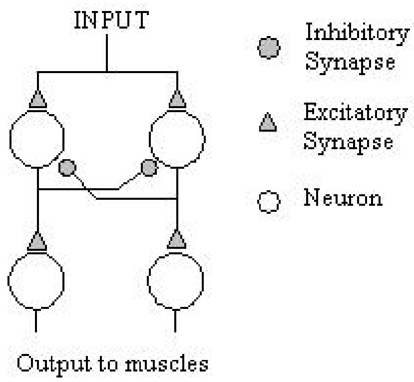

Figure 1. The half centre model representation of a CPG

1) A tonic input is received by both neurons in the network.
2) Biologically the neurons are similar but one neuron is going to be more sensitive than the other. The more sensitive neuron (neuron 1) becomes excited, fires and inhibits the neuron on the opposite side of the network (neuron 2).
3) After a period of time Neuron 1 will then fatigue and release neuron 2 from its inhibition.
4) Neuron 2 will then respond to the tonic input and become excited, fire and inhibit neuron 1.
5) After a period of time Neuron 2 will fatigue and release neuron 1 from its inhibition.
6) Neuron 1 then responds to the tonic input and the process is repeated.

Other CPG models include the pacemaker model and the network oscillator [11][12]. The pacemaker model has rhythms that are based upon the membrane properties of the cell. The rhythms are still present even when the pacemaker is isolated from the nervous system [12]. The network oscillator is based upon a network of cells which individually are much simpler than the pacemaker type. Both of these are more complicated CPGs than the half centre model but are believed still to output rhythmic signals which can be interpreted by the body to perform rhythmic actions.

Very few bipeds can walk immediately after birth. There is a process of learning. In some animals it is quicker than in others. There is an innate learning procedure that makes the central pattern generator produce the correct amplitude and frequency output for the desired movement. This learning procedure is not yet understood. In this paper conventional optimisation procedures are applied to obtain the correct parameters.

3. SOFTWARE

3.1. The model

This project seeks to simulate aspects of walking. A mathematical model was developed describing the movement of a 2D biped walker. If parameters could be found to enable the model to move to particular targets, such as time and distance, then the project would be classed as a success.

The model was of a simple biped with no knee joint. In order for the biped to prevent the trailing leg hitting the floor the leg was shortened. The leg could be thought of as the distance between the heel and the hip rather than a representation of the actual leg. These two legs were mass-less but supported a body with a mass above them. Each of the legs had a stiffness value associated with it. There was also a stiffness value between the two legs. Hence there were three stiffness values associated with the model.

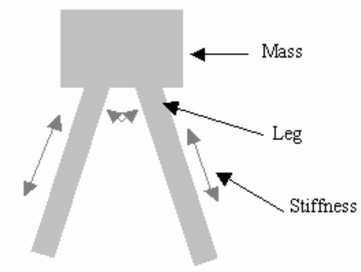

Figure 2. A diagram displaying structure of the biped model

The legs come into contact with a ground that has a frictional value of 0.8. This is the frictional value of rubber on concrete [13] so is considered similar to a shoe on a pavement. The other parameters that affect the model are listed in Table 1.

The biped model can be in one of six states.

1) Airborne (both feet in air)
2) Left foot on floor and fixed
3) Right foot on floor and fixed
4) Left foot on floor and slipping
5) Right foot on floor and slipping
6) Fallen over

Table 1. The parameters that can be changed by the model

Parameter	Description	Minimum	Typical	Maximum
Nu	Friction	-	0.8	-
M	Mass of body	-	1	-
RLL	Length of Right Leg	0.01	1	2
LLL	Length of Left Leg	0.01	1	2
K	Stiffness value associated with each leg and between legs	[0 0 0]	[100 100 100]	[10000000 10000000 10000000]
a	Amount for foot kick off	0.001	-	1
b	Angle leg moves (radians)	0.0001	-	1
c	Initial horizontal velocity	0.001	-	1
d	Size of CPG array	100	-	200

