MULTI-OBJECTIVE OPTIMIZATION OF SNAKE ROBOT IN SERPENTINE LOCOMOTION

Marwan Badran, Md Raisuddin Khan*,
Siti Fauziah Toha and Zulkifli Zainal Abidin

*Department of Mechatronics Engineering,
International Islamic University Malaysia,
Jalan Gombak, 53100 Kuala Lumpur, Malaysia*

Corresponding author: raisuddin@iium.edu.my

ABSTRACT: This paper presents multi-objective optimization for a snake robot with serpentine locomotion. Genetic algorithm (GA) is used to achieve two objectives: minimizing the total *travelling time* and minimizing the total *energy consumption*. The effect of initial values of *winding angle* and *acceleration* on *energy consumption* and *average speed* is depicted. The simulation results show a periodic pattern of the joint torques when the robot maintains a serpenoid curve during travel. Moreover, a Pareto-optimal front was generated for optimal solutions of both of the objectives, while the weighted sum method was used for selecting the best solution. Finally, the simulation results were verified experimentally on an eight-link snake robot considering the limitations of the servomotors used in the experiment. The experimental results with the *winding angle of* 30° was found as the optimum *winding angle* that can achieve both objectives of minimizing the *energy consumption* and the *travelling time*.

ABSTRAK: Kajian ini berkenaan pelbagai-objektif optimum bagi robot ular dengan gerakan serpentin. Algoritma genetik (GA) diguna bagi mencapai dua objektif ini iaitu mengurangkan jumlah masa gerakan dan guna tenaga. Gambaran kesan awal nilai sudut belitan dan pecutan pada guna tenaga dan purata kelajuan dihasilkan. Dapatan simulasi menunjukkan corak berkala tork sendi yang tetap terhasil semasa robot ini berkeadaan lengkung serpenoid ketika bergerak. Tambahan, Pareto-optimal berdepan terhasil bagi solusi optimum pada kedua-dua objektif, sementara kaedah berat campuran digunakan bagi menentukan solusi terbaik. Akhirnya, dapatan simulasi disahkan secara eksperimen pada robot ular lapan-bahagian dengan menimbangkan kekurangan servomotor yang digunakan dalam eksperimen. Dapatan eksperimen menunjukkan sudut belitan 30° adalah sudut belitan optimum bagi kedua-dua objektif iaitu mengurangkan tenaga dan masa gerakan.

KEYWORDS: *snake robot; serpentine locomotion; genetic algorithm; multi-objective optimization; Pareto optimality*

1. INTRODUCTION

Snake robots are robotic mechanisms inspired by the motion of biological snakes. Snake robots were studied during the last five decades, although the early work on real snakes was conducted by the zoologist, Gray [1]. The first snake robot was developed by Hirose in 1972 [2]. A considerable number of research projects on snake robots focused on their design and modelling; thus, various models of snake-like robots were proposed in the literature [3-7]. On the other hand, several studies focused on the control of the snake

robot locomotion [8,9], while others focused on applications for snake robots such as pipe inspection [10], and rescue tasks [11,12].

The pioneer model of a snake robot was presented by Hirose in his book [3], where he presented a thorough study of the locomotion patterns of the natural snake and formulated what he called a *serpenoid curve*, which closely mimics the locomotion of living snakes. Hirose used a snake robot with passive wheels in his experiments to achieve normal friction much higher than tangential friction, which was an essential principle in snake movement. On contrary, Saito et al. [4] designed a wheel-less snake robot with direct contact to the ground. The snake has the potential of adapting to the environment at the cost of increased power consumption. Saito developed his model based on Hirose's principle and obtained equations of motion for a multi-link snake robot travelling with serpentine locomotion. Their model, which was based on the directional friction coefficients, had the feature of decoupling the inertial locomotion from the internal shape motion. Another detailed work on snake robot was introduced in [13]. They presented a systematic complete description of a snake robot in travelling wave locomotion based on the snake robot model introduced by Hirose. The effects of the initial *winding angle* and friction coefficient on the joint input torques were also discussed. Simulation and experiment in that study were carried out using locomotion along a vertical plane. However, this type of locomotion usually needs two periods of serpenoid curve to guarantee two supporting points in contact with the ground to keep the snake robot balanced.

Unlike traditional robots, snake robots are unique in terms of their shape variant, degree of freedom, and locomotion types. Hence, a snake robot has a controllable shape that can achieve a specific locomotion that helps the snake to follow a certain path in a certain direction. More precisely, the snake robot's amplitude (which is controlled by the initial *winding angle*) and the number of cycles can be controlled so that the shape and the size of the snake robot are both variable. Besides, the acceleration of the snake robot plays a role that affects the *travelling speed* and *energy consumption*, and that leads to the need for optimization of locomotion parameters. Therefore, optimization was another area of interest in the research on snake robots, as noticed in [14]. The authors in this study introduced the snake robot model in such a way that optimization could be used to find the optimal parameters for power and speed efficiency. In their study, they used Pareto optimality to obtain the optimal values of snake parameters. For example, they found that one period of the serpenoid curve could achieve the optimal power efficiency. Similar work was proposed in [15] for the locomotion efficiency of snake robots. Optimization was also used by [16], who proposed a path planning technique based on a serpenoid curve and real time dual genetic algorithm to control the snake-like robot. They used multi-objective optimization to find the shortest path and the minimum curvature deviation, as well as decrease the influence of motion error. Another study in [17] used genetic algorithm (GA) to find the optimal central pattern generator (CPG) parameters in terms of moving speed of a snake robot considering the effect of friction coefficients. The study included the design of a fuzzy tuner for CPG-network parameters during robot motion to achieve maximum speed. Genetic algorithm (GA) was also used to optimize the control parameters of a snake robot in [18]. The proposed PI/Backstepping control in that study aimed to reduce the tracking error and control energy in snake robot motion.

The focus of this paper is on multi-objective optimization of a snake robot's parameters to enhance its efficiency in terms of *travelling speed* and *energy consumption*. This study considers the serpentine locomotion of the snake robot while travelling along a straight line.

2. SYSTEM DESCRIPTION

In this section, a brief description is introduced for the snake robot model used in this study including the dynamics of the snake robot. Furthermore, the parameters that play role in achieving the objective functions are also discussed.

2.1 Model of Snake Robot

The model of snake robot used in this study is the one introduced in [3]. In that model, the serpenoid curve is used to control the body shape of the snake robot during travel in serpentine locomotion. The serpenoid curve can be achieved by the curvature function as follows:

$$\rho(s) = \frac{-2\pi\alpha}{L} sin\left(\frac{2\pi}{L}s\right) \tag{1}$$

where L is the length of the snake body, α is the initial *winding angle* of the curve, and s is the body length along the body curve, as shown in Fig. 1.
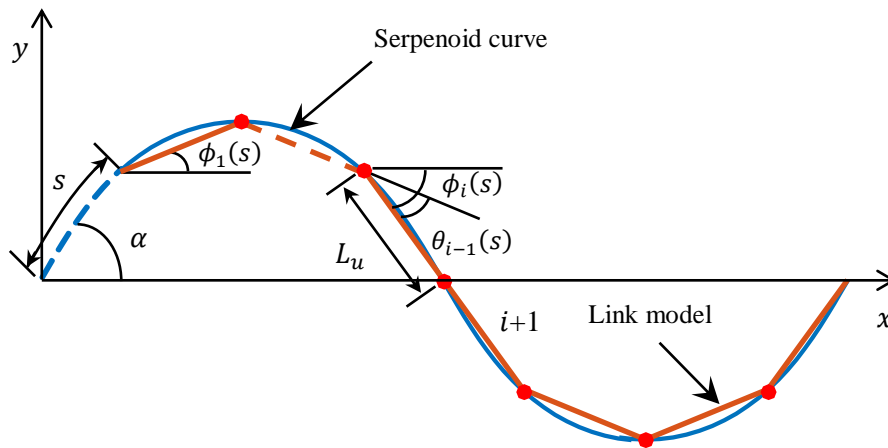

Fig. 1: Snake robot scheme with serpenoid curve.

