Article

Adaptive Particle Swarm Optimization of PID Gain Tuning for Lower-Limb Human Exoskeleton in Virtual Environment

Mohammad Soleimani Amiri 1, Rizauddin Ramli 1,*, Mohd Faisal Ibrahim 2, Dzuraidah Abd Wahab 1 and Norazam Aliman 3

1 Department of Mechanical and Manufacturing Engineering, Faculty of Engineering and Built Environment, Universiti Kebangsaan Malaysia, Bangi, Selangor 43600, Malaysia; msa0911@gmail.com (M.S.A.); dzuraidah@ukm.edu.my (D.A.W.)
2 Department of Electrical, Electronic and Systems Engineering, Faculty of Engineering and Built Environment, Universiti Kebangsaan Malaysia, Bangi, Selangor 43600, Malaysia; faisal.ibrahim@ukm.edu.my
3 Mechanical Engineering Department, Politeknik Sultan Azlan Shah, Perak, Behrang 35950, Malaysia; azam_justice_01@yahoo.com
* Correspondence: rizauddin@ukm.edu.my

Received: 15 October 2020; Accepted: 13 November 2020; Published: 16 November 2020

Abstract: Tuning of a proportional-integral-derivative (PID) controller for a complex multi-joint structure, such as an exoskeleton, using conventional methods is difficult and imprecise. In this paper, an optimal PID tuning method for a 3-dimensional model of a lower-limb human exoskeleton in gait training condition is presented. The dynamic equation of the human-exoskeleton is determined using a Lagrangian approach, and its transfer function is established in a closed-loop control system. PID controller gains, initialized by the Ziegler–Nichols (Z-N) method, are used as the input to an adaptive particle swarm optimization (APSO) algorithm for minimizing the multi-joint trajectory error. The optimized controller is tested in the Gazebo virtual environment and compared with the Z-N and conventional optimization methods. The numerical analysis shows that the PID controller tuned by a combination of Z-N and APSO improves the performance of a lower-limb human exoskeleton in gait training.

Keywords: exoskeleton; proportional-integral-derivative; particle swarm optimization; optimization; optimal control

1. Introduction

Stroke and spinal cord injury are the main causes of disability in the elderly [1,2], creating problems in daily life. For many years, rehabilitation therapies have been concentrated on recovering their motion function using various manual methods. In recent years, the lower-limb exoskeleton, which is a wearable robot, has been used to aid patients with mobility problems [3–5].

For the exoskeleton wearer to walk properly, the desired trajectory of the joints requires a robust control system to minimize steady-state errors to reduce disturbances. Huo et al. [6] proposed an active impedance control strategy for controlling a lower-limb exoskeleton for the sit-to-stand motion. Many researchers focused on adaptive-based control approaches to design a controller with uncertainties and non-linear systems. For instance, Daachi et al. [7] controlled one degree of freedom (DoF) lower-limb knee joint orthosis using a neural network based on adaptive control using the Lyapunov approach for rehabilitation reasons. Chen et al. [8] studied an adaptive robust cascade force control for a one DoF exoskeleton with a hydraulic actuator to minimize the effect of uncertainties.
Lu et al. [9] described an adaptive learning control system with unknown parameters using the Lyapunov approach for stability. In another study, Yang et al. [10] designed a robust learning controller with hybrid electro-hydraulic actuators, and verified their exoskeleton using a periodic reference.

For many years, proportional-integral-derivative (PID) controllers have been commonly used because of their reliable performance, and convenient and robust operation [11–13]. In our approach, we used particle swarm optimization (PSO) to tune the PID controller’s gains. PSO has a simple concept and is easy to implement in different optimization problems. Hence, it has been used to control the robustness of PID parameters [14]. PSO has few parameters that need to be adjusted, and it has high probability and efficiency in finding the global optima, but initializing its parameters can be difficult. Although PSO has fast error convergence, it can converge prematurely and be trapped in the local optima [15]. Therefore, researchers have combined different tuning methods to optimize its gains. For example, Belkadi et al. [16] developed an optimal method to obtain the parameters of the PID controller and verified the performance in a simulation using PSO with random initialization. Similarly, Durgadevi et al. [17] used PSO for tuning the parameters of proportional-integral (PI) and PID controllers of the temperature of a continuous stirred tank reactor.