States four and five are determined by comparing the force on a leg with the angle for friction $\tan^{-1}(\eta)$

$$\frac{dVx}{dt} = -xB \quad (1)$$

$$\frac{dVy}{dt} = -yB - 9.8 + \frac{1}{m}Fl + \frac{1}{m}Fr \quad (2)$$

where:

$$Fl = k(L_l - L_{ol})\cos\alpha_l \quad (3)$$

$$Fr = k(L_r - L_{or})\cos\alpha_r \quad (4)$$

Where Lr and Ll are the lengths and Lol and Lor are rest leg lengths of the right and left legs respectively, , m is the mass, g is gravity and Kl is the leg stiffness. Equation (1) is used when the leg is on the ground to calculate whether the leg is slipping. When both legs are on the ground then the angle between them cannot be changed. This can only happen when one or both of the legs are off the ground.

3.2. Hard coded walking

To ensure that the model was advanced enough for locomotion to occur a walking CPG was hard coded. This showed the walker could walk forever if desired with only slight foot slippage. This slippage was not eliminated due to the time it took to hard code the walker and was considered insignificant. The results and parameters for this hard coded walk can be seen in section 4.

3.3. Optimisation

In order to get the model to move, various optimisation techniques were explored. The object was to optimise the leg and body parameters. The leg parameters consisted of the length and the angle between each leg. The body parameters consisted of total forward movement of the body. This forward movement is made up of two components; vertical and horizontal segments. These are the parameters that need to be optimised to achieve successful animal locomotion. The leg movements will vary depending on the locomotion that is to be optimised. For a bipedal walking type locomotion the leg movement can be considered in the following figure:

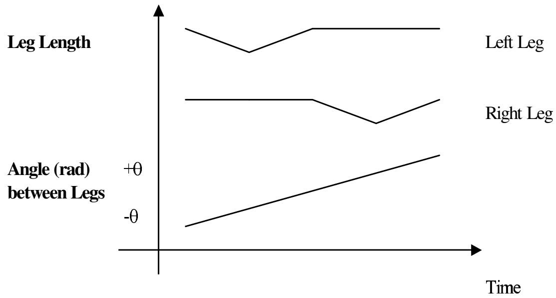

Figure 3. The leg parameters for walking locomotion

In contrast the bipedal leg movement for running is as follows:

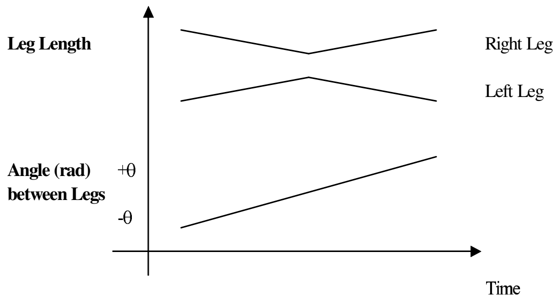

Figure 4. The leg parameters for a running locomotion

The parameters on each of the proceeding two diagrams are the leg parameters that need to be optimised. The parameters of the body movement cannot be shown diagrammatically as they are simply two scalars. All parameters are kept in there own separate arrays. The body parameters are in an array of size two. The leg parameters are held in three larger arrays holding leg lengths and leg angles for the locomotion cycle. The time through the movement cycle decides where in the array a value is taken from. That is at time equals zero at the beginning of the cycle the first element of the array is read. Half way through the cycle the element at the halfway position in the array is read.

There are four parameters that are to be optimised. These are given random value at the start. They are:
1) Amount of force the leg kicks off with.
2) The angle that the leg moves through.
3) The initial horizontal velocity of the biped.

4) How long the time period is for one cycle. That is how size of the CPGs arrays.

The optimisation procedure will optimise the above four parameters to achieve a successful locomotion.

The optimisation occurs after one run of the simulation. The “brain” section of the simulation will inform the body as to whether it is walking, slipping or fallen over. An overall success value will be given to those particular parameters. This is considered similar to how the biological optimisation occurs. This is similar to the human body and the optimisation will occur under instruction from the brain.

