OPTIMISATION OF SWING-UP CONTROL PARAMETERS FOR A ROBOT GYMNAST USING THE BEES ALGORITHM

H G Kamil[1], E E Eldukhri[1], M S Packianather[1]

{KamilHG, EldukhriEE, PackianatherMS}@cardiff.ac.uk
[1]Institute of Mechanical and Manufacturing Engineering, School of Engineering,Cardiff University,Queens Buildings, The Parade, Cardiff CF24 3AA,UK

Abstract: This paper focuses on using the Bees Algorithm for optimising the parameters of the swing-up control for a robot gymnast (Robogymnast) attached to a freely rotating high bar mounted on ball bearings. Robogymnast, which mimics a human acrobat, consists of three links and three joints. Its motion is manipulated by two DC motors mounted at the shoulders and hip joints. The freely rotating high bar represents the third joint to which link 1 (hands and arms as a single rigid part) is firmly attached. Although, this triple pendulum-like structure is difficult to balance at upright posture, its unpowered joint is advantageous during the swing-up phase. The ultimate challenge was to smoothly swing up Robogymnast from the downward (stable) position to the upright (unstable) configuration by finding optimum values of the parameters that regulate the amplitudes and frequencies of the sinusoidal signals applied to the two DC motors. The Bees Algorithm was used as an optimization technique to achieve this. From the randomly obtained set of parameter values, three were selected to simulate the behavior of Robogymnast during the swing-up phase. The results showed successful swing-up of Robogymnast.

Keywords: robot gymnast, inverted pendulum, swing-up control, Bees Algorithm.

1. INTRODUCTION

In this paper, the swing-up problem of a nonlinear, three-link robot gymnast (Robogymnat) is discussed. With the first hinge being passive and the rest are active, Robogymnast is classified as an under-actuated robot (Liu and Yamaura, 2001). Different control strategies were used to study the swing-up of such an inverted pendulum-like mechanism. Also the problem of stabilising inverted pendulums with passive joints at the upright position was extensively investigated (Arai and Tachi, 1991, Furuta et al., 1991, Medrano-Cerda et al., 1995, Saito et al., 1993, Spong, 1995) . Spong (Spong, 1995) proposed an approach based on the partial feedback linearization(Isidori, 1989). This approach was very sensitive to the gain values of the outer loop and the switching times. Eltohamy and Kuo (Eltohamy and Kuo, 1998) used a numerical optimization algorithm for the controller design which included a globally convergent numerical technique. Their designed controller was produced as an optimization problem which accounted for the physical boundaries, stability condition of the system and the infinite dimensional nonlinear difference constraints. Medrano-Cerda et al., (Medrano-Cerda et al., 1995) designed a robust computer control systems for balancing and position control of double and triple inverted

pendulums. The controllers were based on linearised models of the pendulums and included integral actions and optimal state feedback implemented via functional observers. A novel optimization technique called the Bees Algorithm (Pham et al., 2006) provided a new approach for controller tuning. Pham et al., (Pham et al., 2009) used it to regulate the parameters of a fuzzy logic controller designed to balance an under-actuated two-link acrobatic robot (Spong, 1995). They first developed a Linear Quadratic Regulator (LQR) to obtain the scaling gains needed to design the fuzzy logic controller. Eldukhri and Pham (Eldukhri and Pham, 2010) studied the problem of autonomously swinging up a three-link robot with two active (actuated) joints and one passive (unpowered) joint. Their proposed approach depended on manipulating the frequency and amplitudes of the sinusoidal functions applied to the two motors driving the second and third link.

In this paper, the Bees Algorithm (Pham et al., 2006) is used to tune the parameters of the swing up control developed by Eldukhri and Pham (Eldukhri and Pham, 2010). The remainder of the paper is organized as follows. Section 2 describes Robogymnast system and its mathematical model. In section 3 the swing-up control problem is investigated. Section 4 introduces the Bees Algorithm and how it was used for tuning the swing-up control parameters. Section 5 discusses the results. Conclusions and further work are given in section 6.