As an alternative to PSO, other optimization methods have been used for tuning controller parameters [18]. Qianhao et al. [19] developed the whale optimization algorithm, which is a type of swarm intelligence optimization algorithm, to improve the performance of a linear–quadratic regulator controller. Pan et al. [20] presented the fuzzy-PID algorithm for the nonlinear dynamics and uncertainties of the human exoskeleton robot. Their method was verified in a co-simulation technique combining ADAMS and MATLAB/Simulink. Moharam et al. [21] used a hybrid differential evolution and PSO algorithm for tuning a PID controller and compared it with the genetic algorithm (GA) and the Ziegler–Nichols (Z-N) method to prove the hybrid method is robust to disturbances. Lal et al. [22] implemented the grasshopper optimization algorithm (GOA) to obtain the PID and fuzzy-PID controller parameters for a multi-area interconnected microgrid power system. They compared two different PID controllers and concluded that fuzzy-PID produced better dynamic responses compared to a PID controller tuned by GOA.

This paper presents the tuning of PID controller gains by combining Z-N and adaptive particle swarm optimization (APSO). A simulation and the dynamic equation of the human exoskeleton model walking with the aid of a parallel bar in a virtual environment are described. The transfer functions of each joint based on the dynamic equation of human exoskeleton model are established in a closed-loop control system. An objective function formulated by the error of the control system is defined. Our main objective was to optimize a PID controller using initialization with Z-N and optimized by APSO. The tuned PID was verified in the 3-dimensional virtual simulation environment and compared with the conventional method.

2. Human Exoskeleton Simulation in a Virtual Environment

The lower-limb exoskeleton model consists of a waist, left and right femurs, left and right tibias, and left and right feet. The human exoskeleton simulated model was created and controlled in a robot operating system (ROS) with average human anthropometric height. The waist functions as a base link to which the left and right femurs are connected. Consequently, other links are assembled to each other as parent and child links [23]. Figure 1a,b shows the hierarchical relation of the 3D exoskeleton model in an open-source dynamic simulator that imitates the behavior in a real environment.
To simulate the interaction between the exoskeleton and patient, a 3D human model was added in the Gazebo simulation environment. The foot, tibia, and femur of human are coupled to its related links. The configurations of coupling between human and exoskeleton model are shown in Figure 2.

To ensure the simulation accurately mimicked to the gait training task, parallel bars were added to avoid the human model from falling down by providing stability. Both hands of the human model are constrained to the parallel bars using a slide joint to allow forward and backward motion. Figure 3 demonstrates the gait training simulation of the human exoskeleton model walking with both hands holding the parallel bars.
3. Human Exoskeleton Dynamic Modeling

Figure 4 illustrates a free body diagram of the exoskeleton and human model. $O_1$, $O_2$, and $O_3$ are the center position of hip, knee, and ankle joint, respectively; $\theta_1$, $\theta_2$, and $\theta_3$ represent angles of hip, knee, and ankle, respectively; $l_1$, $l_2$, and $l_3$ are length of the femur, tibia, and foot, respectively; and $l_{Gi}$ is the center of gravity (CoG) of links 1, 2, and 3.
In this paper, each link of the exoskeleton and human is modeled as one DoF pendulum model, which replicates one joint of the exoskeleton and human, as shown in Figure 5. The summation of the physical effect of each lower link such as mass and inertia is considered to be $M_i$ [24].

![Figure 5. One degree of freedom (DoF) pendulum model of the exoskeleton.](image)

In this study, the Lagrangian formulation of dynamic equation of the pendulum model is expressed as follows:

$$L_i = E_{ki} - E_{pi},$$ (1)

$$T_i = \frac{d}{dt} \left( \frac{dL_i}{d\dot{\theta}_i} \right) - \frac{dL_i}{d\theta_i},$$ (2)

where $i$ represents number of the links; $L_i$ is the Lagrangian function; $\theta_i$ and $\dot{\theta}_i$ are the angular position and velocity, respectively; $T_i$ is the torque that applies to each joint; and $E_{ki}$ and $E_{pi}$ represent kinetic and potential energy [25], respectively, which are given as follows:

$$E_{ki} = \frac{1}{2} I_i \dot{\theta}_i^2 + \frac{1}{2} m_i (x_{Gi}^2 + y_{Gi}^2) + \frac{1}{2} M_i (x_i^2 + y_i^2),$$ (3)

$$E_{pi} = mg y_{Gi} + M_i g y_i,$$ (4)

where $m_i$ and $I_i$ are the mass and inertia of the link, respectively; $M_i$ is the mass of lower links; $g$ represents gravitational acceleration; $x_{Gi}, y_{Gi}, x_i, y_i, x_{Gi}, y_{Gi}, x_i,$ and $y_i$ represent the position of the CoG, end point of the link, and their velocity, respectively, as follows:

$$x_{Gi} = l_{Gi} \sin \theta_i, \quad \dot{x}_{Gi} = l_{Gi} \dot{\theta}_i \cos \theta_i, \quad x_i = l_i \sin \theta_i, \quad \dot{x}_i = l_i \dot{\theta}_i \cos \theta_i,$$ (5)

$$y_{Gi} = l_{Gi} \cos \theta_i, \quad \dot{y}_{Gi} = -l_{Gi} \dot{\theta}_i \sin \theta_i, \quad y_i = l_i \cos \theta_i, \quad \dot{y}_i = -l_i \dot{\theta}_i \sin \theta_i,$$ (6)

where $l_{Gi}$ and $l_i$ are the length of the CoG and the link $i$, respectively. The total kinetic and potential energies are determined as follows:

$$E_{ki} = \frac{1}{2} I_i \dot{\theta}_i^2 + \frac{1}{2} m_i l_{Gi}^2 \dot{\theta}_i^2 + \frac{1}{2} M_i l_i^2 \dot{\theta}_i^2,$$ (7)

$$E_{pi} = g \cos \theta_i (m_i l_{Gi} + M_i l_i).$$ (8)

The dynamic equation of the pendulum model is determined as follows:

$$T_i = (m_i l_{Gi}^2 + M_i l_i^2 + l_i) \ddot{\theta}_i + b \dot{\theta}_i + (l_i M_i + l_{Gi} m_i) g \sin \theta_i,$$ (9)

where $b$ is the friction coefficient and $\sin \theta_i$ is approximated as $\theta_i$. The transfer function $G(s)$ is represented in Laplace as follows:
\[ G(s) = \theta_i(s) = \frac{1}{s^2 + \frac{b}{(m_i l_i^2 + M_i l_i^2 + I_i)} s + \frac{(l_i M_i + l_i g_i)}{(m_i l_i^2 + M_i l_i^2 + I_i)}. \]  

(10)

Table 1 lists the physical features of the exoskeleton based on computer-automated design (CAD). The human model’s features were obtained from References [26–28]. The mass and inertia of the human model were for a subject that weighed 74.2 kg and was 175.5 cm tall.

**Table 1. Physical features of the exoskeleton and human model.**

|          | Exoskeleton | Femur | Tibia | Foot |
|----------|-------------|-------|-------|------|
| \( m \) (kg) | 0.2943      | 0.2159| 0.1150|      |
| \( l \) (m)   | 0.41        | 0.39  | 0.195 |      |
| \( I \) (kg m\(^2\)) | \(5.7 \times 10^{-3}\) | \(4.3 \times 10^{-3}\) | \(3.9 \times 10^{-4}\) |      |
| \( l_G \) (m) | 0.15        | 0.11  | 0.04  |      |

|          | Human       | Femur | Tibia | Foot |
|----------|-------------|-------|-------|------|
| \( m \) (kg) | 7.33       | 3.4503| 1.075 |      |
| \( l \) (m)   | 0.407       | 0.4334| 0.275 |      |
| \( I \) (kg m\(^2\)) | 0.1502     | 0.0505| 0.0038 |      |
| \( l_G \) (m) | 0.1763     | 0.1849| 0.1179|      |

Based on the features, the transfer function of the exoskeleton \( G_L(s) \) and human \( G_H(s) \) can be determined as shown in Table 2.

**Table 2. Transfer functions of the human–exoskeleton model.**

| Link    | Exoskeleton | Human Model |
|---------|-------------|-------------|
| Femur   | \( \frac{1}{s^2 + 13.25 + 25.96} \) | \( \frac{1}{s^2 + 27.27} \) |
| Tibia   | \( \frac{1}{s^2 + 36.88 + 27.55} \) | \( \frac{1}{s^2 + 29.24} \) |
| Foot    | \( \frac{1}{s^2 + 150.68 + 78.62} \) | \( \frac{1}{s^2 + 66.34} \) |

4. Control System

The plant of the system \( G(s) \) is defined as a combination of transfer functions of the human \( G_H(s) \) and exoskeleton \( G_L(s) \) in parallel. Figure 6 shows the control system with the feedback loop for the joint angle for each joint. Since the human model and exoskeleton are coupled together, it is assumed they are parallel to each other and the joints’ angles are equal. The PID controller employs the input as the error between the actual and desired trajectories, and the output is the torque as a plant [29]. The error is defined as follows:

\[ e = \theta_{act} - \theta_{des}. \]  

(11)

Figure 6. Control system.

The PID controller and plant model are represented as follows:

\[ C(s) = K_p + \frac{K_i}{s} + K_d s, \]  

(12)
where \( C(s) \) is the PID controller; \( K_p, K_i, \) and \( K_d \) are the proportional, integral, and derivative gains to be tuned to reduce the error, respectively [30]; and \( G_L(s) \) and \( G_H(s) \) are the transfer function of the exoskeleton and lower limb of the human, respectively.

