Genetically Optimized ANFIS-based PID Controller Design for Posture-Stabilization of Self-Balancing-Robots under Depleting Battery Conditions

Omer Saleem[1], Mohsin Rizwan[2], Panos S. Shiokolas[3], Babar Ali[4]

[1]*Department of Electrical Engineering, National University of Computer and Emerging Sciences, Lahore, Pakistan (e-mail: omer.saleem@nu.edu.pk)*
[2]*Department of Mechatronics and Control Engineering, University of Engineering and Technology, Lahore, Pakistan (e-mail: mohsin.rizwan@uet.edu.pk)*
[3]*Department of Mechanical and Aerospace Engineering, University of Texas at Arlington, Arlington, TX, 76010, USA (e-mail: shiokolas@uta.edu)*
[4]*School of Engineering and Applied Sciences, University of Sussex, East Sussex, England (email: bali@sussex.ac.uk)*

Abstract: It is well known that in battery-powered control applications, the continuous drop in battery's output-power progressively degrades the dynamic performance of the system. The battery depletion phenomenon deteriorates the reliability of correctional effort if the controller gains are not adaptively adjusted as function of the available power level. Hence, this paper presents an adaptive neuro-fuzzy inference system (ANFIS) that dynamically adjusts the controller gains of a close-loop dynamic system as function of battery power-level in order to maintain desired performance while the battery is depleting. The proposed methodology is verified on an inherently unstable two-wheeled self-balancing-robot. The Proporional-Integral-Derivative (PID) controller is used for robot's posture-stabilization. İnitially, trivial sets of PID gains are selected via genetic algorithm to yield best control effort at various battery power-levels, using hardware-in-the-loop strategy. The acquired data is then used to train a power-level dependedent ANFIS that dynamically adjusts the PID gains in real-time. The performance of a fixed gain PID controller is compared with that of the proposed self-tuning PID controller for two different power-depletion scenarios that emulate real-world situations. The corresponding experimental results validate the robustness of the proposed control scheme to maintain the robot's postural stability under discharging battery condition.

Keywords: Battery power depletion, ANFIS, self-balancing-robot, PID controller, genetic optimization.

1. INTRODUCTION

The Direct-Current (DC) batteries are used to supply electric power at places where conventional alternating-current (AC) sources are not available or viable. They provide a reliable source of electrical energy when they are fully, or reasonably, charged. However, as the battery discharges, the available power supply capacity reduces which adversely affects the performance of the device being powered. The power depletion phenomenon is extremely detrimental to the battery-powered devices such as hybrid electric vehicles, drones, wheeled mobile robots, medical assistive devices, etc. Therefore, the knowledge of battery power depletion-rate, and its effect on the performance of the system being powered must be considered as an integral component of remedial actions in order to ensure that the system maintains the desired performance. A preliminary study performed by the authors clearly demonstrated that the available power level for control purposes affects the performance of the system. The experiments were performed on a Two-Wheeled-Self-Balancing-Robot (TWSBR) with fixed Proportional-Integral-Derivative (PID) controller gains defined for full power availability of the battery. The responses corresponding to the variations in body-angle of TWSBR for three different battery power-levels (15 W, 30 W, 50 W) are presented in Fig. 1. The responses clearly show that for certain power levels (in this case, 30 W) the system could be controlled but not within desired performance specifications. Furthermore, there exists a threshold power level (in this case, 15 W) below which the system cannot maintain the desired performance and becomes unstable.

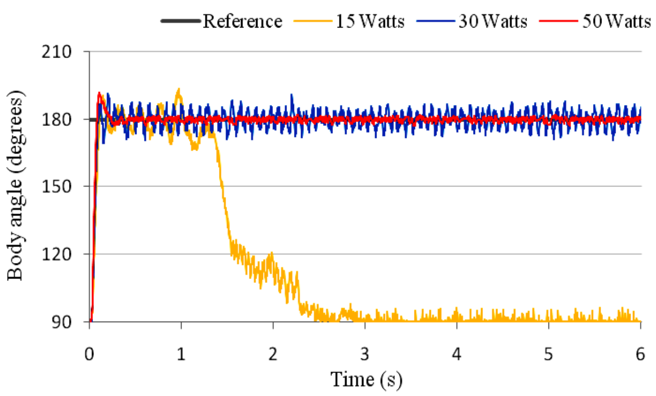

Fig. 1. Response of TWSBR with drop in motor input power with fixed PID gains

2.1 Related work

Extensive research has been performed related to battery power management and relevant control strategies in dynamic systems. The research literature addresses topics such as development of mathematical models for battery behavior, energy management in hybrid vehicles, energy management in wind turbines, optimization of battery - super capacitor storage system, optimal energy management for an aging battery, and risk management strategies for a depleting battery. An empirical model of battery behavior during discharge cycles was developed in (Saha and Goebel, 2009; Dalal et al., 2011). A review article discussed optimal energy management strategies considering fuel consumption, emissions and economy (Panday and Bansal, 2014). A particle swarm based optimal energy management for hybrid wind-micro-turbines with multiple constraints is discussed in (Pourmousaviv et al., 2010). Control strategies for plug-in hybrid vehicles are discussed in (Wirasingha and Emadi, 2011). Reduction of investment and operating cost in hybrid energy storage station is discussed in (Zhou and Sun, 2014). Improved annealing particle swarm optimization is used to optimize the problem with constraints. Optimal energy storage control strategy for grid connected micro-grids is discussed in (Malysz et al., 2014). Power management for an aging battery used Pontryagin's minimum principle to find a solution with a compromise between aging and performance (Serrao et al., 2011). Energy management for plug-in hybrid electric vehicle using a combined "rule based and particle swarm optimization algorithm" showed improvements in energy savings compared to traditional blended strategy (Chen et al., 2015). Risk-gain based energy management for self-docking mobile robots where the battery depletion is considered as a risk and accomplishment of task as a gain and an arbitrator makes the final decision based on the assessment of risk and the level of associated accomplishment (Berenz and Suzuki, 2011).

The available literature mainly discusses energy management strategies for either overall improved performance or a favourable compromise between risk and gain while the battery is depleting. In open literature, the dynamic adjustment of PID gains as function of available power level, as the battery is discharging, is not discussed. However, the aforementioned idea is the main focus of this research article.

2.2 Proposed approach

In this research, the PID controller serves as TWSBR's primary stabilization controller. Tuning the PID gains, K_p, K_i and K_d, is essential to maintain desired system performance. A number of approaches have been suggested for tuning PID gains for various applications. An artificial neural network was used to tune PID gains for load frequency control of power systems (Mossad and Salem, 2014). PID gains were tuned for missile altitude control using genetic algorithm and the results compared with classical Ziegler-Nichols method (Mantri and Kulkarni, 2013). PID gains were tuned using genetic algorithms for stripping section of distillation column and results compared with classical methods (Yusoff et al., 2015). The PID gains for a ball and hoop system were tuned using particle swarm optimization, genetic algorithms (GAs) and bacterial foraging optimization (Pareek at al., 2014). PID-type fuzzy logic controller was tuned using GAs for electronic throttle valves (Gritli et al., 2016). PID controllers are tuned for first order plus dead time system using genetic algorithms and the results are compared with classical tuning techniques on basis of Integral Squared Error and Integral Time Absolute Error (Hussain et al., 2014). PID tuning for a three tank process using genetic algorithm and comparison of results compared with classical tuning approaches is discussed in (Srinivas et al., 2014). As discussed in the open literature, GA either alone or combined with other techniques is used to tune PID controller gains (Gurban et al., 2014; Vijayakumar and Manigandan, 2016). The GA is preferred over classical tuning methods, such as Ziegler-Nichols or Cohen-Coon, because it stochastically evolves and minimizes the objective function to quickly converge to optimum values of K_p, K_i and K_d (Jaen-Cuellar et al., 2013; Lin and Liu. 2010). Extensive research has been done to validate the usage of genetic algorithms for the optimization of robot motions controllers (Halal and Dumitrache, 2006; Stan et al., 2007; Elbori el al., 2018).

The GA is a stochastic search and optimization algorithm based on natural evolution (Tam et al., 2018; Silva et al., 2004). The algorithm evaluates the fitness of individuals from a given population, selects the individuals with highest fitness, and combines (or mutates) them to generate a new set of individuals with improved fitness. This process is repeated iteratively until the termination criterion is satisfied. The GA neither relies on the system model nor on the derivative of the objective function. It can optimize multiple parameters. Owing to its aforementioned attributes, the GA is used to optimally estimate the PID controller gains for discrete levels of available battery power. The acquired reference data is then used to train an online power-dependent gain adjustment law that modifies the controller gains to maintain the desired system performance, even if battery is continuously discharging.