2. SYSTEM DESCRIPTION AND MODELLING

Fig.1 depicts the hardware components of Robogymnast (Eldukhri and Pham, 2010). The structure of Robogymnast is modelled as a human gymnast swinging on a freely rotating high bar with his hands firmly fixed to the bar. Link 1 represents the arms without the elbow and wrist joints. The head, neck and torso are combined in link 2 as a rigid single body. Link 3 represents the legs without the knee and ankle joints. Joint 1 consists of a steel shaft mounted on ball bearings. At one end of the shaft a potentiometer is mounted to measure the angle of link 1. Each of joints 2 and 3 consists of two parts. The first part comprises a DC motor/gearbox combination with its output shaft coupled to respective link. The second part includes the sensor (potentiometer) which measures the relative angles between adjacent links. The potentiometer is attached to a short steel shaft mounted on both sides on ball bearings. The interfacing circuit between the computer and the robot comprises amplifiers and first-order filters. These filters reduce effects introduced by sampling the system's outputs and smooth the control signals sent to the power amplifiers/motor drive units (Medrano-Cerda et al., 1995). Robogymnast is controlled by a PC equipped with appropriated AD/DA converters. C++ programmes are used to transmit the input/output commands between the PC and Robogymnast.

2.1 MATHMETICAL MODEL OF THE ROBOGYMNAST

The equations of motion for Robogymnast represented by the schematic diagram illustrated in Fig.2 was derived using Lagrange equations (Eldukhri and Pham, 2010,

Eltohamy and Kuo, 1998, Furuta et al., 1984, Medrano-Cerda et al., 1995). By using Matlab®/toolboxes and additional M-files written by the authors, the linearised continuous-time, state-space model of Robogymnast was obtained as (Eldukhri and Pham, 2010):

$$\dot{x} = Ax + Bu \qquad (1)$$

$$y = Cx \qquad (2)$$

where

$$A=\begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ -36.42 & -0.35 & 0.21 & -0.20 & 88.38 & 9.17 \\ 13.10 & -22.06 & -2.23 & 0.20 & -168.29 & 7.70 \\ 2.14 & -1.50 & -5.68 & 0.02 & 7.69 & -201.45 \end{bmatrix} , B=\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ -15.19 & -0.74 \\ 28.92 & -0.62 \\ -1.32 & 16.21 \end{bmatrix}$$

$$C=\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \quad x = \begin{bmatrix} \theta_1 \\ \theta_2 - \theta_1 \\ \theta_3 - \theta_2 \\ \dot{\theta}_1 \\ \dot{\theta}_2 - \dot{\theta}_1 \\ \dot{\theta}_3 - \dot{\theta}_2 \end{bmatrix} , \quad u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} , \quad y =$$

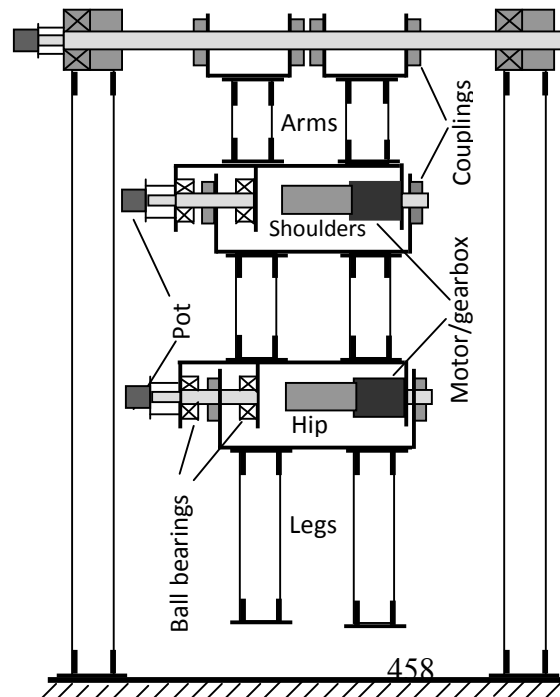$$\begin{bmatrix} \theta_1 \\ \theta_2 - \theta_1 \\ \theta_3 - \theta_2 \end{bmatrix}$$