5. Optimization of the Controller’s Gains

The PID controller is tuned using an algorithm by a combination of the Z-N method and APSO. The PID parameters are obtained from Z-N, which is the classical observation method of tuning [31]. The output is set as the initial value, which provides the searching space to avoid local optima [32]. APSO increases the speed of convergence and the chance of finding the global optimum. PSO is based on population, inspired from observation of biological social phenomena such as a flock of birds, swarms of bees, and schools of birds [33].

An objective function \( f_{obj} \) of the optimization problem is defined as the integral time absolute error (ITAE), which is the summation of the absolute error weighted by the elapsed time [34,35], as follows:

\[
f_{obj} = \int_0^\infty t |e_{obj}(t)| dt,
\]

where the error of control system is given as follows:

\[
e_{obj}(s) = 1 - \frac{C(s)G_i(s)}{1 - C(s)G_i(s)},
\]

where \( t \) is the elapsed time; \( e_{obj}(t) \) and \( e_{obj}(s) \) represent the error of the control system in the time and frequency domains, respectively. The tuning algorithm starts with Z-N, where \( K_i \) and \( K_d \) are set to zero while the \( K_p \) value is established as \( K_u \). The value of \( K_u \) is added until the first oscillation is observed, then its frequency is calculated as \( T_u \). The gains of the PID are determined as follows:

\[
K_p = 0.6 \times K_u,
\]

\[
K_i = \frac{1.2 \cdot K_u}{T_u},
\]

\[
K_d = \frac{3 \cdot K_u \cdot T_u}{40},
\]

where \( x_{zn} \) is the output that is input into APSO, which contains particles as design variables for minimizing the objective function. Initialization is created by randomizing \( x_{zn} \) as follows:

\[
x_{1j} = rand[x_{zn} - \epsilon, x_{zn} + \epsilon],
\]

where \( x_{1j} \) represents the particles in the first iteration; \( \epsilon \) is a positive value, which is selected as a random value between 0 and 1.

\[
\epsilon = rand(0, 1)
\]

The particles of the next iteration are a summation of the positions of previous particles with their velocities:

\[
x_{k,j} = x_{k-1,j} + v_{k,j},
\]

where \( k \) and \( j \) represent the numbers of iterations and particles, respectively; \( x_{k,j} \) and \( x_{k-1,j} \) are the position of the current particle and the particle of the previous iteration, respectively; and \( v_{k,j} \) is the velocity and direction of each particle toward the next iteration as follows:

\[
v_{k,j} = \omega_k \cdot v_{k-1,j} + C_1 \cdot rand_1 \cdot (p_{best_k} - x_{k,j}) + C_2 \cdot rand_2 \cdot (g_{best} - x_{k,j}),
\]
where \( r_{and1} \) and \( r_{and2} \) are random values between 0 and 1; \( \omega_i \) is a positive coefficient, called inertia weight, which reduces linearly with iteration, and is determined as follows:

\[
\omega_i = \omega_{\text{max}} \cdot b_i
\]  

(24)

where \( \omega_{\text{max}} \) is the maximum value for inertia weight; \( b \) is a gain that starts from 1 in the first iteration and end to 0, which allows \( \omega_i \) starts from the maximum value to have wider searching space and decrease to be narrowed toward the global optima \([36]\); and \( b \) is a positive descending coefficient, starting from 1 in the first iteration. \( b \) is represented as follows:

\[
b = 1 - a,
\]  

(25)

where \( a \) is a positive ascending coefficient, which starts from 0 and increases to 1 at the last iteration, as follows:

\[
a = \frac{k}{k_{\text{max}}} \quad k = 1, 2, ..., k_{\text{max}},
\]  

(26)

where \( k \) and \( k_{\text{max}} \) are the maximum number of iterations; \( C_1 \) and \( C_2 \) are non-negative coefficients of self and social recognition, defined as follows:

\[
C_1 = 1.8 \cdot b + 2.1,
\]  

(27)

\[
C_2 = 1.8 \cdot a + 0.1.
\]  

(28)

The value of \( C_1 \) is greater than \( C_2 \), and their summation remains at 4 in all iterations \([36]\). In each iteration, the objective function is determined for each particle, and the smallest objective function is selected as \( p_{\text{best}} \). The value of \( S_{\text{best}} \) will be the lowest value of \( p_{\text{best}} \) when the algorithm ends. ASPO stops when the number of iterations meets the maximum number of iterations \( k_{\text{max}} \). In this paper, the numbers of iterations and particles are 150 and 30, respectively, which were determined by experimentation to obtain the convergence of the objective function values to zero. Algorithm 1 and Figure 7 illustrate the pseudo code and flow chart of the APSO algorithm, respectively, where \( K_{\text{step}} \) is a constant value for a step of \( K_u \).