Here, all the links are assumed to have the same length L_u. Hence, by integration of the curvature function in Eq. (1), the approximate values of relative angle $\theta_i(s)$ can be derived as follows

$$\theta_i(s) = \int_{s+(i-1)L_u}^{s+iL_u} \rho(u)du = -2\alpha \, sin\left(\frac{\pi}{n}\right).sin\left(\frac{2\pi s}{L} + \frac{2\pi i}{n} - \frac{\pi}{n}\right) \tag{2}$$

Consequently, angular velocity $\dot{\theta}_i(s)$ and angular acceleration $\ddot{\theta}_i(s)$ can be obtained as follows:

$$\dot{\theta}_i(s) = -\frac{4\pi\alpha}{L}.sin\left(\frac{\pi}{n}\right).cos\left(\frac{2\pi s}{L} + \frac{2\pi i}{n} - \frac{\pi}{n}\right)\dot{s} \tag{3}$$

$$\ddot{\theta}_i(s) = -\frac{4\pi\alpha}{L}.sin\left(\frac{\pi}{n}\right).cos\left(\frac{2\pi s}{L} + \frac{2\pi i}{n} - \frac{\pi}{n}\right)\ddot{s}$$

$$+\frac{8\pi^2\alpha}{L^2}.sin\left(\frac{\pi}{n}\right).sin\left(\frac{2\pi s}{L} + \frac{2\pi i}{n} - \frac{\pi}{n}\right)\dot{s}^2 \tag{4}$$

where

$$i = 1,2,\dots,n-1$$

n = number of links
L_u = length of single link
$\dot{s}$ is the velocity along the body curve
$\ddot{s}$ is the acceleration along the body curve given as

$$\ddot{s} = \begin{cases} a & 0 \le t < R \\ 0 & R \le t < T - R \\ -a & T - R \le t < T \end{cases} \tag{5}$$

where, a is the initial acceleration, R is the acceleration/deceleration time and T is the total travelling time.

Since $\ddot{s}$ is given, $\dot{s}$ and s can be found by finding the first integral and second integral of $\ddot{s}$, respectively. Based on the above equations, $\theta_i(s)$ can be varied to control the body shape and *travelling speed* of the snake robot.

2.2 Dynamics of the Snake Robot

As mentioned above, the locomotion of the snake robot relies on the difference between tangential friction and normal friction. Thus, the dynamics of the snake robot describes the relationship between the joint torques, frictional force and the robot locomotion as illustrated in Fig. 2.
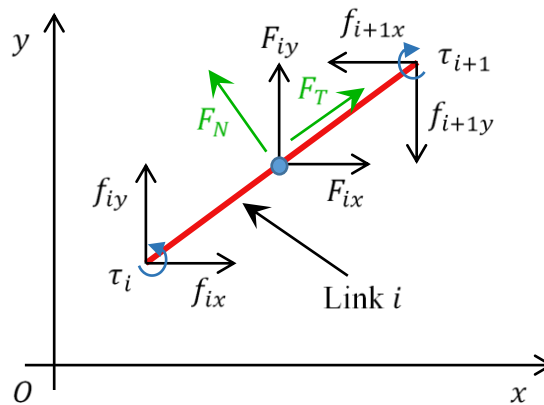

Fig. 2: Scheme of forces acting on link *i*.

To represent the environment dynamics, *Coulomb friction* is used so that the direction of motion is taken in consideration in terms of the sign of the velocity. Hence, tangential friction force (F_T) and normal friction force (F_N) for link i can be calculated as follows:

$$F_{iT} = \mu_t . \operatorname{sgn}(v_{iT}) . mg \tag{6}$$

$$F_{iN} = \mu_n . \operatorname{sgn}(v_{iN}) . mg \tag{7}$$

where μ_t and μ_n are tangential and normal friction coefficients of link i, while v_{iT} and v_{iN} are tangential and normal velocity at the center of gravity of link i. The latter two velocities can be calculated as follows:

$$v_{iT} = \dot{x}_{iG} \cos \phi_i + \dot{y}_{iG} \sin \phi_i \tag{8}$$

$$v_{iN} = \dot{x}_{iG} \sin \phi_i + \dot{y}_{iG} \cos \phi_i \tag{9}$$

where $\dot{x}$ and $\dot{y}$ are the velocity components at the centre of gravity of link i. Using Eqs. (6) and (7), the x and y components of the friction force are expressed as:

$$F_{ix} = F_{iT}.\cos\phi_i - F_{iN}.\sin\phi_i \tag{10}$$

$$F_{iy} = F_{iT}.\sin\phi_i - F_{iN}.\cos\phi_i \tag{11}$$

Referring to Fig. 2 and by applying Newton's second law of motion, the motion for the link i with respect to the link $i+1$ can be described as follows

$$f_{ix} - f_{i+1x} + F_{ix} = m\ddot{x}_{iG} \tag{12}$$

$$f_{iy} - f_{i+1y} + F_{iy} = m\ddot{y}_{iG} \tag{13}$$

$$\tau_i - \tau_{i+1} + (f_{ix} + f_{i+1x} + F_{ix})L_u\sin\phi_i/2 - (f_{iy} + f_{i+1y} + F_{iy})L_u\cos\phi_i/2 = I_i\ddot{\phi}_i \tag{14}$$

However, joint 1 and joint $n+1$ represent the tail and the head of the snake robot where no actuators are attached. Therefore, torque and force at both ends will be zeros as expressed in the following two equations:

$$f_{1x} = f_{1y} = f_{n+1x} = f_{n+1y} = 0 \tag{15}$$

$$\tau_1 = \tau_{n+1} = 0 \tag{16}$$

By taking summation of both sides of the recursive formulas Eqs. (12) and (13) and then satisfying the conditions in Eqs. (15) and (16), the force equations Eqs. (17) and (18) of link i are obtained.

$$f_{ix} = \sum_{j=i}^{n} m\ddot{x}_{jG} - \sum_{j=i}^{n} F_{jx} \tag{17}$$

$$f_{iy} = \sum_{j=i}^{n} m\ddot{y}_{jG} - \sum_{j=i}^{n} F_{jy} \tag{18}$$

Substituting Eq. (17) and (18) in the torque Eq. (14), we get:

$$\begin{aligned}
\tau_i - \tau_{i+1} + \left(2\sum_{j=i}^{n} m\ddot{x}_{jG} - 2\sum_{j=i}^{n} F_{jx} - m\ddot{x}_{iG} + F_{ix}\right)L_u\sin\phi_i/2 \\
- \left(2\sum_{j=i}^{n} m\ddot{y}_{jG} - 2\sum_{j=i}^{n} F_{jy} - m\ddot{y}_{iG} + F_{iy}\right)L_u\cos\phi_i/2 = I_i\ddot{\phi}_i
\end{aligned} \tag{19}$$

Consequently, summation of both sides of Eq. (19) gives:

$$\begin{aligned}
\sum_{i=1}^{n} \Bigg\{ \left(2\sum_{j=i}^{n} m\ddot{x}_{jG} - 2\sum_{j=i}^{n} F_{jx} - m\ddot{x}_{iG} + F_{ix}\right)L_u\sin\phi_i/2 \\
- \left(2\sum_{j=i}^{n} m\ddot{y}_{jG} - 2\sum_{j=i}^{n} F_{jy} - m\ddot{y}_{iG} + F_{iy}\right)L_u\cos\phi_i/2 \Bigg\} = \sum_{i=1}^{n} I_i\ddot{\phi}_i
\end{aligned} \tag{20}$$

By solving Eq. (20) we obtain the only unknown variable $(\ddot{\phi}_1)$ which is the rotation acceleration of the first joint. However, from kinematics of link i at the center of gravity we have:

$$\ddot{x}_{iG} = \ddot{x}_i - \frac{L_u}{2} \cos \phi_i \dot{\phi}_i - \frac{L_u}{2} \sin \phi_i \ddot{\phi}_i \tag{21}$$

$$\ddot{y}_{iG} = \ddot{y}_i - \frac{L_u}{2} \sin \phi_i \dot{\phi}_i + \frac{L_u}{2} \cos \phi_i \ddot{\phi}_i \tag{22}$$