The intelligent control systems are widely used to compensate intrinsic and un-modeled nonlinearities associated with higher-order complex dynamical systems (Chopra et al., 2014). The conventional Fuzzy-Inference-System (FIS) utilizes a two-dimensional set of qualitative logical rules to infer correct control decisions. However, the heuristic development of a finite number of rules is not sufficient to address the unprecedented parametric variations. The Artificial-Neural-Networks (ANNs), on the other hand, have an inherent learning capability that enables them to derive an optimal nonlinear input-output relationship. However, its synthesis requires large sets of training data. Recently, the Adaptive-Neuro-Fuzzy-Inference-Systems (ANFIS) have gained a lot of momentum (Ioanaş, 2012; Cărbureanu, 2014). The ANFIS synergistically combines the learning capability of ANNs with reasoning-based inference of FIS to deliver robust adaptive control effort. The ANN estimation capability is utilized to automatically tune (and update) the parameters of the rule-based FIS to enhance the overall system performance and its robustness against exogenous disturbances (Szymak, 2016). Consequently, the performance of a pre-defined FIS is further improved via ANN-based training. This collaboration helps in quickly learning and developing an accurate numerical model to effectively control nonlinear and complex dynamical systems with minimal training data (Kumar et al., 2015).

The novel contribution of this research is to examine and adaptively compensate the effects of power depletion and remaining power on the performance of a dynamical system. For this purpose, an online gain adjustment mechanism is developed that adaptively modifies the gains of the ubiquitous PID controller online as function of available power level in order to maintain desired system performance. The developed methodology is successfully implemented and demonstrated on a TWSBR where the robot maintains its upright posture within desired performance specifications as function of the available battery power level.

The proposed methodology utilizes a well-postulated GA to generate reference data in order to train a dedicated ANFIS model that is then used to dynamically update the PID gains of TWSBR's stabilization controller, with respect to the variations in battery power conditions, after every sampling interval. This augmentation improves the controller's robustness by maintaining the robot's postural stability while rejecting the influence of continuously reducing battery power. The adaptation scheme operates concurrently with the control system. It recursively computes the error vector that comprises of the difference between the actual and desired system outputs. The ANFIS is trained off-line using the optimized reference data set generated by the GA. Once the ANFIS model is completely trained, the gains (K_p, K_i, K_d) are dynamically adjusted and applied to the PID controller. To conduct the hardware-in-the-loop experiments, the power variations observed during battery depletion is emulated via a DC-DC buck converter. However, the proposed adaptive controller can only effectively stabilize the posture of the TWSBR for a range of available battery power. The lower bound of this range, for this research, is 15.0 W.

The organization of the remaining paper is as follows. The dynamical model of the overall system is presented in Section 2. The primary feedback control scheme is discussed in Section 3. The training-data acquisition methodology is discussed in Section 4. Detailed synthesis of ANFIS model is presented in Section 5. The experimental setup is explained in Section 6. The experimental evaluation is conducted in Section 7. The paper is concluded in Section 8.

2. SYSTEM MODELING

The dynamic model of a TWSBR is similar to that of an inverted pendulum, which is modeled as a rigid pole that is fastened to a rigid cart via a frictionless joint. The cart moves along the longitudinal axis via two coaxial motorized wheels.

2.1. Actuator dynamics

The state-space model of a linear dynamical system is generally given by (1) and (2).

$$\dot{x} = Ax + Bu \tag{1}$$

$$y = Cx + Du \tag{2}$$

where, x is the state-vector, y is the output-vector, u is the control input signal, A is the system matrix, B is the input matrix, C is the output matrix, and D is the feed-forward matrix. The state-vector of a DC motor is given by (3).

$$x = [\varphi \quad \omega]^T \tag{3}$$

where, φ and ω is the angular position and velocity of the wheel, respectively. Correspondingly, the matrices A, B, C, and D in the motor's state-space model are identified in (4). The motor voltage (V_m) acts as the control input (u) of the motor.

$$A = \begin{bmatrix} 0 & 1 \\ 0 & \frac{{K_b}^2}{JR} \end{bmatrix}, B = \begin{bmatrix} 0 \\ \frac{K_b}{JR} \end{bmatrix}, C = [1 \quad 0], D = 0 \tag{4}$$

The motor parameters are experimentally identified in Table 1.

2.2. Wheel dynamics

All the forces acting on the robot body must be considered while deriving TWSBR's dynamic model. The free-body diagram of the right-wheel is shown in Fig. 2. The equations of motion of the right and left wheel are derived using the Newton's laws of motion. The dynamics of both wheels are analogous to each other. The wheel structure is assumed to be uniform and homogenous allowing it to roll over the horizontal surface without slippage. The combined effect contributed by the motors in moving the robot body is given in (5).

$$2\left(m_w + \frac{J_w}{r^2}\right)\ddot{x} = 2\left(\frac{K_b}{Rr}\right)V_m - 2\left(\frac{{K_b}^2}{Rr^2}\right)\dot{x} - (H_R + H_L) \tag{5}$$

where, m_w is the mass of the right-wheel, $\ddot{x}$ is the longitudinal acceleration of the right-wheel, $H_{R/L}$ is the z-axis force of the right/left-wheel with the robot's body, J_w is the moment-of-inertia of the right-wheel, τ_R is the wheel torque, H_{fR} is the frictional force, and r is the radius of the right-wheel. The parameters are identified in Table 1.

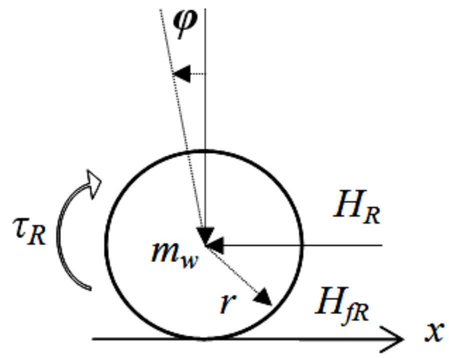

Fig. 2. Free body diagram of the right wheel

2.3. Upper body dynamics

The robot body being balanced by the wheels can be modelled as the arm of an inverted pendulum system. Its free-body diagram is shown in Fig. 3. The sum of horizontal forces acting on the body is given by (6).

$$\sum F_x = m_p\ddot{x} = H_R + H_L - m_p l\ddot{\theta}\cos\theta + m_p l\dot{\theta}^2\sin\theta \quad (6)$$

where, m_p is the mass of the robot chassis, θ is the angle subtended between the robot and the vertical-axis (also known as the pitch-angle), and l is the length of the centre-of-mass of robot's body from the ground. The linear equations of motion are given by (7) and (8).

$$\ddot{\theta} = \frac{m_p l}{(J_p + m_p l^2)}\ddot{x} + \frac{2{K_b}^2}{Rr(J_p + m_p l^2)}\dot{x} - \frac{2K_b}{R(J_p + m_p l^2)}V_m + \frac{m_p g l}{(J_p + m_p l^2)}\theta \quad (7)$$

$$\ddot{x} = \frac{2K_b}{Rr\left(2m_w + m_p + \frac{2J_w}{r^2}\right)}V_m - \frac{2{K_b}^2}{Rr^2\left(2m_w + m_p + \frac{2J_w}{r^2}\right)}\dot{x} + \frac{m_p l}{\left(2m_w + m_p + \frac{2J_w}{r^2}\right)}\ddot{\theta} \quad (8)$$

The continuous-time state-space representation of the two-wheeled self-balancing robot is given by (9) and (10), respectively.

$$\begin{bmatrix}\dot{x}\\ \ddot{x}\\ \dot{\theta}\\ \ddot{\theta}\end{bmatrix} = \begin{bmatrix}0 & 1 & 0 & 0\\ 0 & a_1 & a_2 & 0\\ 0 & 0 & 0 & 1\\ 0 & a_3 & a_4 & 0\end{bmatrix}\begin{bmatrix}x\\ \dot{x}\\ \theta\\ \dot{\theta}\end{bmatrix} + \begin{bmatrix}0\\ b_1\\ 0\\ b_2\end{bmatrix}V_m \quad (9)$$

$$y = \begin{bmatrix}0 & 0 & 1 & 0\end{bmatrix}\begin{bmatrix}x\\ \dot{x}\\ \theta\\ \dot{\theta}\end{bmatrix} \quad (10)$$

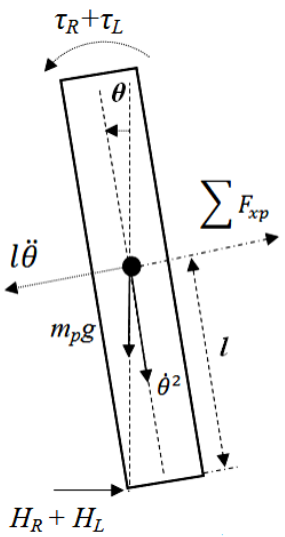

Fig. 3. Free body diagram of the robot body

where, $a_1 = \frac{2{K_b}^2(m_p lr - J_p - m_p l^2)}{Rr^2\gamma}, a_2 = \frac{{m_p}^2 g l^2}{\gamma}, a_3 = \frac{2{K_b}^2(J_p + m_p l^2 - m_p l)}{Rr^2\gamma}$,

$$a_4 = \frac{m_p g l\beta}{\gamma}, b_1 = \frac{2K_b(r\beta - m_p lr)}{Rr\gamma}, b_2 = \frac{2K_b(m_p l - r\beta)}{Rr\gamma},$$

$$\beta = 2m_w + m_p + \frac{2J_w}{r^2}, \gamma = J_p\beta + 2m_p l^2\left(m_w + \frac{J_w}{r^2}\right)$$

The system parameters are identified in Table 1.