There are various different objectives that could be optimised. For example, the biped model could be asked to walk for several seconds. In contrast it could be asked to walk the equivalent of several metres. In addition the importance of slipping can be varied. The model could be asked to complete one of the two above objectives without falling over which might result in slipping which does not matter in which case an ice skating like gait might result. In contrast, the model could be asked to reach the objective without slipping or falling. The latter approach is taken in this case. The primary object was to get the model to walk for eight seconds with no slipping or falling.

There were various optimisation techniques explored and these are discussed in the following sections. These include simple mutation and a variant of the simplex procedure. Other techniques, such as genetic algorithms have previously been used successfully [2][3][6][7][14] but have not been explored here.

3.3.1. Simplex

This method uses an adaptation of the simplex optimisation techniques [15] that were developed and can be obtained from [16]. The standard simplex technique is not suitable for a problem such as this. The search space will consist of several local minima in addition to the several local maxima. The search space will be multi-dimensional. A three dimensional representation of such a search space is shown in Fig. 5. This varies the velocity of the walker and the maximum angle between the legs. All the other parameters are kept the same. This ensures that an easy representation can be seen. Plotting a 5-dimensional plot would be of no use.

This modified simplex procedure will help prevent the solution getting stuck in a minima.

Like the conventional simplex procedure, an objective function is needed. The objective function in this case will move the parameters to “higher ground” in the solution space. This objective function is based on four

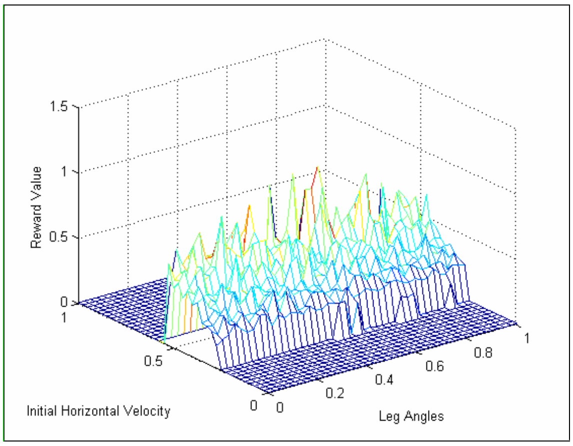

Figure 5. This is a 3D solution space of when joint angles and velocities are changed.

parameters that need optimising. The objective function is:

$$reward\_value = \frac{(\alpha_1 A + \alpha_2 B)}{(\alpha_3 C + \alpha_4 D) + 1} \quad (5)$$

A is the absolute maximum distance the biped moves. The absolute value is taken so that the robot can travel backwards as well as forwards. The maximum value is taken so that if the biped moves about but at the end of the simulation is near to where it started. Therefore the previous movement will still be taken into account

A = max(abs(distance))

B inspects the amount of contact that the biped has with the ground. That is different gaits have different percentages of foot contact compared to the total foot contact they could have. For example when a biped is running the amount of foot contact is a great deal less than for a slow walking biped. The contact of both feet is considered. Assuming that each leg behaves identically with a phase difference then the amount of contact that each foot has can be thought of as the value of B divided by 2. Fig. 6 shows the differing amounts of foot contact of each foot for the different gaits of a biped.

B = percentage_of_contact

C investigates the amount of slippage of each foot using a particular set of parameters. This desired amount of slippage is small. It is a parameter that is important and so has been included.