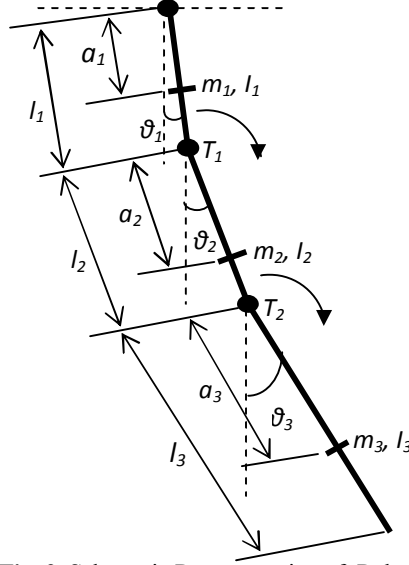**Fig. 1.** Robot Gymnast (Robogymnast)

Fig. 2. Schematic Representation of Robogymnast

3. SWINGING-UP CONTROL PROBLEM

The challenge posed by Robogymnast is how to make it swing up from the downward position (stable equilibrium posture) to upward attitude (unstable configuration). The swing-up must be achieved in a reasonable operation time without the risk of damaging any of the structure components. This requires determining of suitable input control signals to the motors located at joints 2 and 3 to obtain satisfactorily smooth sequence of oscillations. Eldukhri and Pham (Eldukhri and Pham, 2010) proposed a solution to this problem by manipulating the frequency and amplitudes of the sinusoidal input control signals applied to the two DC motors driving links 1 and 2. This enabled the two motors to pump energy into the system and consequently swing Robogymnast until it eventually flipped past the upright position. This was in effective achieved by causing the value of the first angle ($q_1 = \theta_1$) moves from the initial status ($q_1 = 0$) to the upright position, i.e. $q_1 = \pi$ (or $q_1 = -\pi$, depending on the direction of movement). The equations of the input control signals (Eldukhri and Pham, 2010) are given as:

$$u_1 = A_1 \alpha \sin(\emptyset_1) \tag{3}$$

$$u_2 = A_2 \alpha \sin(\emptyset_2) \tag{4}$$

Where A_1 and A_2 are constants and $\emptyset_1$ and $\emptyset_2$ are dependent on δ. During each sinusoidal cycle (multiple of sampling intervals T_s depending on the value of δ), $\emptyset_1$ and $\emptyset_2$ were varied between 0 and 2π with a step increment of η/δ applied during each sampling interval. η is constant and proportional to 2π. At the end of each duty cycle ($\emptyset_1, \emptyset_2 = 2\pi$) α, δ were increased by $\Delta\alpha$ and $\Delta\delta$ respectively. $A_1, A_2,$ α, η and δ were initially set at 3, 2.5, 1, 0.3142 and 1 respectively (Eldukhri and Pham, 2010). By means of exploitation of MATLAB/toolboxes, the discrete time model was obtained by discretising the linearised

459

continuous-time model at a sampling time of 0.025s. The discrete-time model was used to simulate the dynamic behaviour of Robogymnast during the swing up phase. In Eldukhri and Pham's work, the periodic (at every sampling time T_s) increment in δ ($\Delta\delta$) was obtained manually (through trial and error process).

In this paper, the Bees Algorithm was used as an optimization technique to automatically tune the values of the increment in α ($\Delta\alpha$) and the increment in δ ($\Delta\delta$) to obtain acceptably smooth swinging of Robogymnast.

4. The BEES ALGORITHM

The Bees Algorithm is a population based search algorithm that simulates the food foraging behaviour of honeybees to find the optimal solution (Kennedy et al., 2001). The Algorithm emulates the harvesting process of the natural bees by doing a local search, till an acceptable result is found, or a pre-defined number of iterations has been reached (Otri, 2011).

