FOPI Controller for Blood Pressure Regulation using Hippopotamus Algorithm

Gaurav Kumar Gupta[1][0009−0000−9511−4964], Sumit Ranjan Kumar[1][0009−0006−5869−1176], and Ahmad Ali[1][0000−0002−7727−761X]

Indian Institute of Technology Patna, Bihar, India

Abstract. Blood pressure readings indicate the force the heart exerts on arterial walls while pumping blood. Maintaining mean arterial blood pressure during surgeries is critical to controlling excessive bleeding. Vasodilating drugs like Sodium Nitroprusside (SNP) are used to lower blood pressure, but their required dosage and effects vary significantly depending on the condition of the patient. This necessitates continuous monitoring and precise drug delivery. To achieve this, a fractional-order proportional integral (FOPI) controller is designed using a graphical approach based on the maximum sensitivity criterion, ensuring adequate phase margin (ϕ_m) and gain margin (G_m) for stability. The FOPI parameters are then optimized using the hippopotamus optimization algorithm (HOA)), a metaheuristic technique that mimics hippopotamus behavior in the wild. This algorithm minimizes integral absolute error (IAE) and integral time absolute error (ITAE), which are taken as cost functions. This study evaluates the robustness and performance of the proposed controller compared to existing control strategies. A MATLAB environment is used to simulate the system and implement the algorithm, demonstrating that the proposed controller ensures lesser settling time and steady-state error while being more practical for real-time implementation.

Keywords: Mean Artial Blood Pressure · Hippopotamus Algorithm · Fractional-Order Proportional Integral · H_∞ criteria · Integral Error

1 Introduction

Blood pressure readings measure the force exerted by the heart on arterial walls during blood circulation. Blood pressure is influenced by cardiac output and peripheral resistance. Vasodilators like Sodium Nitroprusside (SNP) are administered to hypertensive patients to lower blood pressure by reducing peripheral resistance. However, precise control of the infusion rate and duration is crucial to prevent toxicity from byproducts such as nitric oxide, thiocyanates, and cyanides, which form when SNP breaks down in the body. Excessive blood pressure reduction can lead to adverse effects, particularly in post-operative patients, increasing the risk of shock. Therefore, accurate infusion rate control is essential. Clinical SNP administration remains challenging due to patient variability and the need for controlled drug release. To address this, automated drug delivery systems are required to dynamically adjust infusion rates and maintain optimal blood pressure levels. [1].

Sheppard et al. developed a computer-based control system by analyzing the physiological responses of SNP in post-operative patients. However, the system did not account for patient sensitivity [2]. These early controllers lacked adaptability to real-world disturbances, such as surgical stimuli, necessitating the development of adaptive controllers. To address these limitations, Slate and Sheppard proposed a model-based adaptive controller that incorporated patient sensitivity to the drug [3]. Internal Model Control (IMC) improved peak overshoot, settling time, and steady-state performance. Further tuning of IMC enhanced robustness and disturbance rejection through IMC filtering [4]. Although optimal IMC provided certain advantages, PID controllers demonstrated shorter settling times [5]. IMC-based PID controllers further refined response characteristics compared to standard IMC controllers. Hamamaci [6] introduced an algorithm for regulating mean arterial pressure using a fractional-order time-delay system. This approach utilized fractional-order PID controllers, which exhibited superior disturbance rejection and greater adaptability to uncertain conditions compared to conventional PID controllers. To enhance controller parameter tuning, optimization strategies have gained substantial attention. Both heuristic and metaheuristic techniques are widely employed for this purpose. Several metaheuristic algorithms, such as the seeker optimization algorithm, whale optimization algorithm, multi-strategy modified INFO algorithm, logarithmic spiral search-based arithmetic algorithm, particle swarm optimization, and human-inspired methods, have been applied. Additionally, genetic algorithms are frequently used to optimize PID parameters [7].

This paper focuses on designing a FOPI controller for regulating MABP. The control parameter search space for K_p and K_i is determined using a graphical approach based on the H_∞ criterion [10]. Within this search space, the hippopotamus optimization algorithm (HOA) is employed to identify suboptimal parameters by minimizing integral errors namely, integral absolute error (IAE) and integral time absolute error(ITAE), taken as cost functions. The algorithm operates through a three-phase model that mathematically simulates group dynamics, including positional updates in aquatic environments, predator defense mechanisms, and evasion strategies [8].

The key contributions of this work are as follows:

- Development of a two-stage FOPI controller.
- Finding search space using H_∞ criterion based graphical approach.
- Tuning the FOPI parameters using HOA by minimizing integral error.
- Analysis of simulation results showing reduced steady-state error and settling time.
- The FOPI controller, having 2 less parameters than a FOPID controller is easier to implement in real-time and also has lesser computational burden.

The rest of the paper is structured as follows: Section II presents the patient response model, which serves as the plant. Section III details the controller design and the H_∞ criterion. Section IV introduces the HOA method, followed by Section V, which discusses simulation results and analysis. Finally, Section VI concludes the study with key findings and future directions.

2 Patient Response Model

Slate et al. [1] investigated the impact of SNP infusion on patients by analyzing five parameters that exhibit significant variability in MABP in response to the drug.