---

**Figure 7.** Flow chart of algorithm combining Ziegler–Nichols (Z-N) and adaptive particle swarm optimization (APSO).
Algorithm 1 Combination of Ziegler–Nichols (Z-N) and adaptive particle swarm optimization (APSO).

1: Start
2: Set \( K_i = 0 \);
3: Set \( K_d = 0 \);
4: Set \( K_p = K_u \);
5: Initialize \( K_u \);
6: Set step value for \( K_u \)
7: while Observe oscillation do;
8: \( K_u = K_u + K_{step} \);
9: end while
10: Calculate frequency of the oscillation as \( T_u \);
11: Calculate \( K_p, K_i, \) and \( K_d \)
12: Initialize population around the output of Z-N results;
13: Evaluate initial population;
14: Selecting the initial \( p_{best} \);
15: while Number of iteration less than \( i_{max} \) do;
16: Create new iteration;
17: Select the \( p_{best} \);
18: if \( p_{best} \) less than the previous \( g_{best} \) then;
19: Store it as \( g_{best} \)
20: end if
21: end while
22: Select the particle of \( g_{best} \) as the result
23: End

Figure 8 shows the \( p_{best} \) in each iteration for three joints of the human exoskeleton.

![Figure 8](a) Hip (b) Knee (c) Ankle

Figure 8. \( p_{best} \) in each iteration.
The changes in $C_1$ and $C_2$ for 150 iterations are shown in Figure 9 and the $\omega$ in each iteration.

![Figure 9](image)

**Figure 9.** Changes in adaptive particle swarm optimization (APSO) gains.

Table 3 demonstrates the optimal gains of the PID controller.

| Joints | $K_p$  | $K_i$   | $K_d$  |
|--------|--------|---------|--------|
| Hip    | 1656   | 4731.41 | 24.15  |
| Knee   | 1314   | 3864.62 | 0.5    |
| Ankle  | 101.90 | 615.57  | 4.26   |

### 6. Results and Discussion

The desired trajectory of each joint is based on gait training to move the joints simultaneously. The trajectories of each joint are driven by two different functions, repeated periodically. Table 4 shows the desired trajectory.

| Joints | $0 < t < \frac{t'}{2}$ | $\frac{t'}{2} \leq t < t'$ |
|--------|------------------------|------------------------|
| Right hip | $-0.24\sin(\pi t)$ | $0.4\sin(\pi t)$ |
| Right knee | 0 | $-0.48\sin(\pi t)$ |
| Right ankle | $-0.04\sin(\pi t)$ | $0.4\sin(\pi t)$ |
| Left hip | $0.4\sin(\pi t)$ | $-0.24\sin(\pi t)$ |
| Left knee | $-0.48\sin(\pi t)$ | 0 |
| Left ankle | $0.4\sin(\pi t)$ | $-0.04\sin(\pi t)$ |

$t$ represents the elapsed time and $t'$ is the time of one walking cycle. Figure 10 provides the frame-by-frame pictures of gait training with parallel bars.

The APSO algorithm was coded in Python programming language, and the virtual environment was Gazebo, integrated with ROS. All tests were performed using a PC with a Ubuntu 18.04 LTS operation system with 16 GB RAM and an Intel core-i3 Xeon processor. Figure 11 exhibits the performance of the simulated model in Gazebo. The graphs compare the trajectories of Z-N, PSO, and APSO with the desired trajectory. The Z-N measured the initial values for PSO and APSO. The difference between PSO and APSO is the method of setting the parameters. In PSO, $\omega$, $C_1$, and $C_2$ are adjusted to 1.2, 2, and 2, respectively.

Figure 12 shows the angular error trajectory for the hip, knee, and ankle. For all joints, the APSO error was the lowest, followed by PSO and Z-N. For example, the average error determined by APSO for the hip was 24% and 48% is less than PSO and Z-N, respectively. APSO showed less overshoot than the other methods because of its smaller peak, whereas PSO and Z-N experienced more shock when the trajectory changed.

Figure 13 demonstrates the output torque of each link. The maximum torques of the knee and ankle were less than 3 and 0.05 Nm, respectively. The maximum amplitude of torque for the hip was
7 Nm, which is the highest because the actuator of the hip tolerates a heavier mass compared to the knee and ankle.