The Bees Algorithm starts with scout bees being placed arbitrarily on the search space. The main steps of the Algorithm are (Pham et al., 2006):
1- Initialize population with random solutions.
2- Evaluate fitness of the population.
3- While (stopping criterion not met) //Forming new population.
4- Select sites for neighborhood search.
5- Recruit bees for selected sites (more bees for best e sites) and evaluate fatnesses.
7- Select the fittest bee from each patch.
8- Assign remaining bees to search randomly and evaluate their fitnesses.
9- End While.

The Bees Algorithm parameters are (Pham et al., 2009): number of scout bees (n), number of sites selected out of n visited sites (m), number of best sites out of m selected sites (e), number of bees recruited for best e sites (nep), number of bees recruited for the other (m-e) selected sites (nsp), initial size of patches (ngh) which includes site and its neighborhood and stopping criterion. The use of the Algorithm to optimize the increment $\Delta\delta$ will be discussed in the following section.

4.1   TUNING THE SWING-UP CONTROL PARAMETERS BY USING   THE BEES ALGORITHM

The Bees Algorithm was used to investigate the optimum values of the increments in α ($\Delta\alpha$) and δ ($\Delta\delta$) that will enable smooth swing up of the Robogymnast model described in equations 1 and 2 in a reasonable time. This is achieved by manipulating independently the amplitudes and the frequencies of the control signals given in equations 3 and 4. The parameter values of the Bees Algorithm were set as in table 1.The number of scouts (range of $\Delta\alpha$ and $\Delta\delta$) was selected to be 20 (equivalent to 20 randomly selected values of $\Delta\alpha$ and

$\Delta\delta$) satisfying, respectively, the conditions $0.1 < \Delta\alpha < 0.7$ and $2 < \Delta\delta < 7$. The other parameters were selected based on internally imposed conditions limiting the error boundary in the upright angle θ_1. Using these conditions, the Bees Algorithm starts to compute the fitness (i.e. the swing-up angle of Robogymnast θ_1 reaching approximately -180^0).

Table 1. The parameter values of the Bees Algorithm.

The parameters of the Bees Algorithm	Values	Description
Number of scout bees(n)	20	Number of randomly chosen values of $\Delta\alpha$ and $\Delta\delta$ from the solution space (ranges of $\Delta\alpha$ and $\Delta\delta$).
Number of recruited bees around elite selected patches(nep)	15	Number of bees (foragers) recruited to search new values of $\Delta\alpha$ and $\Delta\delta$ which are placed within squares of sides ngh centred on the elites of $\Delta\alpha$ and $\Delta\delta$.
Number of recruited bees around best selected patches(nsp)	10	Number of bees (foragers) recruited to search new values of $\Delta\alpha$ and $\Delta\delta$ which are placed within a square of side ngh centred on the (m-e) locations of $\Delta\alpha$ and $\Delta\delta$.
Patch radius for neighbourhood search(ngh)	0.001	The boundary of neighbourhood search for new values of $\Delta\alpha$ and $\Delta\delta$ by recruited bees.

The randomly selected values of $\Delta\alpha$ and $\Delta\delta$ vary each time the Bees Algorithm code is re-executed. Table 2 shows for each selected $\Delta\alpha$ and $\Delta\delta$, how long it takes to swing up Robogymnast to the upright position ($\theta_1 = -180^0$).

Table 2. Starting with n scout bees to find the fitness of the sites.