Hence, substituting Eqs. (21) and (22) in Eqs. (17) and (18) gives:

$$n\ddot{x}_1 - \sum_{i=1}^{n} \left[\sum_{k=1}^{i-1} \left(L_u \cos \phi \dot{\phi}_k^2 + L_u \sin \phi_k \ddot{\phi}_k \right) + \frac{L_u}{2} \cos \phi_i \dot{\phi}_i + \frac{L_u}{2} \sin \phi_i \ddot{\phi}_i \right] = \sum_{i=1}^{n+1} F_{ix}/m \tag{23}$$

$$n\ddot{y}_1 - \sum_{i=1}^{n} \left[\sum_{k=1}^{i-1} \left(L_u \sin \phi \dot{\phi}_k^2 - L_u \cos \phi_k \ddot{\phi}_k \right) + \frac{L_u}{2} \sin \phi_i \dot{\phi}_i - \frac{L_u}{2} \cos \phi_i \ddot{\phi}_i \right] = \sum_{i=1}^{n+1} F_{iy}/m \tag{24}$$

Substituting ($\ddot{\phi}_1$) obtained from Eq. (20) into Eqs. (23) and (24), we get the linear acceleration of the first link ($\ddot{x}_1$) and ($\ddot{y}_1$).

Moreover, substituting the obtained variables from Eqs. (20), (23) and (24) into Eq. (19), all the joints' torques required to generate the robot motion are obtained. Consequently, the power consumption of each joint can be calculated as

$$P_i = |\tau_i \dot{\theta}_i| \tag{25}$$

Hence, total *energy consumption* of all joints for the travelling time, *T* can be calculated as

$$E_T = \sum_{i=1}^{n} \int_0^T |\tau_i \dot{\theta}_i| \, dt \tag{26}$$

Since time segment used in this case is discrete, *trapezoidal numerical integration* can be used to find an approximation of the total *energy consumption*.

2.3 Energy Consumption and Average Speed

As mentioned earlier in this paper, the target of this study is to achieve the maximum *average speed* (or minimum *travelling time*) of the snake robot using minimum *energy consumption*. For this purpose, we need to study the factors that may affect the *average speed* and the *energy consumption* of the snake robot. Namely, we focus on the *winding angle* and the *acceleration* of the snake robot, as illustrated in the following subsections.

2.3.1 Effect of Winding Angle on Energy Consumption and Average Speed

In order to study the effect of *acceleration* and *winding angle* of the snake robot on *energy consumption* and *average speed* of the forward motion, several simulations were performed. In these simulations, the snake robot travelled forward using serpentine locomotion for a distance of 3 meters. In the next step, *winding angle* was varied within the range 0.3 to 1.5 radians, while the *acceleration* was kept constant at 0.1 m/s². The results of this simulation are shown in Fig. 3.

As seen in Fig. 3, increasing the *winding angle* of the snake robot has an undesired effect on both *energy consumption* and *average speed*. In other words, if the *winding angle* is increased, the *average speed* of the snake robot is decreased, and the *energy consumption* is increased. Since the objective is to minimize the *energy consumption* as well as the *travelling time* of the snake robot, it is obviously clear that selecting a minimum *winding angle* would achieve both objectives. The detailed results of this simulation are listed in Table 1.
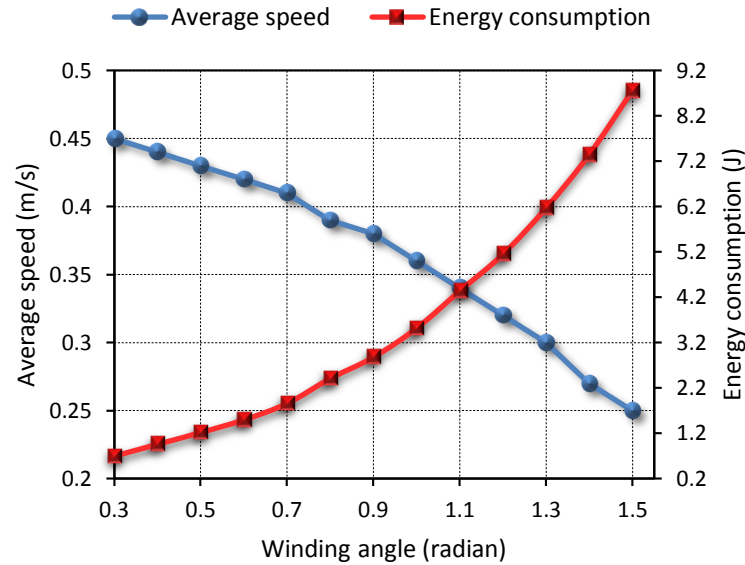
Fig. 3: Effect of *winding angle* on *energy consumption* and *average speed.*

Table 1: Effect of *winding angle* on *energy consumption* and *average speed*

Winding Angle		Travelling Time	Average Speed	Energy Consumption
radian	(°)	(s)	(m/s)	(J)
0.3	**17.2**	06.70	**0.45**	**0.70**
0.4	22.9	06.80	0.44	0.96
0.5	28.6	06.90	0.43	1.22
0.6	34.4	07.10	0.42	1.50
0.7	40.1	07.30	0.41	1.86
0.8	45.8	07.70	0.39	2.41
0.9	51.6	08.00	0.38	2.88
0.1	05.7	08.30	0.36	3.52
1.1	63.0	08.90	0.34	4.34
1.2	68.8	09.40	0.32	5.15
1.3	74.5	10.20	0.30	6.17
1.4	80.2	11.10	0.27	7.35
1.5	85.9	12.20	**0.25**	**8.76**

As illustrated in Table 1, while the *winding angle* is gradually increased from 0.3 to 1.5 radians, the *travelling time* of the snake robot is also increased. Accordingly, the *average speed* (travelled distance divided by *travelling time*) is decreased. Table 1 shows that applying minimum *winding angle* (0.3 radian) can achieve maximum *average speed* (0.45 m/s) and minimum *energy consumption* (0.70 J). On the contrary, applying maximum *winding angle* (1.5 radian) leads to undesired minimum *average speed* (0.25 m/s) and undesired maximum *energy consumption* (8.76 J).

2.3.2 Effect of Acceleration on Energy Consumption and Average Speed

To study the effect of *acceleration* along the body of the snake robot on *energy consumption* and *average speed*, simulation tests were carried out using different values of

acceleration ranging between 0.1 and 1.3 m/s^2, while the *winding angle* was kept constant at $\pi/3$ radian. Simulation results are shown in Fig. 4.
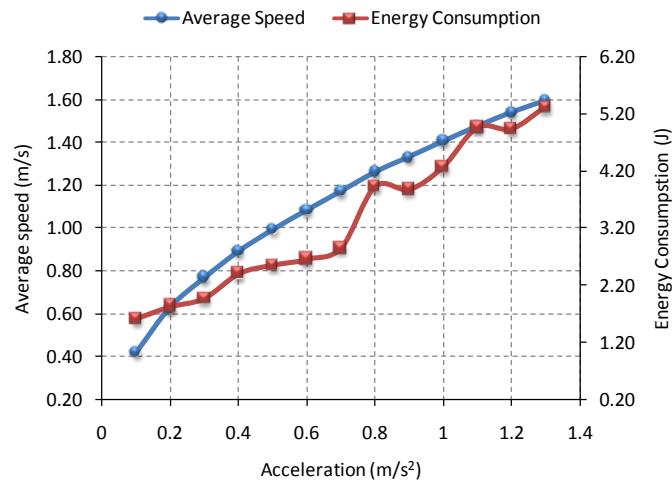

Fig. 4: Effect of *acceleration* on *energy consumption* and *average speed*.