2.4. LiPo battery discharge dynamics

In this research, the robotic system is powered using a Lithium Polymer (LiPo) battery. The nonlinear discharging behavior of the LiPo battery is discussed in (Kim et al., 2013). The circuit diagram of the LiPo battery cell is shown in Fig. 4. The piecewise nonlinear equations presented in (11) demonstrate the discharging behavior of a LiPo battery.

$$V_{terminal}(t) = \begin{cases} V_{oc}(t) - I(t)R_{dc} - I(t)R_s\left(1 - e^{-\frac{(T-t)}{R_s C_s}}\right), t \le T \\ V_{oc}(t) - I(t)(R_{dc} + R_s) - I(t)R_l\left(1 - e^{-\frac{(T-t)}{R_l C_l}}\right), t \le T - \tau_s \\ V_{oc}(t) - I(t)(R_{dc} + R_s + R_l), t \le T - \tau_s - \tau_l \end{cases} \quad (11)$$

such that, $T = m \times C_b, t = SOC(t) \times C_b, \tau_s = R_s C_s, \tau_l = R_l C_l s$

Table 1. System parameters of TWSBR

Parameter	Symbol	Value
Motor Stall Torque	T_{max}	0.784 Nm
Motor moment of inertia	J	
Motor Viscous friction coefficient	b	0.082 Nms/rad
Motor drive torque constant	K_b	0.073 Nm/A
Motor Back-EMF constant	K_t	0.073 V/(rad/s)
Motor's Armature resistance	R	3.3 Ω
Motor's Armature inductance	L	0.047 H
Motor's rated current	I_{rated}	5.0 A
Gravitational Acceleration	g	9.81 m/s^2
Wheel radius	r	0.121 m
Wheel mass (with motor)	m_w	0.258 kg
Wheel moment of inertia	J_w	1.87×10^{-3} kgm^2
Body center-of-mass height	l	0.158 m
Body mass	m_p	1.85 kg
Body moment of inertia	J_p	0.0247 kgm^2
Body dimensions	*Length × width × height*	0.2 m × 0.2 m × 0.33 m

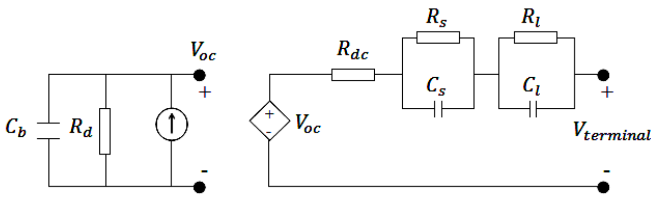

Fig. 4. LiPo battery cell circuit model

where, I is the current provided by the cell, SOC is the state-of-charge, and m is the initial SOC of the battery before discharge, C_b is the capacity of battery cell, R_d is the resistance of self-discharge, V_{oc} is the open-circuit voltage of the battery cell, R_{dc} is the equivalent DC-resistance of the cell, R_s and C_s is the resistance and capacitance of the short time-constant RC circuit in the model, respectively, R_l and C_l is the resistance and capacitance of the short time-constant RC circuit in the model, respectively.

When the discharging initiates, at $t = T$, the voltage appearing at the battery terminals ($V_{terminal}$) is lesser than the V_{oc} due to the voltage drop across R_{dc} as well as the exponential voltage drop across the short time-constant RC circuit for $t \leq T$. The voltage continues to drop for the entire duration of τ_s. Afterwards, the long time-constant RC circuit contributes in the exponential voltage-drop for $t \leq T - \tau_s$. It keeps dropping for the duration of τ_l. After this interval, the voltage drop becomes completely resistive and is significantly larger than the previous cases (Kim et al., 2013).

3. FEEDBACK CONTROL SYSTEM

This section presents a comprehensive overview of the feedback control architecture of the system. The performance of the conventional model-based controllers is prone to be degraded by modelling and identification errors associated with the system. Hence, in this research, the model-free PID control scheme is employed for posture stabilization of TWSBR (Li et al., 2013).

The effects of battery depletion on system response are evaluated via hardware-in-the-loop experiments using a TWSBR powered by LiPo battery driving two co-axial DC motors. As discussed earlier, the TWSBR works on the principle of inverted pendulum. If the system is inclined in one direction, the motors tend to move in the same direction to counter the effect. Attitude stabilization is achieved by monitoring and comparing the variations in the robot body angle with the reference angle using an Inertial-Measurement-Unit (IMU). The resulting errors are fed to the PID controller to appropriately drive the motors and keep the robot vertically balanced. Initially, the controller PID gains (K_p, K_i and K_d) are tuned assuming a fully charged battery. However, as the battery discharges, the DC input power to the motors drops. This reduction in the available power reduces the motor's speed and torque and as such its ability to deliver the required dynamic actuation effort. The body angle of the robot as function of time for different levels of available power is already shown in Fig. 1. At reduced power, the robot begins to vibrate vigorously about its reference position. Note that without adaptive control, these vibrations (and hence the error) tend to increase and the system eventually fails; the robot system used eventually collapses, as shown in Fig. 1, at a power level of ~15.0 W. The problem of performance degradation due to reduced available power could be addressed by dynamically scheduling the values of the PID gains according to the changes in the available battery power.

Owing to its simplicity and reliability, the PID controller is one of the most widely used control schemes (Astrom and Hagglund, 2016). In this research, it is used to control and regulate the pitch-angle (ϕ) of TWSBR's body within desired performance characteristics (Bhatti et al., 2015). The horizontal motion (or position) of TWSBR is not controlled in this research. PID controller is simply the weighted sum of error-dynamics occurring in the state-variable being controlled; namely, error, error-integral, and error-derivative (Bhatti et al., 2018). The error, $e_\theta(n)$, at sample time n is defined as the difference between the desired or set-point, $\theta(des, n)$, and actual, $\phi(n)$, states as shown in (12). The mathematical expression of the control law for hardware implementation is given by (13).

$$e_\theta(n) = \theta(des, n) - \phi(n) \tag{12}$$

$$u(n) = u(n-1) + K_p e_\theta(n) + K_d \frac{e_\theta(n) - e_\theta(n-1)}{T_s} + K_i \sum_{j=1}^{n} e_\theta(j) T_s \tag{13}$$

where $u(n)$ is the control signal at sample time n, and T_s is the sampling period. The diagram of the closed loop control scheme is shown in Fig. 5. Initially, the gains of the PID controller are tuned using GA assuming a fully charged battery and available maximum actuator/motor power level. As the system operates, the battery power level is reduced and the PID gains are optimally tuned via GA for a set of discrete levels of actuator or motor power to generate a reference data-set of gain values as function of available power level. In this research, 100 discrete power levels were selected. However, one could select a different number depending on the application and desired granularity for available power level. After setting the motor power at a certain level, the GA is initiated to identify the PID gains such that the system operates satisfactorily and remains within acceptable performance specifications, i.e. close to the desired vertical posture.

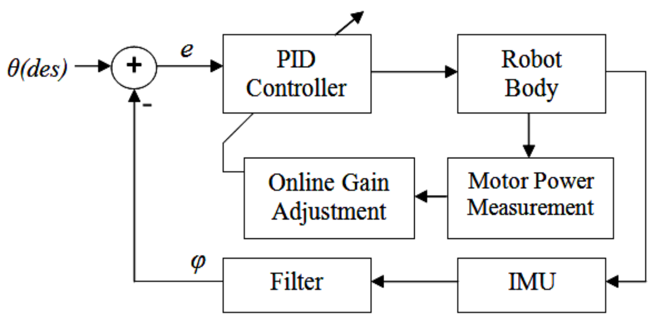

Fig. 5. Closed-loop feedback control scheme

Once the algorithm converges, the optimized PID gains for the given power level are recorded. The process of tuning the PID gains is repeated for each discrete power level. The PID gains recorded for each power level are then used to synthesize mathematical models for each gain as function of available power. The recorded data is used to train the ANFIS model that dynamically updates the PID gains to deliver appropriate control actions, *u(n)*, as function of available power level to maintain desired system performance.