$\Delta\alpha$	$\Delta\delta$	Angle of Robogymnast Position (θ_1 Deg.)	Duration time(sec) to reach the upright position
0.3632	5.2787	-180.0166	162.1000
0.5459	3.3925	-180.0405	172.3250
0.3353	4.7344	-180.1167	173.7000
0.2662	6.7858	-179.8697	170.1500
0.5169	4.1088	-179.8671	152.6500
0.2903	6.5787	-179.8615	165.0500
0.5072	5.1618	-179.9775	132.2500
0.6701	5.9610	-180.0446	122.9000
0.5593	6.2456	-179.8404	128.0500
0.5236	2.7881	-180.0658	209.1000
0.1277	4.4269	-179.9333	361.6500
0.5941	2.7094	-179.9327	203.7500
0.3289	2.1786	-179.9220	273.8500
0.1583	6.0014	-179.8813	294.6000
0.1207	6.7975	-179.8803	433.5000
0.2027	6.8244	-179.8724	244.3750
0.5771	6.6700	-179.8548	108.4500
0.5546	2.4877	-179.8228	233.2750

461

0.4933	6.7875	-179.8224	110.4000
0.1191	6.8530	-179.8109	437.1250

The Flowchart in Fig. 3 explains the steps of using the Bees Algorithm to optimise the parameters ($\Delta\alpha$ and $\Delta\delta$) for attaining reasonably smooth swing-up control of Robogymnast.

5. RESULTS DISCUSSION

The results obtained in Table 2 were used to further tune $\Delta\alpha$ and $\Delta\delta$ by the Bees Algorithm to achieve an optimum fitness ($\theta_1 = -180^0$) with small margins of error (<0.02) at which the Algorithm will stop searching. The fine-tuned results are shown in Table 3. To simulate the behaviour of Robogymnast during the swing-up phase, three values of $\Delta\alpha$ and $\Delta\delta$ were selected from Table 3. At each sampling interval, the control input signals u_1 and u_2 described in equations (3) and (4) were recalculated and applied to the discrete-time model of Robogymnast obtained from equations (1) and (2). As shown in Fig. 4, the time taken to reach the upright position is relatively short which may cause damage to the motor/gearbox structures. At $\Delta\alpha$ equal to 0.1191 and $\Delta\delta$ equal to 6.8530, the system's operation was very smooth. However, it reached the upright position in a very long time (437.1250 seconds) as shown in Fig. 5. Compared to Eldukhri's and Pham's work (Eldukhri and Pham, 2010), $\Delta\alpha$ and $\Delta\delta$ equal to 0.3621 and 5.2779 respectively gave a satisfactory response in a reasonable duration (162.1 seconds) as illustrated in Fig. 6.

Table 3.   Results after tuning by the Bees Algorithm.

$\Delta\alpha$	$\Delta\delta$	Angle of Robogymnast Position (θ_1 Deg.)	Duration time(sec) to reach the upright position
0.3621	5.2779	-180.0004	162.1000
0.5436	3.3931	-180.0003	172.3500
0.3317	4.7343	-180.0002	173.6750
0.2687	6.7875	-179.9847	170.1250
0.5235	4.1070	-179.9994	152.5750
0.2909	6.5795	-179.9999	165.1250
0.5088	5.1620	-179.9995	132.2500
0.6715	5.9633	-180.0004	122.9500
0.4026	2.6161	-179.9994	244.3500
0.5236	2.7881	-180.0658	209.1000
0.1277	4.4269	-179.9333	361.6500
0.5941	2.7094	-179.9327	203.7500
0.3289	2.1786	-179.9220	273.8500
0.1583	6.0014	-179.8813	294.6000
0.1207	6.7975	-179.8803	433.5000

Table 3.   Results after tuning by the Bees Algorithm. (Cont.)

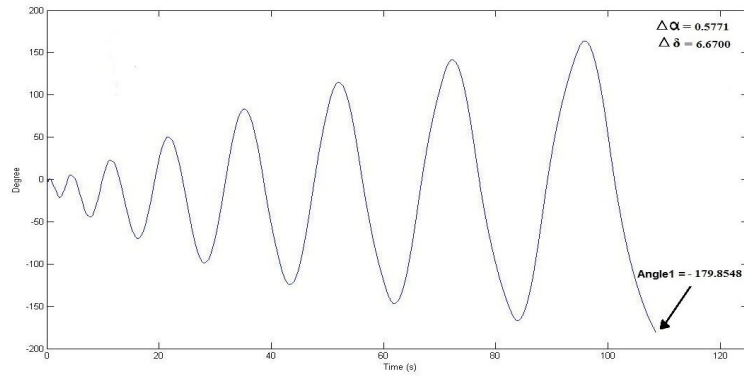| 0.2027	6.8244	-179.8724	244.3750
0.5771	6.6700	-179.8548	108.4500
0.5546	2.4877	-179.8228	233.2750
0.4933	6.7875	-179.8224	110.4000
0.1191	6.8530	-179.8109	437.1250