C = amount_of_slippage

D examines how high the hip (centre of mass) of the biped reaches during the simulation. The model has no limit on the height the biped can reach. It was not included so that this factor of the objective function

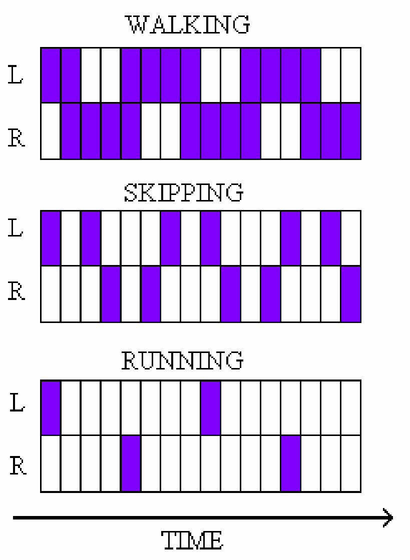

Figure 6. The differing amount of foot contact for different gaits of a biped. The coloured squares indicate foot contact

could be varied and the different results could be examined. For example if the robot wanted to walk for eight seconds and the maximum height was not an issue then the top diagram in Fig. 7 would be a possible solution. If the maximum height was an important factor then a more conventional walking approach might be found which is the lower diagram in Fig. 7

D = max(height_reached_by_body)

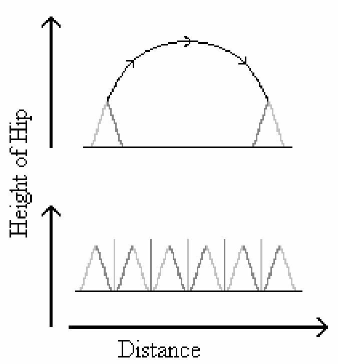

Figure 7. Two diagrams showing less conventional and conventional movement of a biped moving a certain distance

The values of $\alpha1$, $\alpha2$, $\alpha3$ and $\alpha4$ shown in (5) are varied according to the type of movement desired. For example if the amount of contact is important and the height the robot reaches is less of an issue then $\alpha1$ and $\alpha4$ will have very differing values. It should be noted that a walking and running gait will not be programmed. Desirably, the objective function will produce similar gaits to them. The final "look" of the gait is unknown and will be dependent on the learning procedure. Therefore, with different parameters the objective function could produce various novel gaits. The addition of one to the denominator is included to prevent the objective function having to contend with a divide by zero.

The initial parameters given to the simplex technique were random values. Therefore the technique started from a random point in the solution space.

The success of this method is discussed in section 4.

3.3.2. Extended simplex

Another optimisation technique that could have been explored was simulated annealing. This has been used in several optimisation problems very successfully [17] and has been developed and extended in several ways [18].

Simulated annealing explores several sections of the search space concurrently. There is also the notion of momentum to stop the solution getting trapped in local minima.

4. RESULTS

4.1. Hard coded walking

The parameters for the hard coded walk were achieved by a trial-and-error approach. This tedious and time-consuming approach demonstrated the usefulness of the optimisation techniques whose results are shown in section 4.2 following this.

Fig. 8 shows the pattern of foot contact the walker has. It can be seen to bounce slightly when the foot comes into contact with the ground. This is expected due to the spring-like model the legs have. Initially the model is in the air and falls to ground with the right foot forward. Hence the left foot makes the first step. The model then takes 4 steps (2 complete walk cycles). The model can walk indefinitely so the pattern would just repeat.

The parameters produced in table 1 showed that the model was advanced enough to have some kind of walking locomotion. It is assumed that it would also be able to produce other types of locomotion as well.

4.2. Optimisation results

As expected the simplex optimisation methods are the most successful and quickest of the techniques. Both mutation methods find results but take far longer. It appears that the complexity of the techniques has a bearing upon the time it takes for a solution to be found. The more complex the technique, the quicker optimisation occurs.

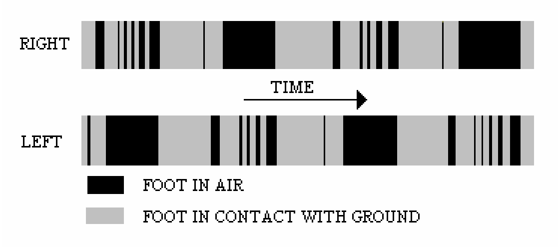

Figure 8. Foot contact for each foot when considering a walking type gait

5. CONCLUSION

This paper demonstrates the calculation of central pattern generator parameters that enable various types of locomotion.