Figure 4 demonstrates that the *acceleration* along the body of the snake robot has a contradictory effect on *average speed* and *energy consumption*. Hence, increasing the *acceleration* will increase *average speed*, and that matches the first objective. On the other hand, increasing the *acceleration* will increase *energy consumption* of the snake robot, which mismatches the second objective. Therefore, there is no single value of *acceleration* that can achieve both objectives.

For more details, Table 2 illustrates the effect of the initial *acceleration* of the snake robot on *average speed* and *energy consumption*. For example, applying the maximum *acceleration* (1.3 m/s^2) achieves the maximum *average speed* (1.60 m/s), however, that leads to undesired maximum *energy consumption* (5.32 J). While applying the minimum *acceleration* (0.1 m/s^2) achieves the minimum *energy consumption* (1.61 J), but leads to undesired minimum *average speed* (0.42 m/s).

Table 2: Effect of *acceleration* on *energy consumption* and *average speed*

Acceleration (m/s^2)	Travelling Time (s)	Average Speed (m/s)	*Energy Consumption* (J)
0.1	7.2	**0.42**	**1.61**
0.2	4.9	0.63	1.83
0.3	4.0	0.77	1.96
0.4	3.5	0.89	2.40
0.5	3.1	0.99	2.55
0.6	2.8	1.08	2.65
0.7	2.6	1.17	2.85
0.8	2.5	1.27	3.93
0.9	2.3	1.33	3.88
0.1	2.2	1.41	4.27
1.1	2.1	1.48	4.97
1.2	2.0	1.54	4.94
1.3	1.9	**1.60**	**5.32**

In conclusion, both objective functions (minimizing of *energy consumption* and maximizing of *average speed*) are dependent on *winding angle* and *acceleration* of the

snake robot. Consequently, the latter two parameters need to be optimized in order to achieve the mentioned objective functions. Once the optimization parameters are defined, the simulation setup can be started as described in the following section.

3. SIMULATION SETUP

Multi-objective optimization described in the previous section was implemented using the *gamultiobj* function in MATLAB that applies a genetic algorithm to find *Pareto-optimal front* of multiple objective functions. Pareto-optimal front contains optimal solutions that fit the objective functions. The general form of the *gamultiobj* function is expressed as follows:

[X, FVAL] = gamultiobj (FitnessFcn, nVars, A, b, , , LB, UB, options)

where X: is a matrix of all solutions of the design variables (*winding angle* and *acceleration*)

FVAL: is a matrix of all values of objective functions defined in *FitnessFcn* (*average speed* and *energy consumption*)

A fundamental step in optimization techniques is that initial values need to be set before moving forward in the optimization process. The initial values of the function's parameters that have been used in this study are described in Table 3.

Table 3: Initial values for parameters of *gamultiobj* function

Parameter	Value	Description
FitnessFcn	multiObjFcn	Fitness Function
nVars	2	Number of design variables
LB	$[\pi/6, 0.1]$	Lower bounds
UB	$[\pi/3, 2.0]$	Upper bounds
PopulationSize	50	Number of individuals
Generations	15	Total number of generations

The fitness function in this simulation uses the snake locomotion model as described in Section 2. The fitness function has two parameters as input: *winding angle* and *acceleration*, the values of which are generated by the genetic algorithm. On the other hand, the fitness function returns the values of the two parameters: *energy consumption* and *average speed*, which will be used by genetic algorithm to determine the fitness value of the input variables. Moreover, the other variables related to snake robot are initialized in the fitness function as described in Table 4.

Table 4: Initial values for snake robot variables

Variable	Value	Description
n	8	Number of links
L	1	Length of snake body (m)
m	0.1	Mass of single link (kg)
I	0.0001	Moment of inertia (kg m^2)
Mu_t	0.07	Tangential friction coefficient
Mu_n	0.7	Normal friction coefficient

The travelling distance was set to 3 meters, which is enough to get more than one full cycle of the snake shape. Meanwhile, the speed of the snake $\dot{s}$ along the serpenoid curve is determined by rise time R as described in $\ddot{s}$ function in Eq. (5). Arbitrarily, the value of R is set to 5 seconds. The process of multi-objective optimization is illustrated in the flowchart shown in Fig. 5.
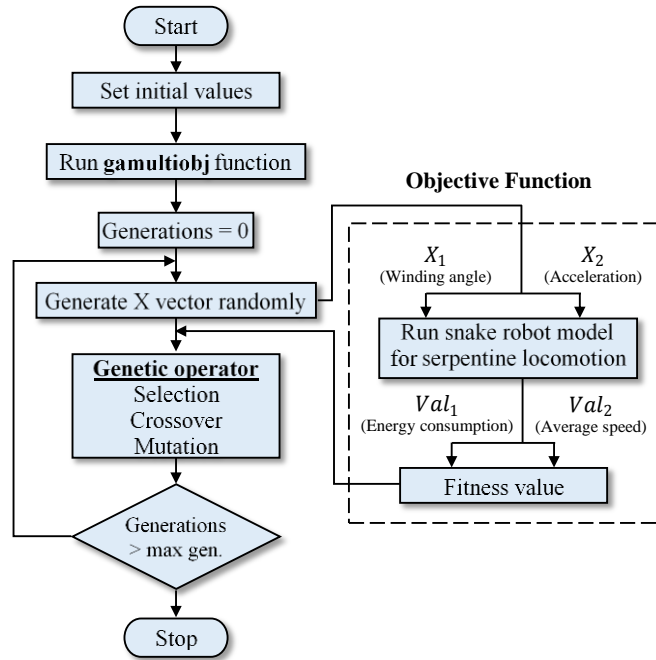

Fig. 5: Flowchart of multi-objective optimization process.

4. SIMULATION RESULTS

At first, the fitness function was simulated using settings in Section III. The fitness function uses kinematics and dynamics of the snake robot to achieve the serpentine locomotion. The results show a variety of input torques of the snake robot joints as seen in Fig. 6.
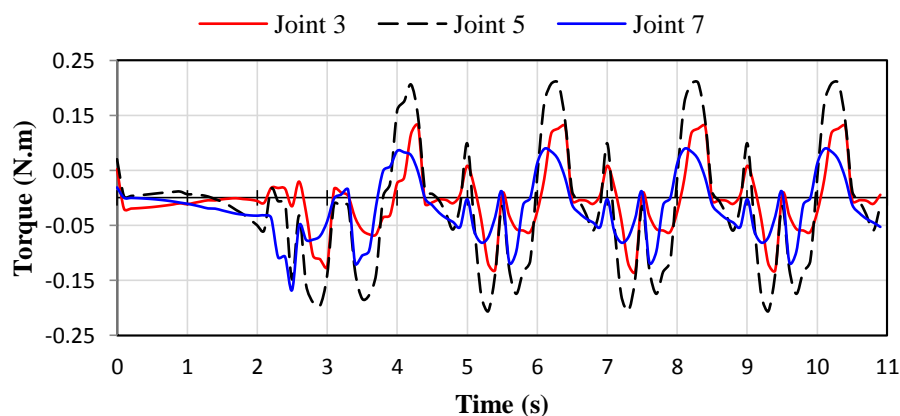

Fig. 6: Input torque for snake robot joints.