4. TRAINING-DATA ACQUISITION USING GA

As discussed earlier, optimal sets of PID gains must be identified at discrete levels of battery power to derive the numerical model of the ANFIS-based online gain adjustment law as function of available power level. This section presents the methodology used to generate and acquire the reference data set for training the ANFIS gain scheduler using the genetic optimization algorithm.

The GA is a population-based stochastic search and optimization algorithm (Rout and Mittal, 2009; Petcut and Dragomir, 2010; Mohammed at al., 2014). It probabilistically explores a randomly selected population (or search-space) of candidate solutions to find the global-best solution. After the initialization of the search space, the algorithm encodes all potential solutions into strings of binary numbers, also referred to as "chromosomes", and evaluates their performance using an objective function that assigns a performance based fitness value to all the strings in a population. The highly fit strings are selected, paired and mutated to form a new population of candidate solutions. The parent strings mate with each other, via the process of "cross-over", to generate off-springs with relatively higher fitness. A few strings in the population are mutated to ensure a uniformity of fitness in the entire population and avoid the possibility of new stagnant or unsuitable populations (stuck in local minima regions). After every iteration of reproduction, the succeeding generations tend to evolve the search space with strings having improved fitness values. Thus, populating the space with high-ranking solutions enables the algorithm to converge quickly to the global-best solution. The process is repeated iteratively and is terminated only if either the desired performance criterion is achieved or the pre-defined maximum number of iterations is reached. The workflow of GA is shown in Fig. 6.

In this research, the GA generates an initial random population by encoding the PID gains into binary strings (or chromosomes). For a given power level, each trivial set of gain is fed to the PID controller and the corresponding body-angle response of the robot is recorded for 10 seconds (1000 samples). The resulting fitness of each population member is evaluated after decoding the binary string into real-valued PID gains and applying them to the actual hardware shown in Fig. 5. In this work, considering the objective of keeping the robot vertically balanced, the cost function is defined as the sum of squared-error within the defined time-span for every individual experiment.

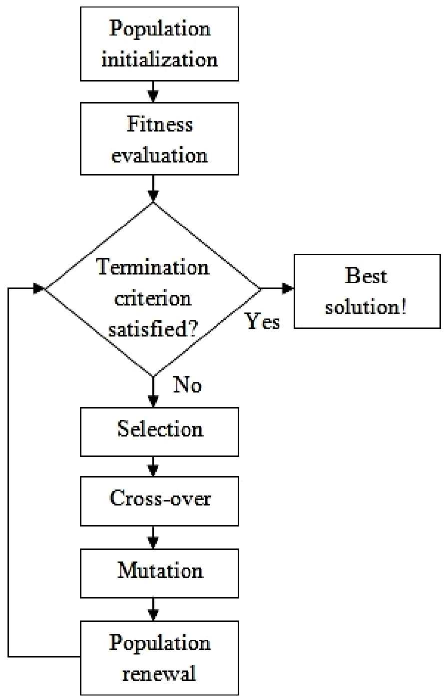

Fig. 6. Work-flow of GA procedure

This process is repeated until the defined number of generations is reached in an effort to reach near-optimal values of the PID gains meeting the desired performance specification of balancing the robot at a given power-level while minimizing the fitness function. The details and steps followed to tune the PID controller gains for each motor-power level are discussed as follows.

Initialization: An initial population of 50 strings is randomly generated to represent the three PID gains. The gains in the population are bounded within the interval 0 to 30 considering the convergence point in repeated trial. Each string is composed of 30 bits. The length of the string is divided in three sections; each containing a sub-string of 10 bits with each distinctly corresponding to one of the three PID gains.

Fitness evaluation: An objective function is required to evaluate the fitness and assess the performance of each string in the population. In this research, the objective function, J, to be minimized is defined as the sum-of-squared-error, given in (14). This function is selected as the criterion to evaluate the fitness of the PID gain strings.

$$\mathrm{J} = \sum_{n=1}^{N} (e_\theta(n))^2 \tag{14}$$

where, n is the sample number and N is defined as 1000 for all cases irrespective of how quickly the parameters converge. Every time a string is subjected to the objective function, the string is divided to extract the three PID gains which are directly applied to the robot's PID control law. The corresponding control commands are serially transmitted to the motor driver circuit of the robot. The variations in the error of the robot's body-angle are recorded to evaluate the fitness of the particular string.

Selection: The number of strings selected to form the subspace must be decided carefully; a large subspace may deliver a "better" solution but it would lead to a slower convergence rate, where, a smaller subspace would speed up the evolution process, but it may lead to premature convergence and possibly deliver bad solutions. In this work, 12 best (highly fit) strings are selected from the existing population and another 20 strings are randomly selected from the remaining 38 strings, thus generating a subspace of 32 strings as parents.

Crossover: The selected parent strings are then combined via the multi-point crossover operator, wherein the alternating segments of the corresponding subsections of parent strings are swapped to generate new off-springs. The crossover operator is applied with a probability of 0.7. Each new offspring is compared with the existing worst-string and if the off-spring has better fitness, then it is retained; otherwise, it is discarded.

Mutation: The evolved population is slightly mutated in order to prevent the algorithm from falling and possibly remaining in local minima regions and to improve the performance of the strings. The mutation operator is applied with a low probability of 0.063. One or more randomly selected bits of the chosen strings are flipped.

Termination: Once the new population is formed, it is analyzed for the termination criterion. The algorithm is only terminated when the difference between the fitness values of highest-ranked and the lowest-ranked strings in a population is less than or equal to the fitness limit (β), as shown in (15).

$$\left|J_{i,best} - J_{i,worst}\right| \leq \beta \qquad (15)$$

where, i is the current iteration (or generation) and J is the fitness value of a given string in a given population. In this research, the value of β is empirically selected as 0.1 to ensure convergence and to guarantee that all the candidate solutions are closely spaced. Once the proposed criterion is satisfied, the algorithm is terminated and the best string is selected as the output. Otherwise, the process is repeated until a maximum number of iterations are reached to avoid the possibility of non-convergence or an infinity loop of execution. The process of iterative genetic optimization of the PID gains (K_p, K_i, K_d) at a power level of 50.0 W is shown in Fig. 7. The training data of the K_p, K_i, and K_d generated by the GA at 100 discrete power levels is diagrammatically illustrated in Fig. 8, 9, and 10, respectively.

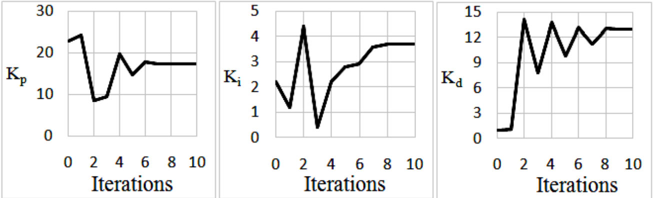

Fig. 7. Iteration history of PID gains at 50.0 W power level using GA

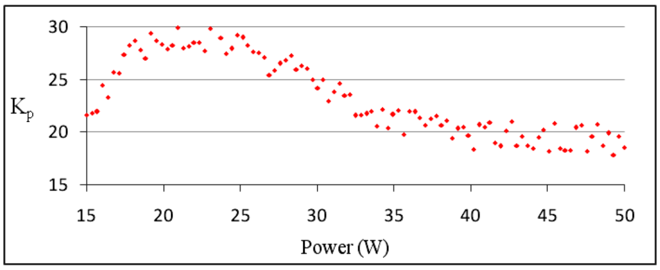

Fig. 8. Genetically optimized training data of K_p

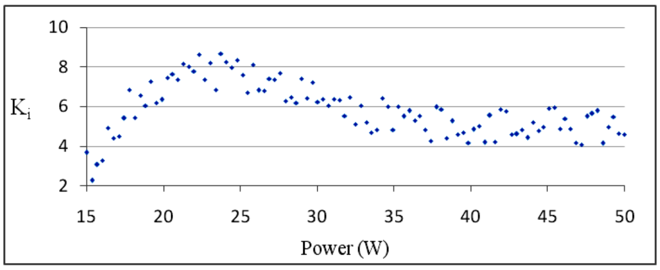

Fig. 9. Genetically optimized training data of K_i

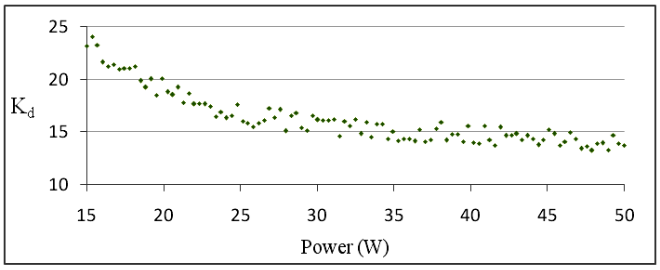

Fig. 10. Genetically optimized training data of K_d

5. PROPOSED ONLINE GAIN-ADJUSTMENT SCHEME

This section presents the functional description and synthesis of the proposed ANFIS based gain-scheduling mechanism. ANFIS is an adaptive realization of an FIS that utilizes a multi-layered feed-forward network (Toufouti et al., 2009). First of all, a two-dimensional table yielding the heuristically developed fuzzy rules is identified. The hypothesis of the parameterized rule-base serves to relate the fuzzified inputs to input Membership-Functions (MFs) to rules to outputs to output MFs. In the next phase, the optimized data set acquired by the genetic algorithm is applied to train the ANFIS. The trained ANFIS is then used to formulate the fuzzy inference mechanism by optimally adjusting its MFs such that they fulfill the desired control objective(s). The ANFIS model imitates a feed-forward back-propagation network. In this research, the neural part of the ANFIS is trained using the Hybrid-Learning-Algorithm (HLA). In the first phase of training, the HLA identifies the consequent parameters in the forward pass with the aid of Least-Squares method. In the second phase of training, the HLA it uses the error between the actual and desired system output as well as the gradient descent method to update the premise parameters in the backward pass. The training process terminates only when the maximum number of iterations is reached or the desired control objective is achieved**.** The fuzzy system is developed using the first-order Takagi-Sugeno inference model due to its high interpretability and computational efficiency.