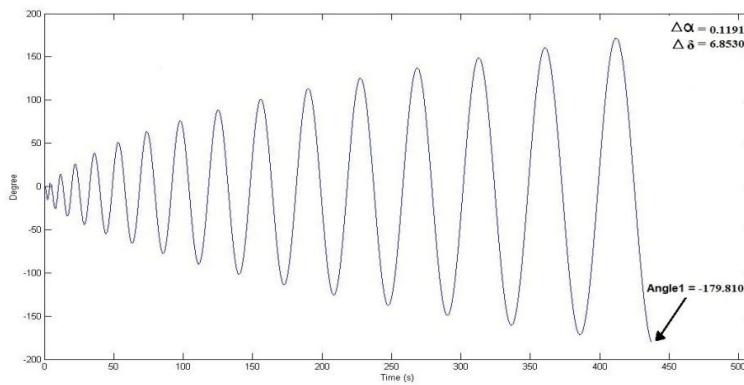**Fig. 4.** Simulated angular position θ_1 at $\Delta\alpha$=0.5771 and $\Delta\delta$= 6.6700

Fig. 5. Simulated angular position θ_1 at $\Delta\alpha$=0.1191 and $\Delta\delta$= 6.8530
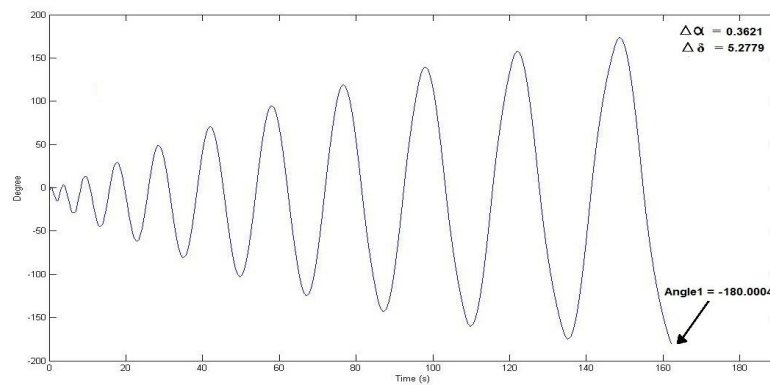

Fig. 6. Simulated angular position θ_1 at $\Delta\alpha$=0.3621 and $\Delta\delta$=5.2779

463

Start

Set the values of n=20, nep=15, nsp=10, ngh=0.001, $\Delta\alpha_{(min)}$=0.1, $\Delta\alpha_{(max)}$=0.7, $\Delta\delta_{(min)}$=2, $\Delta\delta_{(max)}$=7

Randomly choose 20 values (number of scout bees (n)) from the solution space of $\Delta\alpha$ and $\Delta\delta$

Evaluate the fitness (θ_1) of each bee where it landed at each value of $\Delta\alpha$ and $\Delta\delta$ by running the program which is used to simulate the dynamic behaviour of Robogymnast during the swing-up phase

Select the fittest (θ_1).
(m) locations are selected from the solution space. The selection depends on the minimum error margin and reasonable swing-up duration time.

Find the top elite (e) sites which have a higher fitness from the (m) selected

Assign the (n–m) remaining bees to random search

If the error margin of the (m) selected locations is <= 0.02

Yes

Solution

No

Recruit **nep** bees (foragers) to search for the higher fitness (θ_1) from the elite sites placed within a square of side **ngh** centred on the fittest elite from the previous iteration. If a forager lands on a position of higher fitness then an exchange will happen, else exploration will continue.