The pattern of the joints' torques shows a periodic shape because the snake is following a repeated sinusoidal path. However, it can be noticed that the middle joint

(joint 5) has the highest torque compared to the other joints. Consequently, the torque is lower when joints are farther from the centre of the snake robot (joints 3 and 7), which makes sense, as each joint is affected by the forces of the preceding and succeeding links. Several simulations were carried out using *gamultiobj* tool in order to optimize the *winding angle* and *acceleration* parameters. Therefore, different values of these two parameters were used for multi-objective optimization. On the other hand, *Population Size* was selected as follows: 20, 30, 40 and 50, along with different values of the number of *Generations* and *Pareto Fraction*. The travelling distance was selected as 3 meters. Eventually, the configuration that showed the least gaps between solution points was selected, which was as follows: *PopulationSize*: 50, *Generations*: 15, *Pareto Fraction*: 0.7. Using the previous configuration, simulation was repeated for three trials giving the optimal solutions using Pareto-optimal front as shown in Fig. 7.
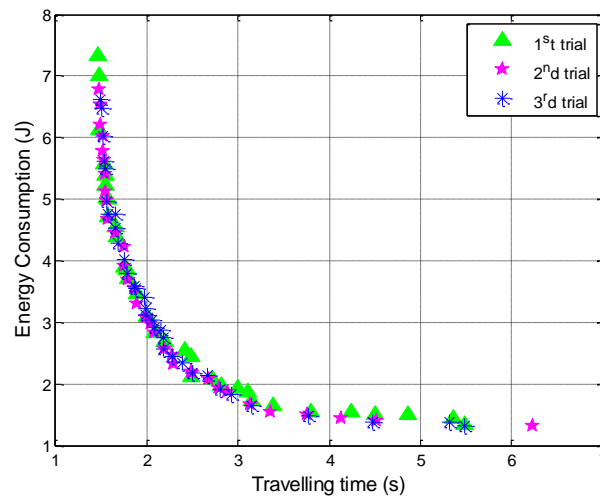

Fig. 7: Pareto-optimal front using (*PopulationSize* = 50, *Generations* = 15).

Despite using the same parameters in the three trials, the possible solutions are different from one trial to another as seen in Fig. 7. That is referred to the algorithm of the optimization technique, which uses random numbers for the initialization process. Nevertheless, by looking at the Pareto-optimal front in Fig. 7, all the possible solutions of the three trials lie within the same precision, which denotes the stability of the optimization process. The three trials have almost the same range of solutions that is located between 1.3 and 6.8 Joules for *energy consumption*, and between 1.5 and 6.2 seconds for *travelling time*. It is also noticed that concentration of solutions decreases as *travelling time* increases.

The solutions in the Pareto-optimal front are used to determine the relative angles for each joint of the snake robot to achieve the objective functions. Table 5 shows the relative joints' angles based on the optimized parameters. By looking at Table 5, it is noticed that all values of *winding angle* (α) are approaching $\pi/6$ which is the minimum value within the angle bounds as initialized in *gamultiobj* function. These values match with *winding angle* curve behaviour in Fig. 3. On the other side, values of *acceleration* have wider range that almost covers the whole range of *acceleration* (0.1 to 2.0) as initialized in the *gamultiobj* function. However, all Pareto optimal solutions in Table 5 are slightly different from each other, which denotes the fact that all solutions can be accepted with a little difference in fitting the objective functions

Table 5: Relative joints' angles for optimized parameters Pareto-optimal front

Pareto Solution	Winding Angle (°)	Acceleration (m/s²)	Joints' Angles (°)						
			θ_1	θ_2	θ_3	θ_4	θ_5	θ_6	θ_7
1	31.0	0.22	-09.2	-22.0	-21.8	-8.9	09.2	22.0	21.8
2	31.0	0.62	-09.5	-22.1	-21.7	-8.6	09.5	22.1	21.7
3	31.0	1.80	-10.3	-22.4	-21.4	-7.8	10.3	22.4	21.4
4	31.0	1.57	-10.1	-22.3	-21.5	-8.0	10.1	22.4	21.5
5	31.1	0.51	-09.5	-22.1	-21.8	-8.7	09.5	22.1	21.8
6	31.1	1.12	-09.9	-22.3	-21.7	-8.3	09.9	22.3	21.7
7	31.2	0.72	-09.6	-22.2	-21.8	-8.6	09.6	22.2	21.8
8	31.2	0.93	-09.8	-22.3	-21.8	-8.5	09.8	22.3	21.8
9	31.3	0.57	-09.6	-22.3	-21.9	-8.8	09.6	22.3	21.9
10	31.3	0.83	-09.7	-22.3	-21.9	-8.6	09.7	22.3	21.9
11	31.3	1.27	-10.0	-22.5	-21.7	-8.3	10.0	22.5	21.7
12	31.3	0.31	-09.4	-22.2	-22.0	-8.9	09.4	22.2	22.0
13	31.3	1.87	-10.5	-22.6	-21.6	-7.8	10.5	22.6	21.6
14	31.3	1.61	-10.3	-22.6	-21.6	-8.0	10.3	22.6	21.6
15	31.3	1.41	-10.1	-22.5	-21.7	-8.2	10.1	22.5	21.7
16	31.3	0.78	-09.7	-22.4	-21.9	-8.6	09.7	22.4	21.9
17	31.3	1.78	-10.4	-22.6	-21.6	-7.9	10.4	22.6	21.6
18	31.4	0.95	-09.9	-22.5	-21.9	-8.5	09.9	22.5	21.9
19	31.5	1.26	-10.1	-22.6	-21.9	-8.3	10.1	22.6	21.9
20	31.5	1.13	-10.0	-22.6	-22.0	-8.4	10.0	22.6	22.0
21	31.6	1.03	-10.0	-22.6	-22.0	-8.5	10.0	22.6	22.0
22	31.6	0.16	-09.4	-22.4	-22.3	-9.1	09.4	22.4	22.3
23	31.8	1.84	-10.6	-23.0	-21.9	-8.0	10.6	23.0	21.9
24	31.8	1.93	-10.7	-23.0	-21.9	-7.9	10.7	23.0	21.9
25	31.8	0.15	-09.4	-22.5	-22.5	-9.2	09.4	22.6	22.5

4.1 Selection of the Best Solution

After obtaining the Pareto optimal solutions, which can be considered as equally accepted solutions, only one solution is needed to achieve the objective functions. For such a case, several methods are available and can be used to select a single solution from all Pareto optimal solutions. One of these methods *the weighted sum method* (also called *utility function*), which is one of the simplest and most meaningful methods for solving multi-objective optimization problems [19]. Moreover, *the weighted sum method* is more suitable for a convex Pareto frontier (which is the case in this study). In this method, a weighting factor (w_i) is given for each of the objective functions (f_i) based on its importance compared to the other objective functions. Then a total or overall utility function U can be defined as follows [20]:

$$U = \sum_{i=1}^{k} w_i f_i(X) \tag{27}$$

where

w_i is the weighting factor
k is the number of objective functions
X is a vector of design variables.

The solution can be found by maximizing (or minimizing) the total utility U based on the objective function. The result of this method is a single value, but it may denote more than one solution when they have the same utility value.

Let us say there are two objective functions: f_1 and f_2, then there should be two preselected weighting factors: w_1 and w_2. The weights used in this method usually satisfy the normalization condition:

$$\sum_{i=1}^{k} w_i = 1 \tag{28}$$

Using the solution set of trial 1 in Fig. 7, the *weighted sum method* was applied several times using various weights for each of the objective functions. However, all values of the objective functions need to be normalized (between 0 and 1) to eliminate the effect of the big difference among values of different units. Hence, the following formula was used for normalization purposes:

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{29}$$

Each time the *weighted sum method* was applied, only one solution was selected out of all solutions in the Pareto curve. The results of using different combination of weights are shown in Table 6, where f_1 and f_2 represent the objective functions, while w_1 and w_2 represent their weights.

As mentioned earlier, the *weighted sum method* needs a predefined weight for each of the objective functions. These weights are user-defined parameters, and vary from one case to another. Therefore, Table 6 has various values to include a wide range of weights for each objective function to show the effect of this variety on the selection of the best answer. The last two columns in Table 6 clearly show that the values match with the behaviour shown on the curves in Fig. 3 and Fig. 4. For example, when high priority is given to minimize *energy consumption*, as in the first case in Table 6, the minimum *energy consumption* is achieved (1.323 J) with the cost of minimum *average speed* (0.119 m/s). On the other hand, when high priority is given to maximize the *average speed*, as in the last case in Table 6, the maximum *average speed* is achieved (2.039 m/s) with the cost of maximum *energy consumption* (6.784 J). Nevertheless, in some of the cases, different pairs of weights may lead to the same result, like cases 6 and 7 in the same table. For more clarification, Fig. 8 illustrates the optimization results, including all possible solutions, the Pareto-optimal front, and the best solution (using $w_1 = 0.4$ and $w_2 = 0.6$).