In humans, central pattern generators are known to control such rhythmic movement and hence some sort of optimisation technique must be used within the body. There is certainly a trial and error approach initially with human children. This project has shown several optimisation techniques that calculate these parameters which conform to the

This project has shown a new way of obtaining parameters for a biped walker. It is based on the learning of a biological biped. It is hoped that the work can be extended and the technique included into a real walking biped. Disadvantages of such an approach are that the model simulation has to be highly advanced. In the time taken to develop this model more conventional control methods could be used to control the walker.

These techniques, as they are based on biological realities, can be developed and will in the end be a viable alternative to the abstract control strategies that are used in conventional simulations and robotics.

Acknowledgements: I would like to thank William Harwin for his constant help during the project and the aid he gave me in developing the initial model. Without his help and guidance the project would have had limited success due to the time constraints surrounding it.

6. REFERENCES

[1] L. M. Jordon, "Imitation of Locomotion in Mammals", Annals of the New York Academy of Science 860, pp 83-93, 1994

[2] A.J.Ijspeert J. Kodjabachian, "Evolution and Development of a Central Pattern Generator for the Swimming of a Lamprey", Artificial Life, 1999,

[3] R. Reeve and J. Hallam, "Control of Walking by Central Pattern Generators", Proceedings of Intelligent Autonomous System, 1995

[4] J. Hoff and G.A. Berkey, ,"A Cerebella Approach to Adaptive Locomotion in Legged Robots", IEEE International Symposium on Computational Intelligence in Robotics and Automation, 1997

[5] A Billard M.J.Mataric, "A Biologically Inspired Robotic Model for Learning by Imitation", Proceedings of Autonomous Agents, 2000, pp373-380

[6] K.Simms, "Evolving Virtual Creatures", Computer Graphics Annual Conference Series, pp15-22, 1994

[7] K. Simms, "Evolving 3D Morphology and Behaviour by Competition", Artificial Life IV Proceedings, MIT Press, pp28-39, 1994

[8] Y. Asia, T Nomura, S. Sate, "A Coupled Oscillator Model of Ordered and Disordered Inter-Limb Coordination", 4th Neural Coding Workshop, Plymouth, 1994

[9] R. Calabrese, "Half Centre Oscillator Underlying Rhythmic Movement", The Handbook of Brain Theory and Neural Networks 1st edition MIT Press pp444-447, 1995

[10] S. Grillner, "Control of Locomotion in Bipeds, Tetrapods and Fish", The Handbook of Physiology Section I: The Nervous System Vol. II, Oxford University Press, pp1179-1236

[11] "Principle of Neural Science", Edited by E.R.Kendel, J.H.Schwartz, T.M.Jessel, Prentice Hall, 2001, FOLIO 612.8-PRI

[12] J.Dean and H Cruse, "Motor Pattern Generation", The Handbook of Brain Theory and Neural Networks 1st edition MIT Press pp600-605, 1995

[13] "The Forces of Friction" by S. Wirt, 2003 found at http://www.sciencejoywagon.com/physicszone/lesson/02forces/friction/friction.htm

[14] M.A.Lewis, A.H.Fagg, G.A.Bekey "Genetic Algorithms for Gait Synthesis in a Hexapod Robot", published in Zheng Recent Trends in Mobile Robotics, World Scientific, pp317-331, 1994

[15] "Linear Programming", John Blake, 2002 found at http://oregonstate.edu/instruct/ch490/lessons/lesson10.htm

[16] Matlab File Exchange Optimisation section found at http://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectId=102&objectType=file, Z. Bajzer, 2003

[17] S.Kirkpatrick, C.D.Gelatt Jr, M.P.Vecchi, "Optimisation by Simulated Annealing", Science Volume 2000, pp671-680

[18] R.J.Koshel, "Enhancement of the downhill simplex method of optimisation", International Optical Design Conference, 2002