The conventional five-layered architecture of ANFIS is shown in Fig. 11. The layers are denoted as the fuzzy layer, product layer, normalization layer, defuzzification layer, and total output layer. In this structure, the error in battery power (P_e) and the change in power-error (ΔP_e) are the inputs. The inputs are given by (16) and (17).

$$P_e(n) = P_{max} - P(n) \tag{16}$$

$$\Delta P_e(n) = P_e(n) - P_e(n-1) \tag{17}$$

where, P_{max} is equal to 50.0 W in this research, The vector containing PID gain updates, denoted as K_z, is the output of the proposed ANFIS model. The training values and the estimated values are represented by the input and output nodes, respectively. The nodes in the hidden layers operate as the MFs and rules. The nodes represented by a circle are fixed-nodes while the ones represented by a square shape are adaptive in nature. For a first-order Sugeno inference system, the governing rules are as follows:

Rule 1: If $g=P_e$ is A_1 and $h=\Delta P_e$ is B_1, then $K_{z1} = p_1g + q_1h + r_1$

Rule 2: If $g=P_e$ is A_2 and $h=\Delta P_e$ is B_2, then $K_{z2} = p_2g + q_2h + r_2$

where, A_j and B_j are the linguistic variables of the fuzzy rule-base, p_j, q_j, r_j are the consequent parameters, and j is the number of rule being considered (j = 1 or 2). The function of each layer is as follows (Saleem et al., 2018). The output of node i in layer l is denoted as O_i^l.

Layer 1 is the fuzzification layer. It fuzzifies the inputs g and h using the bell-shaped MFs due to their smoothness. The mathematical expression of MF is given by (18).

$$O_i^1 = \frac{1}{1 + \left|\frac{x-c_i}{a_i}\right|^{2b_i}} \tag{18}$$

where, a_i, and c_i are the width and the center of the MF, respectively, and b_i is used to control the slopes of the cross-over points. These premise parameters are dynamically adjusted via the ANN to yield optimum postural stability with minimum tracking deviations, even under depleting battery conditions. Every node in this layer is adaptive.

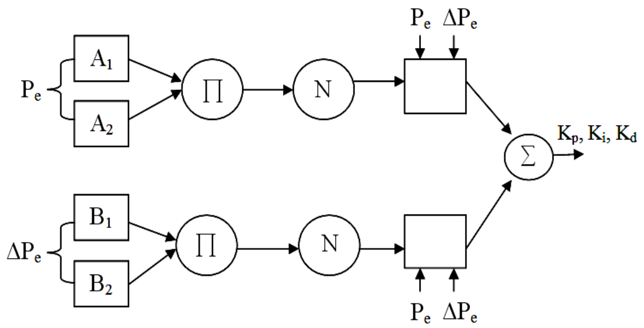

Fig. 11. The ANFIS structure

The layer evaluates the degree (μ) of input variables in the fuzzy set. The output of this layer is $\mu_{Aj}(g)$ or $\mu_{Bj}(h)$ and are given by (19) and (20).

$$O_i^1 = \mu_{Ai}(g), \qquad \text{such that } i = 1, 2 \tag{19}$$

$$O_i^1 = \mu_{Ai}(h), \qquad \text{such that } i = 3, 4 \tag{20}$$

The output of this layer forms the antecedents of 2nd layer. Once the consequent parameters are selected (see Layer 4), the approximate error is propagated back to every layer and the gradient-descent method is used to update the premise parameters.

Layer 2 is denoted as the product layer. It mathematically infers the updates required in the PID gains. Every node in this layer is fixed and corresponds to a fuzzy antecedent rule (the "*IF*" part). A total of 25 rules are used in this layer. The product T-norm aggregation operator is used with each fuzzy neuron. Hence, the output of this layer is the product of the input signals as shown in (21).

$$O_i^2 = w_i = \mu_{Ai}(g) \times \mu_{Bi}(h) \tag{21}$$

The resulting weightage, w, represents the firing-strength of each rule. The fuzzy MFs of the input-variables, P_e and ΔP_e, and output-variables, K_p, K_i, K_d, are linguistically defined as: Zero (Z), Small (S), Medium (M), Big (B), and Very Big (V). The two-dimensional rule-base is shown in Table 2.

Layer 3 is the normalization layer. Every node in this layer is fixed. This layer serves to normalize the firing strengths by evaluating the ratio of the i^{th} fuzzy-rule's firing strength to sum of all firing strengths of all the rules. The normalized firing strength is given by (22).

$$O_i^3 = \widehat{w}_i = \frac{\sum w_i}{\sum_i w_i} \tag{22}$$

Layer 4 is defined as the consequent layer. It is responsible for defuzzifying the consequent rules. The nodes in this layer are adaptive and they compute the values of i^{th} rule towards the overall output (the "*THEN*" part) using the linear combination shown in (23).

$$O_i^4 = s_i = \widehat{w}_i\,(p_ig + q_ih + r_i) \tag{23}$$

Table 2. Fuzzy rule base of PID controller gains

K_p,K_i,K_d		**ΔP_e**				
		Z	**S**	**M**	**B**	**V**
P_e	**Z**	Z,Z,Z	Z,S,S	S,S,M	S,Z,B	S,Z,V
	S	S,M,S	S,M,M	S,S,B	S,Z,V	M,Z,V
	M	M,M,M	M,B,M	M,B,B	M,S,V	M,Z,V
	B	B,S,M	B,S,M	B,M,B	B,S,V	B,Z,V
	V	V,S,M	V,S,M	V,M,B	V,S,V	V,Z,V

The consequent parameters are adaptively adjusted to yield best control effort, even under battery discharge conditions. Since O_i^4 is the linear combination of the consequent parameters, therefore, their optimal values are estimated via Least Squares method in the backward pass of the training phase.

Layer 5 is summation layer. The single node in this layer computes the crisp output value by aggregating all the incoming signals from Layer 4. The finalized output is given by (24).

$$O_i^5 = \sum_i s_i = \frac{\sum_i w_i \, (p_i g + q_i h + r_i)}{\sum_i w_i} \tag{24}$$

6. EXPERIMENTAL SETUP

In this section, the experimental setup used to verify the proposed methodology of considering power depletion and dynamic PID gain definition while maintaining desired performance is described. The TWSBR platform is shown in Fig. 12.

4.1. Hardware setup

The robot uses an 8-bit embedded microcontroller for monitoring and control tasks. An IMU (MPU-6050) is placed at the geometric centre of the robot's body (Liu and Dai, 2015). The IMU contains an accelerometer and a gyroscope that provide the body-angle and its rate-of-change, respectively. Two permanent magnet DC metal-geared motors are attached beneath the base of the robot structure to drive the wheels. Each motor has a 19:1 gear box. It is capable of delivering a torque of 0.588 Nm and a non-load speed of 500 rpm (Pololu motor item #1440). The motors are driven by a MC33926 dual H-bridge motor driver.