Table 6: Applying *weighted sum method* using variant weights for objectives functions

| w_1	w_2	$w_1 f_1$	$w_2 f_2$	$\sum_{i=1}^{2} w_i f_i$	α (radian)	Acceleration (m/s²)	Avg. Speed (m/s)	Energy cons. (J)
0.10	0.90	0.100	0.000	0.100	0.541	0.119	**0.481**	**1.323**
0.20	0.80	0.147	0.034	0.181	0.541	0.393	0.894	1.552
0.34	0.66	0.236	0.041	0.277	0.545	0.449	0.956	1.662
0.40	0.60	0.188	0.110	0.298	0.541	0.848	1.309	2.326
0.54	0.46	0.158	0.167	0.325	0.543	1.244	1.584	2.326
0.60	0.40	0.051	0.246	0.297	0.541	1.785	1.907	4.679
0.70	0.30	0.059	0.184	0.244	0.541	1.785	1.907	4.679
0.82	0.18	0.054	0.126	0.180	0.541	1.823	1.937	5.131
0.90	0.10	0.000	0.100	0.100	0.551	1.975	**2.039**	**6.784**
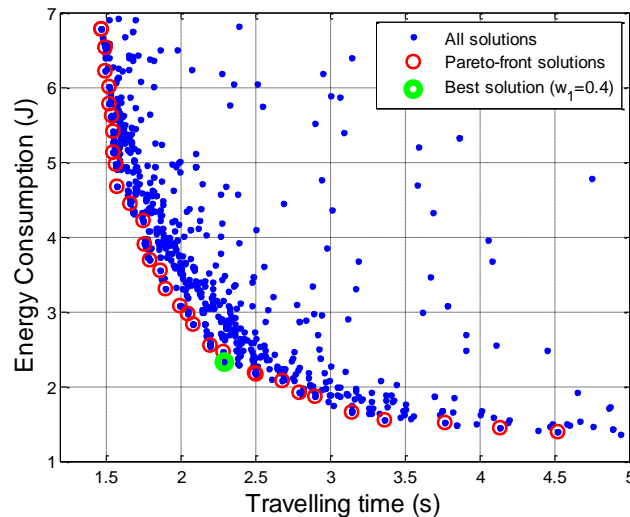
Fig. 8: Pareto-optimal front showing dominated points.

4.2 Validation of the Obtained Results

In order to validate the obtained results of multi-objective optimization, arbitrarily variant values of *winding angle* and *acceleration* were tested and compared with the optimized parameters. The tested values were selected from the same range of *winding angle* ($\pi/6$ to $\pi/3$) and *acceleration* (0.1 to 2.0) as initialized in *gamultiobj* function. The comparison of results is shown in Table 7.

Table 7: Comparison of selected and optimized parameters

Winding angle radian	(°)	Acceleration (m/s²)	Average Speed (m/s)	Energy Cons. (J)	Remarks
1.047	60.0	0.100	0.349	03.978	Not optimized
1.047	60.0	0.500	0.860	07.907	Not optimized
1.047	60.0	1.100	1.285	17.501	Not optimized
1.047	60.0	1.400	1.440	19.691	Not optimized
0.785	45.0	0.200	0.599	02.743	Not optimized
0.785	45.0	0.700	1.118	04.749	Not optimized
0.785	45.0	1.000	1.345	07.136	Not optimized
0.785	45.0	1.500	1.659	11.385	Not optimized
0.628	36.0	0.300	0.770	01.962	Not optimized
0.628	36.0	0.600	1.085	02.653	Not optimized
0.628	36.0	0.120	0.473	01.644	Not optimized
0.542	31.1	0.314	0.797	01.511	Optimized
0.541	31.0	0.848	1.309	02.326	Optimized
0.541	31.0	1.272	1.611	03.549	Optimized
0.543	31.1	1.427	1.709	03.909	Optimized
0.541	31.0	1.823	1.937	05.131	Optimized

The results in Table 7 clearly show that using any combination of values for *winding angle* and *acceleration* other than the values obtained by Pareto-optimal front do not achieve better results in terms of fitting the objective functions. Therefore, these results show the validity of using multi-objective optimization technique for snake robot locomotion.

5. EXPERIMENTAL RESULTS

To verify the simulation results, several experiments were conducted to check the effect of changing the related factors on power consumption, namely, acceleration, *winding angle* and surface friction.

5.1 Effect of Acceleration on Energy Consumption

Acceleration here means the change of speed in the forward motion of the snake robot along the serpentine path after it starts moving. Consequently, this acceleration will determine the forward speed of the snake robot, which is theoretically assumed to be constant. However, direct control of acceleration is not an easy job; instead, controlling the rotation speed of the servomotors is much easier. Accordingly, the effect of the acceleration on power consumption can be simply found by varying the rotational speed of the servomotors and getting the final reading of *energy consumption*.

To conduct the experiment, the value of the *winding angle* needs to be fixed, while the rotation speed of the servomotor needs to be changed within a specific range. For more analysis, the experiment was repeated several times using a range of *winding angle* from 20° to 45° with a 5-step increment, while the range of the servomotor *rotation speed* was varied for each *winding angle* between the *rotation speed code* '30' which represents 0.77 rad/s, and the *rotation speed code* '60' which represents 1.43 rad/s.

In this experiment, the snake robot was run 42 times for 6-meter travel. Meanwhile, the *winding angle* and *rotation speed* were varied for each experiment. Digital wattmeter was used to measure the total *energy consumption* for each travel of the snake robot. The results are illustrated in Fig. 9.
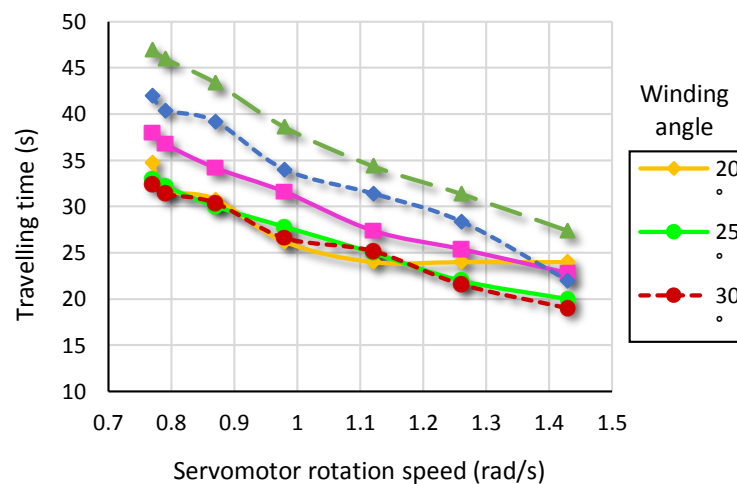

Fig. 9: Travelling time for 6-meter distance.

Figure 9 illustrates the relationship between the servomotor *rotation speed* and the travelling time of the snake robot. The experiments were conducted using different values of servomotor *rotation speed*, starting from *rotation speed code* '30' (0.77 rad/s) until *rotation speed code* '60' (1.43 rad/s) with a 5-step increment. Meanwhile, the *winding angle* was varied from 20° to 45° with a 5-step increment also. As seen in the above figure, all the curves demonstrate a similar trend, which shows that the travelling time is inversely proportional to the servomotor *rotation speed*. In other words, increasing the servomotor *rotation speed* can decrease the travelling time of the snake robot. It is also noticed that the travelling time increased when the *winding angle* was above 30° as the body of snake

robot became more twisted. Therefore, the objective of decreasing the travelling time of the snake robot when it is used for exploration task can be achieved by simply increasing the rotation speed of the servomotors.

The speed range used in these experiments considered the limitations of the servomotors. Therefore, the minimum rotation speed selected was 0.77 rad/s, since the speed below this value is relatively slow, while the maximum speed was 1.43 rad/s, as controlling the snake robot became difficult beyond that speed.