Recruit **nsp** bees to search for the higher fitness (θ_1) from the (m-e) sites of the best selected locations each placed within a square of side **ngh** centred on the fittest (m-e) sites from the previous iteration. If a forager lands on a position of higher fitness then an exchange will happen, else exploration will continue.

Randomly choose (n-m) values from the solution space of $\Delta\alpha$ and $\Delta\delta$

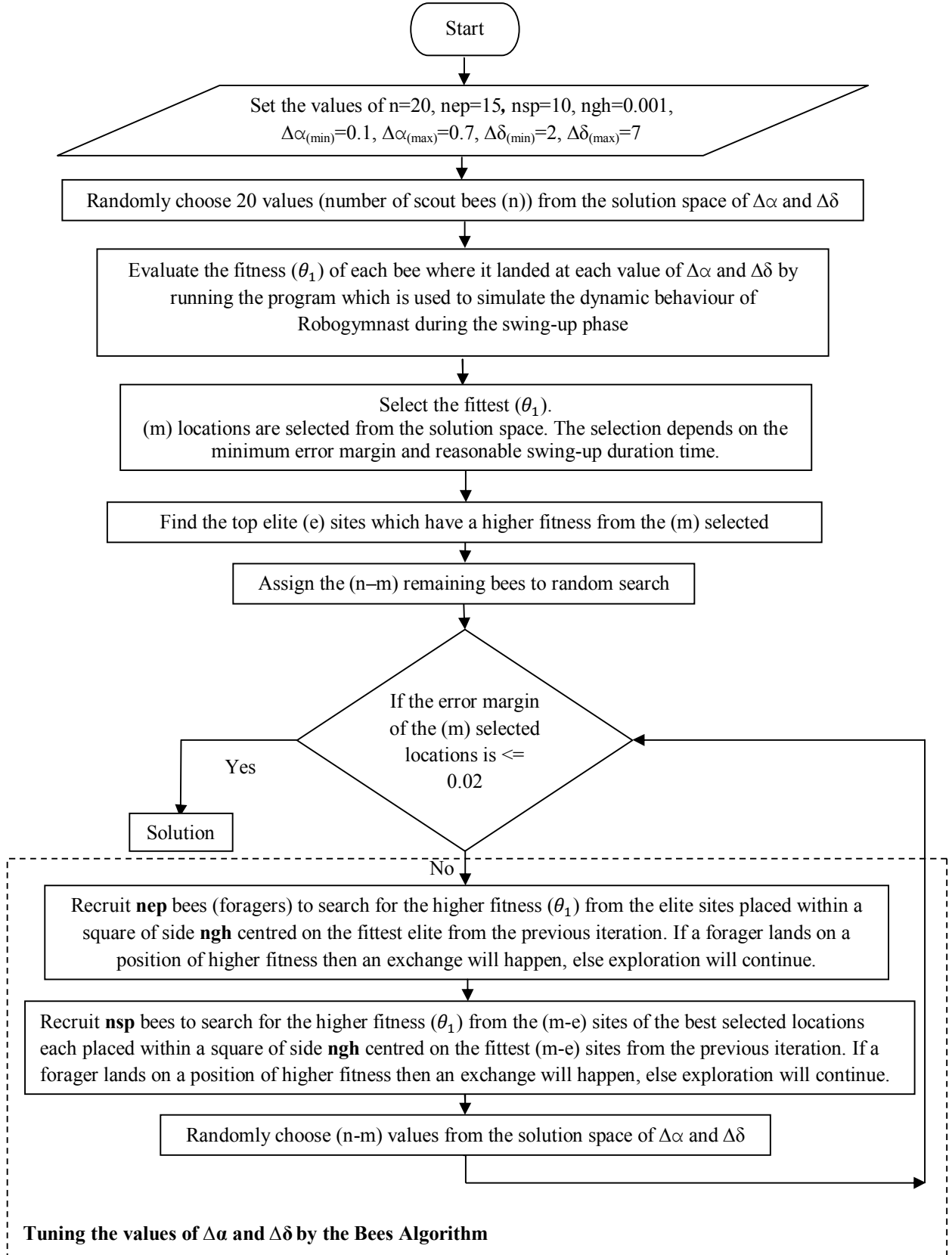**Tuning the values of $\Delta\alpha$ and $\Delta\delta$ by the Bees Algorithm**