Figure 10. Frames of human model gait training with parallel bars.

Figure 11. Cont.
Figure 11. Trajectory angle of each joint.

Figure 12. Cont.
Figure 12. Angular trajectory error for each joint.

Figure 13. Torque for each link of the human exoskeleton model.

Table 5 provides the numerical analysis of Z-N and APSO performance.

Table 5. Numerical analysis.

| Joint   | Left   | Right   |
|---------|--------|---------|
|        | ME     | AE      | RMSE   | $R^2$ | ME     | AE      | RMS    | $R^2$ |
| Hip    | 0.0285 | 0.0080  | 0.0096 | 0.9983 | 0.0184 | 0.0077  | 0.0096 | 0.9983 |
| Knee   | 0.0092 | 0.0021  | 0.0024 | 0.9998 | 0.0037 | 0.0019  | 0.0022 | 0.9998 |
| ankle  | 0.023  | 0.0042  | 0.0047 | 0.9991 | 0.0088 | 0.0044  | 0.0049 | 0.9990 |
ME, AE, RMSE, and $R^2$ represent the maximum error, average error, root mean square error, and coefficient of determination, respectively. The maxima of ME and AE were less than 0.03 m, which are within the acceptable zone [37]. The maximum of RMSE was 0.0130, while the $R^2$ values of all joints were greater than 99%, indicating the close fitted of all variability in the actual trajectories.

7. Conclusions

In this study, a PID controller was optimally tuned using an optimization algorithm that combines of Z-N and APSO. Parameter changes in each iteration enabled the finding of the optimal solution and avoided the local optima. Furthermore, the PID gains were established by Z-N and used for initializing the APSO. The proposed method was compared and tested in a virtual environment on a human exoskeleton model. The results demonstrated the acceptable performance of the APSO controller by statistical analysis.

The optimal PID controller minimized the trajectory error of a multi-joint human exoskeleton structure for gait training. For a new trajectory and different training conditions, the optimization process should be repeated, which is the limitation of this study. In future work, the model can be applied in different situations such as climbing stairs and the stand-to-sit movement.

Author Contributions: Conceptualization, M.S.A. and R.R.; methodology, M.S.A. and R.R.; software, M.S.A. and M.F.I.; validation, M.S.A. and N.A. formal analysis, M.S.A. and R.R.; investigation, M.S.A., R.R., and N.A.; resources, M.S.A.; data curation, M.S.A., R.R., M.F.I., and N.A.; writing—original draft preparation, M.S.A.; writing—review and editing, M.S.A., R.R., M.F.I., and N.A.; visualization, M.S.A.; supervision, R.R. and M.F.I.; project administration, R.R. and M.F.I.; funding acquisition, R.R. and D.A.W. All authors have read and agreed to the published version of the manuscript.

Funding: The authors would like to thank Universiti Kebangsaan Malaysia (UKM) and the Ministry of Education Malaysia for the financial support received under research grants GP-2020-K014062 and KK-2020-014.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Wang, S.; Wang, L.; Meijneke, C.; Van Asseldonk, E.; Hoellinger, T.; Cheron, G.; Ivanenko, Y.; La Scaleia, V.; Sylos-Labini, F.; Molinari, M.; et al. Design and Control of the MINDWALKER Exoskeleton. IEEE Trans. Neural Syst. Rehabil. Eng. 2015, 23, 277–286. [CrossRef] [PubMed]

2. Zhou, L.; Chen, W.; Chen, W.; Bai, S.; Zhang, J.; Wang, J. Design of a passive lower limb exoskeleton for walking assistance with gravity compensation. Mech. Mach. Theory 2020, 150, 103840. [CrossRef]

3. Murray, S.A.; Ha, K.H.; Hartigan, C.; Goldfarb, M. An assistive control approach for a lower-limb exoskeleton to facilitate recovery of walking following stroke. IEEE Trans. Neural Syst. Rehabil. Eng. 2015, 23, 441–449. [CrossRef]

4. Chen, B.; Zhong, C.H.; Zhao, X.; Ma, H.; Guan, X.; Li, X.; Liang, F.Y.; Cheng, J.C.Y.; Qin, L.; Law, S.W.; et al. A wearable exoskeleton suit for motion assistance to paralysed patients. J. Orthop. Transl. 2017, 11, 7–18. [CrossRef]

5. Kermavnar, T.; O’Sullivan, K.J.; Casey, V.; de Eyto, A.; O’Sullivan, L.W. Circumferential tissue compression at the lower limb during walking, and its effect on discomfort, pain and tissue oxygenation: Application to soft exoskeleton design. Appl. Ergon. 2020, 86, 103093. [CrossRef] [PubMed]