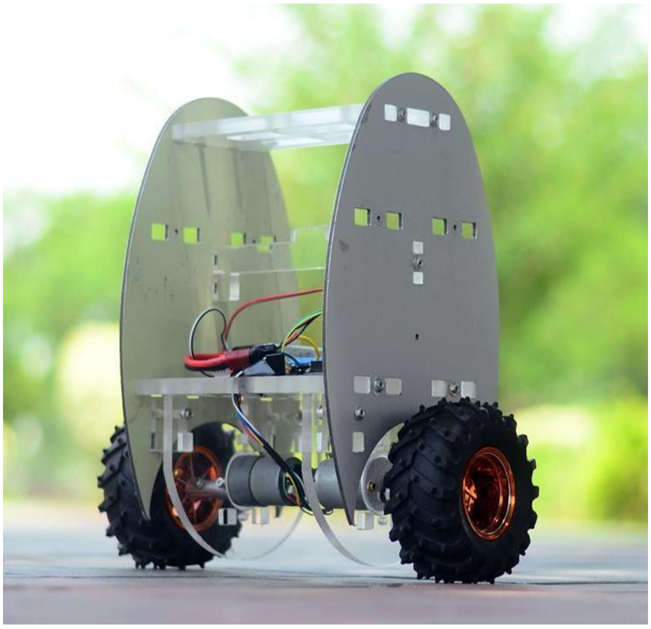

Fig. 12. Two-wheeled self-balancing robot

For experimentation purposes and to emulate *fast* battery power depletion, the input power to the motors is varied by controlling a 100 W programmable DC-DC step-down buck converter circuit. The wirelessly controlled circuit effectively alters the magnitude of the input power to the motors. Buck converters are energy-efficient switch-mode power supplies (Rashid, 2011). The block diagram of the power variation setup is shown in Fig. 13.

The MOSFET (Q) in the circuit is responsible for chopping down the DC supply voltage to the desired level. The inductor-capacitor filter removes the harmonics and improves the quality of waveform of the output-voltage. An analog voltage-divider, constructed with a 2.0 MΩ and a 1.0 MΩ, 0.5W resistors, is used to measure the instantaneous value of the voltage across each of the motors. The voltage division reduces the magnitude of V_{out} so that the resulting signal, V_s, can be safely interpreted by the 5.0 V microcontroller. The current consumption of the motors is measured with an ACS712 current sensor (Jamaluddin et al., 2013). The ANFIS model uses the instantaneous input power to the motors which is evaluated as the product of the analog current, I_s, and voltage, V_s, readings provided to the microcontroller.

The variation pattern in the reference voltage (and hence the power) is visualized using a graphical user interface developed in LabVIEW by National Instruments (http://www.ni.com) running on a remote computer. These commands are serially transmitted at 9600 bps to the on-board microcontroller over a wireless communication link established by a 2.4 GHz transceiver. Based on the commanded input, the microcontroller compares the reference voltage with the instantaneous voltage supplied to the motors (via the H-bridge motor driver) and generates appropriate pulse-width-modulated commands to the switching transistor in the circuit. The overall electronic hardware setup is shown in Fig. 14.

A 12.0V, 3300 mAh LiPo battery is used to supply DC power to all the modules of the robot via dedicated linear voltage-regulator circuits. When fully charged, the battery reads 12.6V across its terminals. The maximum power drawn by the motors is 50.0 W. Preliminary experiments indicate that below 15.0 W, the motors cannot provide the appropriate control torque to keep the robot upright (see Fig. 1).

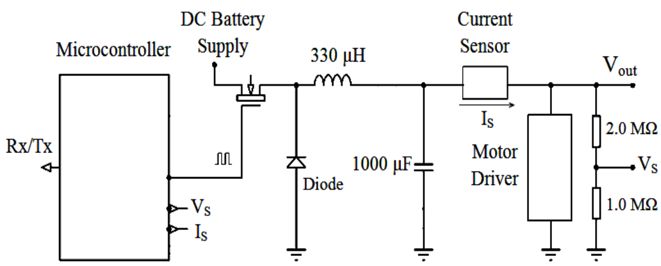

Fig. 13. Block diagram of power variation setup

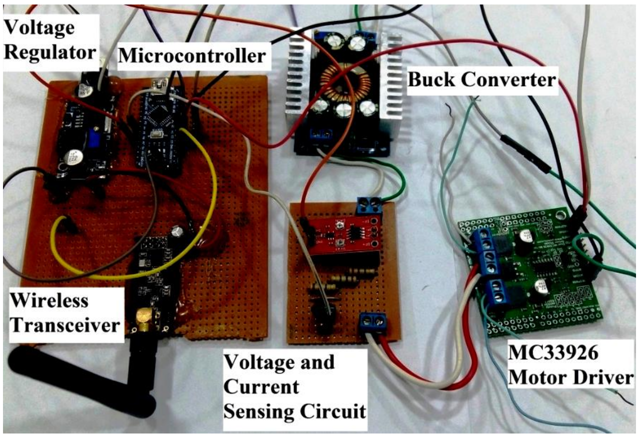

Fig. 14. Electronic hardware setup

4.2. Control software architecture

The control software architecture consists of the firmware program required to vertically balance the robot. When the system is initialized, a Kalman Filter uses the raw IMU and sensor readings and fuses the gyroscope and accelerometer readings provided by IMU to estimate the robot body-angle (Gui et al., 2015). The error between the actual and the desired body-angles is provided as input to the PID controller which controls the speed and direction of motor rotation. The sampling-time (T_s) for data acquisition and control updates is set at 10.0 ms. The ANFIS model is developed using the recorded dataset of the tuned PID gains. Once synthesized and connected to the PID balancing controller, the ANFIS begins to dynamically update the PID gains as functions of power variations.

7. EXPERIMENTAL EVALUATION

The perfromance of the proposed architecture of the ANFIS-based PID controller is compared with that of a fixed-gain PID controller. The gains of the fixed PID controller are defined at the maximum motor power level. According to Fig. 8, 9, and 10, the gains at 50.0 W are $K_p = 17.3$, $K_i = 3.7$, $K_d = 13.0$. For fixed-gain PID controllers, these gains are kept constant throughout the experiment. On the other hand, the gains of ANFIS-based PID controller are adaptively varied as functions of the available battery power. Two power depletion test scenarios are considered, continuous and step depletion, to study and compare the performance of each controller vertically balancing the robot with minimum deviations in the state-variable. The comparison of the robot's body-angle response is performed in terms of root-mean-square-error (RMSE) and maximum-absolute-error (MAE) in the body-angle. The experimental results and observations for each test are discussed.

5.1. Continuous power change

In this test, the motor input power is varied as shown in Fig. 15 using the wirelessly operated programmable buck converter circuit. The continuous power decay is selected to emulate the actual discharging pattern of a DC battery. The error in the robot body-angle from the vertical reference position, when the controller is used with fixed gains and with the proposed ANFIS model (variable gains) is shown in Fig. 16 and 17, respectively.

As the power level decreases, the error of the system without the ANFIS model significantly increases reaching an MAE of 14.8° and an overall RMSE of 6.8°. On the other hand, the controller effectively maintains the postural stability of the robot when augmented with the proposed ANFIS model, despite the continuous power decay. Even under low battery power conditions, the system exhibits an overall RMSE of only 1.5° and an MAE of 4.7° that is well within the desired angular deviation of ±5.0° in the robot body. The results clearly validate the efficacy of the proposed gain adjustment mechanism in maintaining the postural stability of the robot under continuous battery discharge conditions.

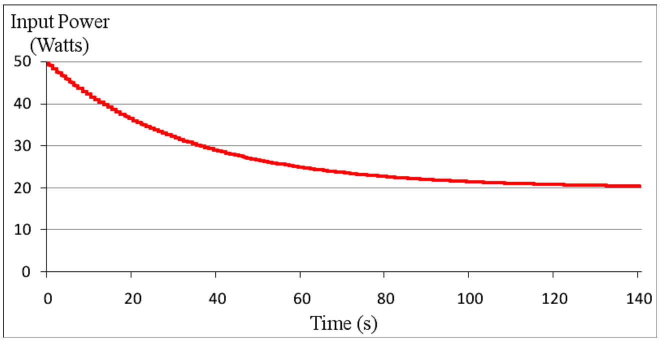

Fig. 15. Motor input power pattern for continuous power change

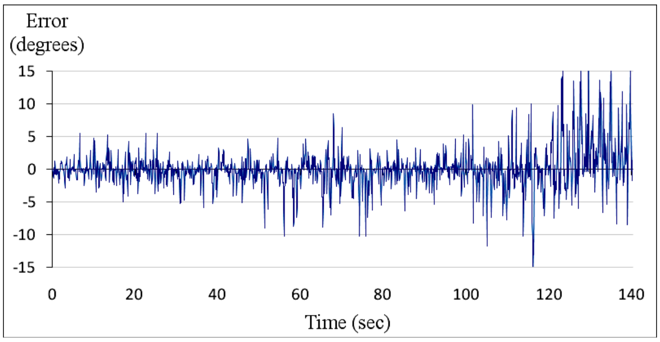

Fig. 16. Error in body-angle using generic PID controller for continuous power change

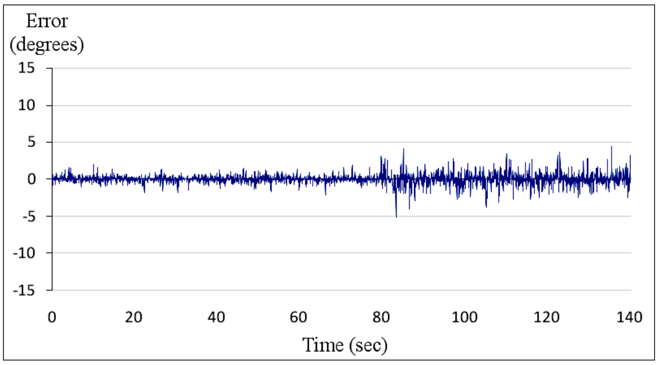

Fig. 17. Error in body-angle using the gain-scheduled PID controller for continuous power change

5.2. Step power change

In this test, the motor input power is varied in a step fashion as shown in Fig. 18, which is again accomplished by using the wirelessly operated programmable buck converter circuit. The power level is decreased in steps of 7.5 W at a regular interval of 10 sec. It is to noted that this power change is arbitrarily selected. The error in the robot's body-angle from the vertical reference position, when controlled without and with the proposed ANFIS gain adjustment mechansim is shown in Fig. 19 and 20, respectively.