5.2 Effect of Winding Angle on Energy Consumption

The following experiment aims to find the relationship between the *winding angle* of a snake robot and the *energy consumption* after travelling a specific distance. This experiment was actually a part of the previous experiment, where the actual time needed by the snake robot to travel for 3 meters and 6 meters was measured and recorded. As mentioned in the previous experiment, the range of *winding angle* was 20° to 45° while the range of servomotor *rotation speed* was from 0.77 rad/s to 1.43 rad/s.

In this experiment, the *acceleration* of the snake robot or, in other words, the servomotor *rotation speed* was kept constant (e.g. 1.43 rad/s) while the *winding angle* was varied within a range of 30° to 60°. Later on, the experiment was repeated using different values of *winding angles* while the servomotor *rotational speed* was kept constant. In this case, increasing the *winding angle* will obviously increase the undulation of the snake robot and consequently reduce its length (along the body axis) as shown in Fig. 10, which demonstrates snapshots of a snake robot with three different *winding angle*s.

$\alpha = 20°$
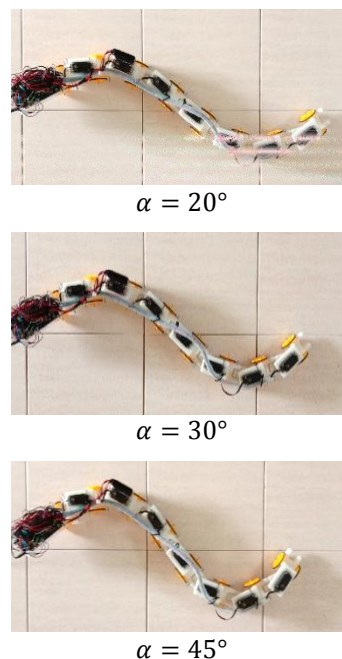
$\alpha = 30°$

$\alpha = 45°$

Fig. 10: Undulation of the snake robot with different *winding angles*.

Based on the above figure, increasing the undulation of the snake robot increases the travelling distance (because of the body shrinking), thus, the snake robot will take a longer time to reach its destination. The *energy* consumptions were measured starting from the beginning of each travel until the end of each travel of the snake robot using a digital wattmeter. The results of these experiments for 3 meters and 6 meters travelling distances are presented in Tables 8 and 9, and plotted in Figs. 11 and 12 respectively.

Table 8: *Energy consumption* for 3-meter distance (Wh)

Winding Angle (°)	Servomotor rotation speed (rad/s)						
	0.77	0.79	0.87	0.98	1.12	1.26	1.43
20	72	72	86	65	65	65	58
25	72	72	65	65	65	72	58
30	79	79	72	72	72	58	65
35	79	79	86	72	72	72	65
40	86	94	94	94	79	79	72
45	101	101	94	94	101	94	86

Table 9: *Energy consumption* for 6-meter distance (Wh)

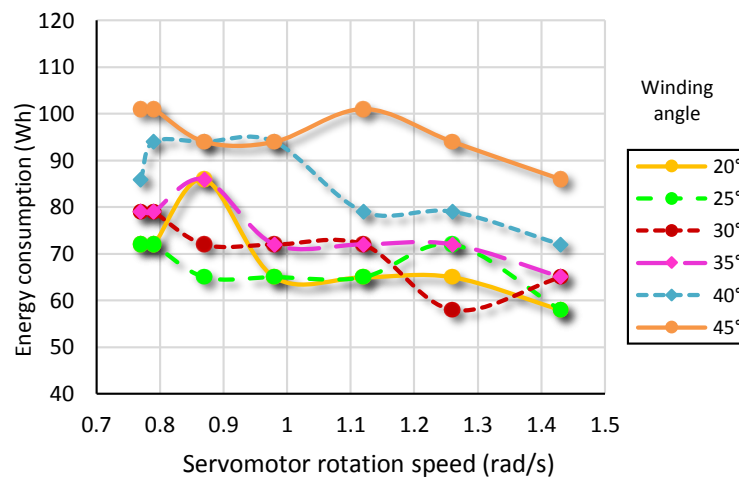| Winding Angle (°)	Servomotor rotation speed (rad/s)						
	0.77	0.79	0.87	0.98	1.12	1.26	1.43
20	137	122	115	122	115	108	108
25	137	122	122	122	122	115	108
30	130	122	115	122	115	115	101
35	137	144	137	130	130	130	115
40	158	166	158	151	144	144	137
45	173	166	166	166	173	158	144

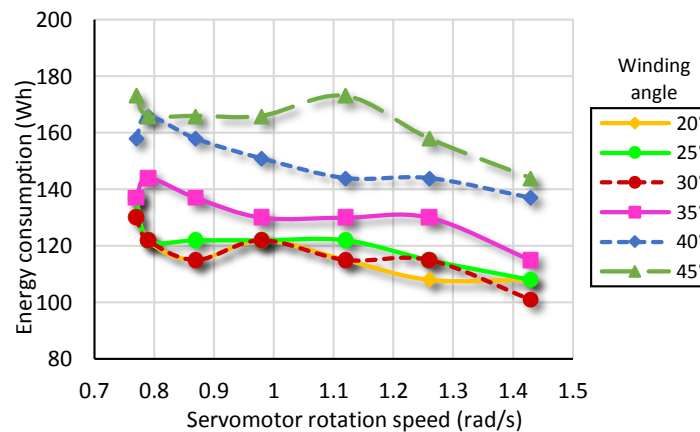Fig. 11: *Energy consumption* for 3-meter distance.

Fig. 12: *Energy consumption* for 6-meter distance.

The experiments were conducted using different values of servomotor *rotation speed*, starting from *rotation speed code* '30' (0.77 rad/s) until *rotation speed code* '60' (1.43 rad/s) with a 5-step increment. Meanwhile, the *winding angle* was varied from 20° to 45° with a 5-step increment. The curves in both the figures demonstrated a general trend showing that *energy consumption* is inversely proportional to the servomotor *rotation speed*. However, that is not the case in some ranges such as in the range between the rotation speed of 0.87 rad/s and 1.12 rad/s for the *winding angles* 25° and 30°.

Furthermore, it can be noticed from Figs. 11 and 12 that doubling the travelling distance did not necessarily double the *energy consumption*. Instead, the *energy consumption* for the 6 meters travelling distance increased in the range of 1.3 to 2 times the energy consumption of the 3 meters travelling distance for different winding angles and servomotor speeds.

On the other hand, the direct proportional relationship between the *winding angle* and the *energy consumption* can be seen clearly in both figures, which means that increasing the *winding angle* increases the *energy consumption*. However, the curve of 30° *winding angle* demonstrates an exception as it shows less *energy consumption* than the 25° curve, and even less than the 20° curve in some points.

It can be deduced from the above analysis that minimizing *energy consumption* can be achieved by selecting a low value of *winding angle*. However, the practical experiment proved that the optimal *winding angle* is not necessarily the one with minimum value as seen in Fig. 11, where the 30° curve shows the minimum *energy consumption* despite it not being the minimum *winding angle*. Therefore, from an optimization perspective, selection of the *winding angle* of 30° could achieve the minimum *energy consumption* in serpentine locomotion of the snake robot.

6. COMPARISON OF SIMULATION AND EXPERIMENTAL RESULTS

A comparison is conducted between simulation results as shown in Fig. 13 and experimental results shown in Fig. 12, with respect to *energy consumption* and *travelling speed* of the snake robot. In both simulation and experiment, the snake robot was tested for travelling for a specific distance using different values of *winding angle* and *rotation speed* at each time. The trend of energy consumption is divided into three regions as shown in Fig. 13. In the first region (*Region 1*) and the last region (*Region 3*), the *energy consumption* is directly proportional to *acceleration*, while the middle region (*Region 2*) demonstrates an inversely proportional relationship.