6. Huo, W.; Mohammed, S.; Amirat, Y.; Kong, K. Active Impedance Control of a lower limb exoskeleton to assist sit-to-stand movement. In Proceedings of the IEEE International Conference on Robotics and Automation, Stockholm, Sweden, 16–21 May 2016; pp. 3530–3536. [CrossRef]

7. Daachi, M.E.; Madani, T.; Daachi, B.; Djouani, K. A radial basis function neural network adaptive controller to drive a powered lower limb knee joint orthosis. Appl. Soft Comput. J. 2015, 34, 324–336. [CrossRef]

8. Chen, S.; Chen, Z.; Yao, B.; Zhu, X.; Zhu, S.; Wang, Q.; Song, Y. Adaptive Robust Cascade Force Control of 1-DOF Hydraulic Exoskeleton for Human Performance Augmentation. IEEE/ASME Trans. Mechatron. 2017, 22, 589–600. [CrossRef]

9. Lu, R.; Li, Z.; Su, C.Y.; Xue, A. Development and learning control of a human limb with a rehabilitation exoskeleton. IEEE Trans. Ind. Electron. 2014, 61, 3776–3785. [CrossRef]
10. Yang, Y.; Huang, D.; Dong, X. Robust repetitive learning control of lower limb exoskeleton with hybrid electro-hydraulic system. In Proceedings of the 7th Data Driven Control and Learning Systems Conference, Enshi, China, 25–27 May 2018; pp. 718–723. [CrossRef]

11. Merayo, N.; Juárez, D.; Aguado, J.C.; De Miguel, I.; Durán, R.; Fernández, P.; Lorenzo, R.M.; Abril, E. PID Controller Based on a Self-Adaptive Neural Network to Ensure QoS Bandwidth Requirements in Passive Optical Networks. J. Opt. Commun. Netw. 2017, 9, 433. [CrossRef]

12. Castillo-Zamora, J.; Camarillo-Gomez, K.A.; Pérez-Soto, G.I.; Rodríguez-Reséndiz, J. Comparison of PD, PID and Sliding-Mode Position Controllers for V-Tail Quadcopter Stability. IEEE Access 2018, 6, 38086–38096. [CrossRef]

13. Pan, Z.; Dong, F.; Zhao, J.; Wang, L.; Wang, H.; Feng, Y. Combined Resonant Controller and Two-Degree-of-Freedom PID Controller for PMSLM Current Harmonics Suppression. IEEE Trans. Ind. Electron. 2018, 65, 7558–7568. [CrossRef]

14. Lee, K.Y.; Park, J. Application of Particle Swarm Optimization to Economic Dispatch Problem: Advantages and Disadvantages. In Proceedings of the IEEE PES Power Systems Conference and Exposition, Atlanta, GA, USA, 29 October–1 November 2006; pp. 188–192. [CrossRef]

15. Abdmouleh, Z.; Gastli, A.; Ben-brahim, L.; Haouari, M. Review of optimization techniques applied for the integration of distributed generation from renewable energy sources Review of optimization techniques applied for the integration of distributed generation from renewable energy sources. Renew. Energy 2017, 113, 266–280. [CrossRef]

16. Belkadi, A.; Oulhadj, H.; Touati, Y.; Khan, S.A.; Daachi, B. On the robust PID adaptive controller for exoskeletons: A particle swarm optimization based approach. Appl. Soft Comput. 2017, 60, 87–100. [CrossRef]

17. Durgadevi, S.; Sundari, K.T.; Raaghai, D.; Akshaya, R. Comparative study of Controller Optimisation for CSTR using Particle Swarm Optimization Technique. In Proceedings of the 5th International Conference on Electrical Energy Systems (ICEES), Chennai, India, 21–22 February 2019; pp. 2–7. [CrossRef]

18. Sadigh, R.S.M. Optimizing PID Controller Coefficients Using Fractional Order Based on Intelligent Optimization Algorithms for Quadcopter. In Proceedings of the 6th RSI International Conference on Robotics and Mechatronics (IcRoM), Tehran, Iran, 23–25 October 2018; pp. 146–151. [CrossRef]

19. Zhai, Q.; Xia, X.; Feng, S.; Huang, M. Optimization design of LQR controller based on improved whale optimization algorithm. In Proceedings of the 3rd International Conference on Information and Computer Technologies (ICICT), Maui Island, HI, USA, 11–14 March 2020; pp. 380–384. [CrossRef]