The body-angle response of the robot with the fixed-gain PID controller begins nornally and is able to maintain the robot performance since full power is available. However, as the power level drops after fixed time intervals, the body-angle exhibits abrupt bounded variations. It is observed that the body-angle response does not converge within ±5.0º of the reference position after each step-change in power level. Instead, the magnitude of the error increases reaching an MAE of 14.8º and an overall RMSE of 9.1º. The performance of the proposed ANFIS model to dynamically define the PID gains as function of the available power level is also evaluated. The PID gains are defined when a power change occurs and remains fixed during that particular time interval. This system exhbibits small overshoots and undershoots at the instance of step-change with an MAE of 6.8º and an RMSE of 2.7º. It is observed that after each power step change, the response initially has relatively a 'large error'. However, this error settles to within ±1.0º of the desired performance within 2 to 3 seconds indicating that the proposed methodology can effectively compensate power transients as well. It is also observed that as the available power tends towards the limiting power level for sucessful control action (15.0 W for this system), the system exhibits a small increase in the magnitude of error compared to higher power levels, but, still remains within closer proximity of desired posture compared to the fixed gain system. This reinforces the premise of dynamically changing control gains as function of available power.

5.3. Discussion

The performances of the two PID controllers (fixed and adaptive) for each of the two testing scenarios of varying power level (uniform exponential decay and step) is summarized in Table 3. The ANFIS-based PID controller achieves considerably smaller MAE and RMSE values as compared to the conventional fixed-gain PID controller.

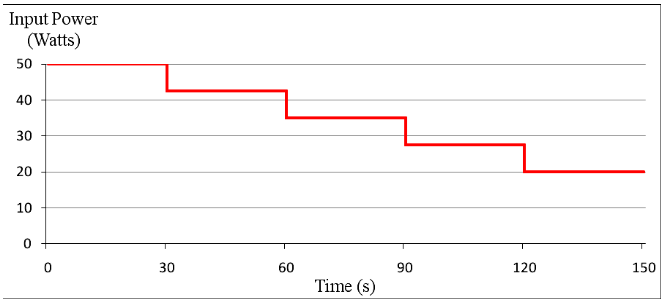

Fig. 18. Motor input power pattern for step power change

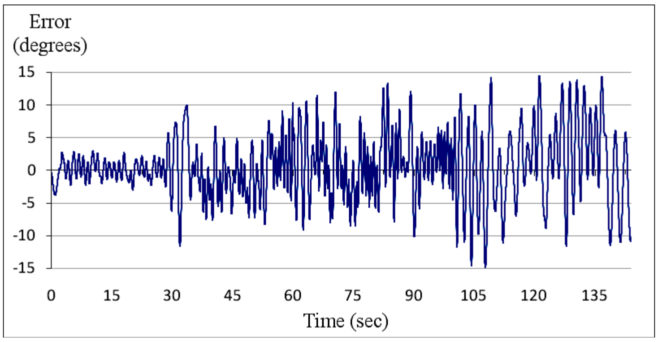

Fig. 19. Error in body-angle using generic PID controller for step power change

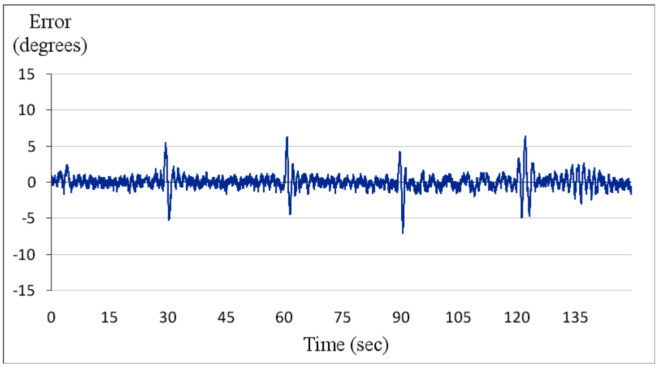

Fig. 20. Error in body-angle using the gain-scheduled PID controller for step power change

The average time taken by the robot body-angle to settle within ±2.0º of the reference position, after each step change in power-level, is denoted as T_s. The ANFIS-based PID controller exhibits a faster error convergence rate and the response settles quickly within ±2.0º of the reference after the introduction of step-change in power. The results demonstrate that the ANFIS model is effective in adaptively updating the PID gains of the controller and maintaining desired performance for both battery depletion scenarios which could be experienced in real-life situations. The comparative performance assessment clearly validates the superior robustness of the proposed control strategy in rejecting the exogenous disturbances in the form of battery depletion.

Table 3. Summary of performance comparison

Test	**PID Controller**	**MAE (º)**	**RMSE (º)**	**T_s (s)**
Exponential Power Change	Without ANFIS	14.8	6.8	-
	With ANFIS	4.7	1.5	-
Step Power Change	Without ANFIS	14.8	9.1	Does not settle
	With ANFIS	6.8	2.7	1.3

8. CONCLUSION

This manuscript presents a methodology to adaptively modify PID controller gains for a TWSBR as function of available power level to maintain desired system performance. The need for such a methodology emanates from the proliferation of battery powered devices such as drones, medical devices, etc. and the possible detrimental effects that power drainage might have on system performance or mission execution. An online gain adjustment mechanism that dynamically adjusts the PID controller gains in real-time was proposed, developed and evaluated on an inherently unstable dynamic system. The proposed technique utilizes GA to acquire reference data from real-time experiments at discrete power levels in order to train an ANFIS model that adaptively modifies the PID gains and applies the updated gains on the actual closed-loop system to observe the improvement rendered in the response. The performance of the adaptive PID controller was compared with a fixed-gain PID controller in terms of controlling and maintaining the vertical stability of a TWSBR as function of two different power depletion scenarios. The ANFIS-based adaptive controller was able to successfully maintain the desired system performance under both power depletion scenarios, where as the fixed-gain controller exhibited a degraded posture stabilization performance. The comparative performance assessment of the experimental results clearly validate the superiority of the closed-loop control system equipped with the proposed intelligent gain adjustment scheme in robustly controlling the battery-powered under-actuated robotic systems, even under the influence of uniformly or abruptly depleting battery power levels. In future, the efficacy of the proposed techniques can be further investigated by applying it to other under-actuated electro-mechanical systems. Other analytical and numerical inverse-problem approaches can also be investigated to develop a stable online adaptation mechanism for controller gains as function of battery power-level.

REFERENCES

Astrom, K.J., and Hagglund, T. (2016). *Advanced PID Control*. Instrumentation, Systems, and Automation Society, North Carolina, USA.

Berenz, V., and Suzuki, K. (2011). Risk and gain battery management for self-docking mobile robots. In: *Proceedings of IEEE International Conference on Robotics and Biomimetics*, Karon Beach, Phuket, Thailand, pp. 1766 - 1771.

Bhatti, O.S., Mehmood-ul-Hasan, K., and Imtiaz, M.A. (2015). Attitude Control and Stabilization of a Two-Wheeled Self-Balancing Robot. *Control Engineering and Applied Informatics*, 17(3), pp. 98-104.

Bhatti, O.S., Tariq, O.B., Manzar, A., Khan, O.A. (2018). Adaptive intelligent cascade control of a ball-riding robot for optimal balancing and station-keeping. *Advanced Robotics*, 32(2), pp. 63–76.

Cărbureanu, M. (2014). The Development of a Neuro-Fuzzy Expert System for Wastewater pH Control. *Control Engineering and Applied Informatics*, 16(4), pp. 30-41.

Chen, Z., Xiong, R., Wang, K., Jiao, B. (2015). Optimal energy management strategy of a plug-in hybrid electric vehicle based on a particle swarm optimization algorithm, *Energies*, 8(5), pp. 3661-3678.

Chopra, V., Singla, S.K., and Dewan, L. (2014). Comparative analysis of tuning a PID controller using intelligent methods. *ACTA Polytechnica Hungarica*, 11(8), pp. 235-249.