By comparing both figures, it can be seen that *energy consumption* in Fig. 12 is inversely proportional to servomotor *rotation speed*, which matches the trend of the middle region (marked with a red rectangle) in Fig. 13, where the *energy consumption* is inversely proportional to *acceleration*. Although the x-axes are not the same in both figures, they both lead to the average speed of the snake robot.

It can be concluded from the above figures that the rotation speed of the servomotors used in this experiment is limited. Therefore, the experimental *energy consumption* shown in Fig. 12 cannot demonstrate the trend of *Region 3* of the simulation results as seen in Fig. 13.
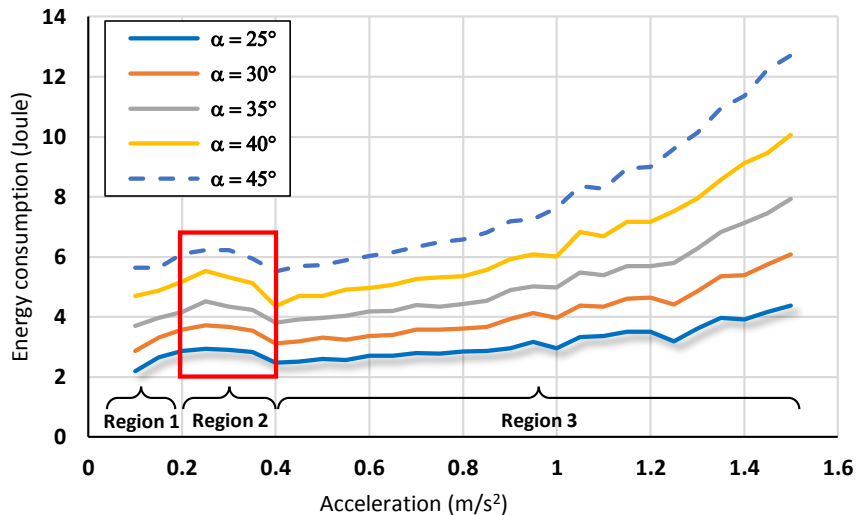
Fig. 13: *Energy consumption* in simulation results.

Moreover, simulation results show that for achieving the objectives of minimizing *travelling time* and *energy consumption* the minimum *winding angle* and fastest *rotation speed* should be selected. On the other hand, the experimental results demonstrated different behaviour when servomotor *rotation speed* was set above 1.2 rad/s as seen in Fig. 9. In the mentioned figures, *winding angle*s 25° and 30° showed less *travelling time* than the *winding angle* 20°. Moreover, the experimental results in Fig. 12 show that the minimum *energy consumption* can be achieved using the *winding angle* of 30°.

Finally, it was experimentally found that the *winding angle* of 30° is the optimum *winding angle* that can achieve the objectives of minimizing the *energy consumption* and the *travelling time* when the snake robot travels in serpentine locomotion.

7. CONCLUSION

In this paper, a multi-objective optimization for snake robot locomotion is presented. The model of the snake robot is introduced including the dynamic effects of the environment including friction. The equations listed in this paper describe the model of the snake robot travelling in serpentine locomotion besides necessary calculation of *energy consumption* and *average speed* of the snake robot. Simulation results show that *acceleration* has a contradictory effect on *average speed* and *energy consumption*, while the initial value of *winding angle* affects both *energy consumption* and *average speed* analogously. MATLAB *gamultiobj* function was used to achieve two objectives: minimizing the total *travelling time* and minimizing the total *energy consumption* by obtaining the Pareto-optimal front. The comparison of selected and optimized parameters shows the validity of using a multi-objective optimization technique for snake robot locomotion to achieve both objectives. On the other hand, the experimental results conformed with the simulation results regarding the effect of acceleration on *energy consumption* except for the *winding angles* of 25°and 30°, which demonstrated less *energy consumption* than the *winding angle* of 20° when the servomotor *rotation speed* was above 1.2 rad/s. Furthermore, the experimental results showed that the *winding angle* 30° was the optimum *winding angle* that can achieve both objectives of minimizing the *energy consumption* and the *travelling time*.

ACKNOWLEDGEMENT

REFERENCES

[1]   Gray J. (1946) The mechanism of locomotion in snakes. Journal of Experimental Biology, 23(2): 101-120.

[2]   Hirose S, Mori M. (2004) Biologically inspired snake-like robots. Robotics and Biomimetics, 2004. ROBIO 2004. IEEE International Conference on (pp. 1-7). IEEE.

[3]   Hirose S. (1993) Biologically inspired robots: snake-like locomotors and manipulators, vol. 1093. Oxford: Oxford university press.

[4]   Saito M, Fukaya M, Iwasaki T. (2002) Modeling, analysis, and synthesis of serpentine locomotion with a multilink robotic snake. IEEE Control Systems Magazine, 22(1): 64-81.

[5]   Chen L, Ma S, Wang Y, Li B, Duan D. (2007) Design and modelling of a snake robot in traveling wave locomotion. Mechanism and Machine Theory, 42(12): 1632-1642.

[6]   Chernousko FL. (2005) Modelling of snake-like locomotion. Applied mathematics and computation, 164(2): 415-434.

[7]   Ariizumi R, Matsuno F. (2017) Dynamic analysis of three snake robot gaits. IEEE Transactions on Robotics, 33(5): 1075-1087.

[8]   Sanfilippo F, Azpiazu J, Marafioti G, Transeth AA, Stavdahl Ø, Liljebäck P. (2017) Perception-driven obstacle-aided locomotion for snake robots: the state of the art, challenges and possibilities. Applied Sciences, 7(4): 336.

[9]   Wang Z, Gao Q, Zhao H. (2017) CPG-inspired locomotion control for a snake robot basing on nonlinear oscillators. Journal of Intelligent & Robotic Systems, 85(2): 209-227.

[10] Trebuňa F, Virgala I, Pástor M, Lipták T, Miková Ľ. (2016) An inspection of pipe by snake robot. International Journal of Advanced Robotic Systems, September-October, 1-12.

[11] Bando Y, Suhara H, Tanaka M, Kamegawa T, Itoyama K, Yoshii K, ... & Okuno HG. (2016) Sound-based online localization for an in-pipe snake robot. in 2016 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR) (pp. 207-213). IEEE.

[12] Whitman J, Zevallos N, Travers M, Choset H. (2018) Snake robot urban search after the 2017 Mexico City earthquake. 2018 IEEE international symposium on safety, security, and rescue robotics (SSRR) (pp. 1-6). IEEE.

[13] Chen L, Ma S, Wang Y, Li B, Duan D. (2007) Design and modelling of a snake robot in traveling wave locomotion. Mechanism and Machine Theory, 42(12): 1632-1642.

[14] Saito M, Fukaya M, Iwasaki T. (2002) Modeling, analysis, and synthesis of serpentine locomotion with a multilink robotic snake. IEEE Control Systems Magazine, 22(1): 64-81.

[15] Kelasidi E, Jesmani M, Pettersen KY, Gravdahl JT. (2018) Locomotion efficiency optimization of biologically inspired snake robots. Applied Sciences, 8(1): 80.

[16] Liu J, Wang Y, Ii B, Ma S. (2004) Path planning of a snake-like robot based on serpenoid curve and genetic algorithms. in Intelligent Control and Automation, 2004. WCICA 2004. Fifth World Congress on (6: 4860-4864). IEEE.

[17] Hasanzadeh S, Tootoonchi AA. (2008) Adaptive optimal locomotion of snake robot based on CPG-network using fuzzy logic tuner. in Robotics, Automation and Mechatronics, 2008 IEEE Conference on (pp. 187-192). IEEE.

[18] Jafari M, Shahmansoorian A. (2014). PI/backstepping control of snake robot optimized by genetic algorithm. In Electrical and Computer Engineering (CCECE), 2014 IEEE 27th Canadian Conference on (pp. 1-6). IEEE.

[19] Rao SS, Rao SS. (2009) Engineering optimization: theory and practice. John Wiley & Sons.

[20] Marler RT, Arora JS. (2010). The weighted sum method for multi-objective optimization: new insights. Structural and multidisciplinary optimization, 41(6): 853-862.