20. Pan, D.; Gao, F.; Miao, Y.; Cao, R. Co-simulation research of a novel exoskeleton-human robot system on humanoid gait with fuzzy-PID/PID algorithms. Adv. Eng. Softw. 2015, 79, 36–46. [CrossRef]

21. Moharam, A.; El-Hosseini, M.A.; Ali, H.A. Design of optimal PID controller using hybrid differential evolution and particle swarm optimization with an aging leader and challengers. Appl. Soft Comput. 2016, 38, 727–737. [CrossRef]

22. Lal, D.K.; Barisal, A.K.; Tripathy, M. Load Frequency Control of Multi Area Interconnected Microgrid Power System using Grasshopper Optimization Algorithm Optimized Fuzzy PID Controller. In Proceedings of the IEEE International Conference on Recent Advances on Engineering, Technology and Computational Sciences (RAETCS), Allahabad, India, 6–8 February 2018; pp. 1–6. doi:10.1109/RAETCS.2018.8443847. [CrossRef]

23. Amiri, M.S.; Ramli, R.; Tarmizi, M.A.A.; Ibrahim, M.F.; Danesh Narooei, K. Simulation and Control of a Six Degree of Freedom Lower Limb Exoskeleton. J. Kejuruter. 2020, 32, 197–204. [CrossRef]

24. Amiri, M.S.; Ramli, R.; Ibrahim, M.F. Hybrid design of PID controller for four DoF lower limb exoskeleton. Appl. Math. Model. 2019, 72, 17–27. [CrossRef]

25. Amiri, M.S.; Ramli, R.; Ibrahim, M.F. Genetically optimized parameter estimation of mathematical model for multi-joints hip–knee exoskeleton. Robot. Auton. Syst. 2020, 125, 103425. [CrossRef]

26. Taylor Dempster, W. Space Requirements of the Seated Operator, Geometrical, Kinematic, and Mechanical Aspects of the Body with Special Reference to the Limb; Wright-Patterson Air Force Base: Green, OH, USA, 1955.

27. Tözeren, A. Human Body Dynamics: Classical Mechanics and Human Movement; Springer Science & Business Media: Berlin, Germany, 1999.

28. De Leva, P. Adjustments to Zatsiorsky-Seluyanov’s segment inertia parameters. J. Biomech. 1996, 29, 1223–1230. [CrossRef]
29. Amiri, M.S.; Ramli, R.; Ibrahim, M.F. Initialized Model Reference Adaptive Control for Lower Limb Exoskeleton. *IEEE Access* **2019**, *7*, 167210–167220. [CrossRef]

30. Wang, T.; Chang, C. Hybrid Fuzzy PID Controller Design for a Mobile Robot. In *Proceedings of the IEEE International Conference on Applied System Invention (ICASI)*, Chiba, Japan, 13–17 April 2018; pp. 650–653. [CrossRef]

31. Diaz-Rodríguez, I.D.; Han, S.; Keel, L.; Bhattacharyya, S. Advanced Tuning for Ziegler-Nichols Plants. *IFAC-PapersOnLine* **2017**, *50*, 1805–1810. doi:10.1016/j.ifacol.2017.08. [CrossRef]

32. Munlin, M.; Anantathanavit, M. New Social-Based Radius Particle Swarm Optimization. In *Proceedings of the IEEE Conference on Industrial Electronics and Applications (ICIEA)*, Siem Reap, Cambodia, 18–20 June 2017; pp. 838–843. doi:10.1109/ICIEA.2017.8282956. [CrossRef]

33. Eberhart, R.; Kennedy, J. A new optimizer using particle swarm theory. In *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, Nagoya, Japan, 4–6 October 1995; pp. 39–43. [CrossRef]

34. Yusuf, L.A.; Magaji, N. GA-PID controller for position control of inverted pendulum. In *Proceedings of the IEEE International Conference on Adaptive Science and Technology (ICAST)*, Ota, Nigeria, 29–31 October 2015; pp. 1–5. [CrossRef]

35. Amiri, M.S.; Ibrahim, M.F.; Ramli, R. Optimal parameter estimation for a DC motor using genetic algorithm. *Int. J. Power Electron. Drive Syst. (IJPEDS)* **2020**, *11*, 1047–1054. [CrossRef]

36. Sumathi, S.; Paneerselvam, S. *Computational Intelligence Paradigms Theory and Applications*, 1st ed.; CRC Press: Boca Raton, FL, USA, 2010. [CrossRef]

37. Wu, J.; Gao, J.; Song, R.; Li, R.; Li, Y.; Jiang, L. The design and control of a 3DOF lower limb rehabilitation robot. *Mechatronics* **2016**, *33*, 13–22. [CrossRef]

**Publisher’s Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).