Dalal, M., Ma, J., and He, D. (2011). Lithium-ion battery life prognostic health management system using particle filtering framework. *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk Reliability*, 225(1), pp. 81-90.

Elbori, A., Turan, M., and Arikan, K. B. (2018). Evaluation and Optimization of Nonlinear Central Pattern Generators for Robotic Locomotion. *Control Engineering and Applied Informatics*, 20(3), pp. 89-98.

Gritli, W., Gharsallaoui, H., and Benrejeb, M. (2016). PID-type fuzzy scaling factors tuning using genetic algorithm and Simulink Design Optimization for Electronic Throttle Valve. In: *Proceedings of International Conference on Control, Decision and Information Technologies*, St. Julian’s, Malta, pp. 216-221.

Gui, P., Tang, L., and Mukhopadhyay, S. (2015). MEMS based IMU for tilting measurement: Comparison of complementary and Kalman filter based data fusion. In: *Proceedings of IEEE 10th Conference on Industrial Electronics and Applications*, Auckland, New Zealand, pp. 2004-2009.

Gurban, E. H., Dragomir, T. L., and Andreescu, G. D. (2014). Greenhouse Climate Control Enhancement by Using Genetic Algorithms. *Control Engineering and Applied Informatics*, 16(3), pp. 35-45.

Halal, F., and Dumitrache, I. (2006). Genetic Algorithm in Mobile Robot Control. *Control Engineering and Applied Informatics*, 8(2), pp. 21-30.

Hussain, K.M., Zepherin, R.A.R., and Shantha, M. (2014). Comparison of PID controller tuning methods with genetic algorithm for FOPTD system. *International Journal of Engineering Research and Applications*, 4(2), pp. 308-314.

Ioanaş, G. L. (2012). Modeling, Identification and Prediction of Inherent quasi-stationary Pressure Dynamics of a Common-Rail System using Neuro-Fuzzy Structures with Local Linear ARX models. *Control Engineering and Applied Informatics*, 14(3), pp. 61-70.

Jaen-Cuellar, A.Y., Romero-Troncoso, R.J., Morales-Velazquez, L., and Osornio-Rios, R.A. (2013). PID-controller tuning optimization with genetic algorithms in servo systems. *International Journal of Advanced Robotic Systems*, 10(9), pp. 1-12.

Jamaluddin, A., Sihombing, L., Supriyanto, A., Purwanto, A., and Nizam, M. (2013). Design real time battery monitoring system using LabVIEW interface for Arduino (LIFA). In: *Proceedings of 2013 Joint International Conference on Rural Information & Communication Technology and Electric-Vehicle Technology*, Bandung, Indonesia, pp. 1-4.

Kim, B.G., Patel, D.D., and Salameh, Z.M. (2013). Circuit Model of 100 Ah Lithium Polymer Battery Cell. *Journal of Power and Energy Engineering*, 1, pp. 1-8.

Kumar, V., Gaur, P., and Mittal, A. P. (2015). Novel AI based On-Line Sequential Learning Technique for High Performance DC Servo motor Control. *Control Engineering and Applied Informatics*, 17(2), pp. 3-11.

Li, Z., Yang, C., Fan, L. (2013). *Advanced Control of Wheeled Inverted Pendulum Systems*. Springer-Verlag, London, UK.

Lin, G., and Liu, G. (2010). Tuning PID controller using adaptive genetic algorithms. In: *Proceedings of 5th International Conference on Computer Science and Education*, Hefei, China, pp. 519-523.

Liu, W., and Dai, J. (2015). Design of Attitude Sensor Acquisition System Based on STM32. In: *Proceedings of Fifth International Conference on Instrumentation and Measurement, Computer, Communication and Control*, Qinhuangdao, China, pp. 1850-1853.

Malysz, P., Sirouspour, S., and Emadi, A. (2014). An optimal energy storage control strategy for grid-connected microgrids. *IEEE Transactions on Smart Grids*, 5(4), pp. 1785-1796.

Mantri, G., Kulkarni, N.R. (2013). Design and Optimization of PID Controller Using genetic algorithm. *International Journal of Research in Engineering and Technology*, 2(6), pp. 926-930.

Mohammed, N.F., Song, E., Ma, X., Hayat, Q. (2014). Tuning of PID controller of synchronous generators using genetic algorithm. In: *Proceedings of IEEE International Conference on Mechatronics and Automation*, Tianjin, China, pp. 1544-1548.

Mosaad, M.I., Salem, F. (2014). LFC based adaptive PID controller using ANN and ANFIS techniques. *Journal of Electrical Systems and Information Technology*, 1(3), pp. 212-222.

Panday, A., and Bansal, H.O. (2014). A review of optimal energy management strategies for hybrid electric vehicle. *International Journal of Vehicular Technology*, 2014, pp. 1-19.

Pareek, S., Kishnani, M., and Gupta, R. (2014). Optimal tuning of PID controller using genetic algorithm and swarm techniques. *International Journal of Electronic and Electrical Engineering*, 7(2), pp. 189-194.

Petcut, F. V., and Dragomir, T. L. (2010). Solar Cell Parameter Identification using Genetic Algorithms. *Control Engineering and Applied Informatics*, 12(1), pp. 30-37.

Pourmousaviv, S.A., Nehrir, M.H., Colson, C.M., and Wang, C. (2010). Real-time energy management of a stand-alone hybrid wind-microturbine energy system using particle swarm optimization. *IEEE Transactions on Sustainable Energy*, 1(3), pp. 193-201.

Rashid, M.H. (2011). *Power Electronics Handbook*. Elsevier, Amsterdam, Netherlands.

Rout, B.K., and Mittal, R.K. (2009). Optimal manipulator parameter selection using evolutionary optimization technique. *Inverse Problems in Science and Engineering*, 17(7), pp. 915-935.

Saha, B., and Goebel, K. (2009). Modeling Li-ion battery capacity depletion in a particle filtering framework. In: *Proceedings of the Annual Conference of the Prognostics and Health Management Society*, San Diego, CA, USA, pp. 1-10.

Saleem, O., Abbas, F., Khan, M. U., Imtiaz, M. A., and Khalid, S. (2018). Adaptive Collaborative Position Control of a Tendon-Driven Robotic Finger. *Control Engineering and Applied Informatics*, 20(2), pp. 87-99.

Serrao, L., Onori, S., Sciarretta, A., Guezennec, Y., and Rizzoni, G. (2011). Optimal energy management of hybrid electric vehicles including battery aging. In: *Proceedings of the 2011 American Control Conference*, San Francisco, CA, USA, pp. 2125-2130.

Silva, M.F.T., Borges, L.M.S.A., Rochinha, F.A., Carvalho, L.A.V.D. (2004). A genetic algorithm applied to composite elastic parameters identification. *Inverse Problems in Science and Engineering*, 12(1), pp. 17-28.

Srinivas, P., Lakshmi, K.V., and Kumar, V.N. (2014). A Comparison of PID Controller tuning methods for three tank level process. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, 3(1), pp. 6810-6820.

Stan, S. D., Maties, V., and Balan, R. (2007). Multi-Objective Design Optimisation of a Planar Micro Parallel Robot using Genetic Algorithms. *Control Engineering and Applied Informatics*, 9(1), pp. 41-46.

Szymak, P. (2016). Using Neuro-Evolutionary-Fuzzy Method to Control a Swarm of Unmanned Underwater Vehicles. *Control Engineering and Applied Informatics*, 18(3), pp. 82-92.

Tam, J.H., Ong, Z.C., Ismail, Z., Ang, B.C., Khoo, S.Y., and Li, W.L. (2018). Inverse identification of elastic properties of composite materials using hybrid GA-ACO-PSO algorithm. *Inverse Problems in Science and Engineering*, 26(10), pp. 1432-1463.

Toufouti, R., Meziane, S., and Benalla, H. (2009). New Direct Torque Neuro-Fuzzy Control Based SVM for Dual Two Level Inverter-Fed Induction Motor. *Control Engineering and Applied Informatics*, 11(2), pp. 3-13.

Vijayakumar, K., and Manigandan, T. (2016). Nonlinear PID Controller Parameter Optimization using Enhanced Genetic Algorithm for Nonlinear Control System. *Control Engineering and Applied Informatics*, 18(2), pp. 3-10.

Wirasingha, S.G., and Emadi, A. (2011). Classification and review of control strategies for plug-in hybrid electric vehicles. *IEEE Transactions on Vehicular Technology*, 60(1), pp. 111-122.

Yusoff, T.A.F.K., Atan, M.F., Rahman, N.A., Salleh, S.F., and Wahab, N.A. (2015). Optimization of PID tuning using genetic algorithm. *Journal of Applied Science and Process Engineering*, 2(2), pp. 97-106.

Zhou, T., and Sun, W. (2014). Optimization of battery–supercapacitor hybrid energy storage station in wind/solar generation system. *IEEE Transactions on Sustainable Energy*, 5(2), 408-415.