Fig. 3. Flowchart of the swing-up control parameters optimization using the Bees Algorithm

6. CONCLUSION AND FURTHER WORK

The paper focused on using a newly developed optimization technique (Bees Algorithm) to tune the parameters of a swing-up control of a robogymnast (Robogymnast) developed by Eldukhri and Pham (Eldukhri and Pham, 2010). Unlike the previous approach, the Bees Algorithm allowed a flexible and random selection of the parameter affecting the amplitudes and frequencies of the sinusoidal signals applied to the two DC motors driving Robogymnast. The simulations results were satisfactory. Further work includes using the Algorithm to study different scenarios for tuning multiple parameters affecting each of the amplitudes and frequencies independently. It is also envisaged to implement the findings on the real the system.

REFRENCES

ARAI, H. & TACHI, S. (1991) Position control of manipulator with passive joints using dynamic coupling. *IEEE Transactions on Robotics and Automation*, 7, 528-534.

ELDUKHRI, E. E. & PHAM, D. T. (2010) Autonomous swing-up control of a three-link robot gymnast. *Proceedings of the Institution of Mechanical Engineers. Part I: Journal of Systems and Control Engineering,* 224, 825-833.

ELTOHAMY, K. G. & KUO, C. Y. (1998) Nonlinear optimal control of a triple link inverted pendulum with single control input. *International Journal of Control,* 69, 239-256.

FURUTA, K., OCHIAI, T. & ONO, N. (1984) Attitude control of a triple inverted pendulum. International Journal of Control, 39, 1351-1365.

FURUTA, K., YAMAKITA, M. & KOBAYASHI, S. (1991) Swing up control of inverted pendulum. *IECON Proceedings (Industrial Electronics Conference).*

ISIDORI, A. (1989) Nonlinear control systems (second ed.), *Berlin, Springer-Verlag.*

KENNEDY, J., EBERHART, R. & SHI, Y. (2001) Swarm intelligence. *Morgan Kaufmann Academic Press.*

LIU, D. & YAMAURA, H. (2001) Giant swing motion control of 3-link gymnastic robot with friction around an underactuated joint. *Journal of System Design and Dynamics*, 5, 925-936.

MEDRANO-CERDA, G. A., ELDUKHRI, E. E. & CETIN, M. (1995) Balancing and attitude control of double and triple inverted pendulums. *Transactions of the Institute of Measurement and Control*, 17, 143-154.

OTRI, S. (2011) Improving the Bees Algorithm for complex optimization problems. *Manufacturing Engineering Centre. UK Cardiff, Cardiff University.*

PHAM, D. T., DARWISH, A. H. & ELDUKHRI, E. E. (2009) Optimisation of a fuzzy logic controller using the Bees Algorithm. *International Journal of Computer Aided Engineering and Technology*, 1, 250-264.

PHAM, D. T., GHANBARZADEH, A., KOC, E., OTRI, S., RAHIM, S. & ZAIDI, M. (2006) The Bees Algoritm – a novel tool for complex optimisation problems. IN PHAM, D. T., ELDUKHRI, E. E. & SOROKA, A. J. (Eds.) *Proceedings of 2nd Virtual International Conference on Intelligent Production Machines and Systems (IPROMS2006)*. Cardiff, UK, Elsevier.

SAITO, F., FUKUDA, T. & ARAI, F. (1993) Swing and locomotion control for two-link brachiation robot. *Proceedings - IEEE International Conference on Robotics and Automation*.

SPONG, M. W. (1995) Swing up control problem for the acrobot. *IEEE Control Systems Magazine*, 15, 49-55.