The transfer function model used by Slate is:

$$P(s) = \frac{\Delta P_d(s)}{I_n(s)} = \frac{Ke^{-T_{id}s}(1 + \alpha e^{-T_{cd}s})}{\tau s + 1} \tag{1}$$

where $\Delta P_d(s)$ represents the change in MABP (mmHg), $I_n(s)$ is the drug infusion rate (ml/hr), The patients are classified into three categories based on their sensitivity to the drug, as shown in Table 1.

Table 1. Parameter Values of Patients

Symbol	Parameter Name	Sensitive	Normal	Insensitive
K (mmHg/(ml/hr))	Drug sensitivity	-9	-0.7143	-0.1786
α	Recirculation constant	0	0.4	0.4
T_{id} (sec)	Initial time delay	20	30	60
T_{cd} (sec)	Recirculation time delay	60	30	45
τ (sec)	Time constant	75	30	40

Biological systems are inherently influenced by disturbances [9]. To achieve greater accuracy, these disturbances should be incorporated into the system model. Enbiya [5] developed a Simulink model to represent these disturbances as shown in Fig 1.

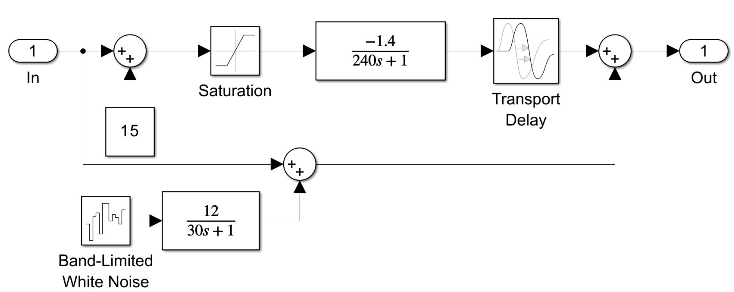

Fig. 1. Model of Disturbances.

3 Controller Design

We are going to design a FOPI controller using a graphical approach based on H_∞ criteria to find the search space for implementing HOA. The control structure using this FOPI controller is shown in Fig. 2.

The objective of this research is to find the search space and then tune the parameters of the FOPI controller to achieve precise tracking of the patient model output to the desired setpoint. We employ HOA to minimize the error, specifically using Integral Absolute Error (IAE) and Integral Time Absolute Error (ITAE) as cost functions.

4 H_∞ Criterion

The H_∞ criteria is a robust control design technique that converts a control problem statement to a mathematical optimization problem. For the suboptimal control, the maximum value of gain of sensitivity function must be less than γ (defined as maximum sensitivity). Therefore our problem is to find the controller minimizing this sensitivity value:

$$\left|\frac{1}{1+P(s)C(s)}\right|_\infty < \gamma \tag{2}$$

. To apply the H_∞ criteria on the human body plant, its equation is taken with real and imaginary parts separated:

$$P(s) = P(j\omega) = P_r(\omega) + jP_i(\omega) \tag{3}$$

And a FOPI controller defined as

$$C(j\omega) = K_p - \frac{K_i}{j\omega^\lambda} \tag{4}$$

Putting the values of $P(s)$ and $C(s)$ in eqn. 2 gives us

$$\left|1 + (P_r(\omega) + jP_i(\omega))\left(K_p - \frac{jK_i}{\omega^\lambda}\right)\right| > \frac{1}{\gamma} \tag{5}$$

Solving and rearranging the above equation gives us the equation of an ellipse:

$$\frac{(K_i - C_1(\omega))^2}{a^2} + \frac{(K_p - C_2(\omega))^2}{b^2} > 1 \quad (13) \tag{6}$$

where

$$C_1(\omega) = -\frac{\omega\omega^\lambda P_i(\omega)}{|P(j\omega)|^2}, \qquad C_2(\omega) = -\frac{P_r(\omega)}{|P(j\omega)|^2}$$
$$a(\omega) = \frac{\omega^\lambda}{\gamma|P(j\omega)|}, \qquad b(\omega) = \frac{1}{\gamma|P(j\omega)|}$$

The ellipses from these equations are plotted in MATLAB. From the region bounded by, yet outside the ellipses,a range of (K_i,K_p) is selected and then HOA is applied on that range with lambda taken between 0 to 2 to select the parameters for FOPI controller which minimizes the integral error cost functions. The plotted ellipses for a sensitive patiend with the bounded region given to HOA is shown in fig. 3.

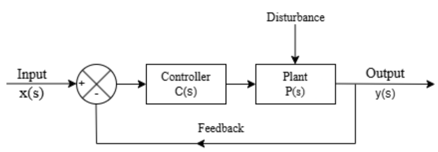

Fig. 2. Proposed Control Structure.

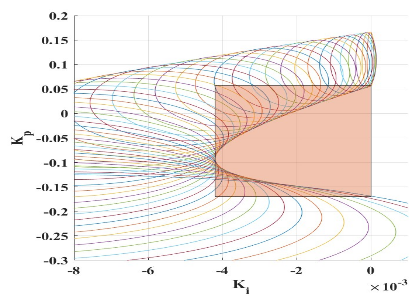

Fig. 3. Ellipses for a sensitive patient and the selected region.

5 Hippopotamus Optimization Algorithm

The Hippopotamus Optimization Algorithm (HOA) is a metaheuristic algorithm inspired by the social behavior of hippopotamuses. It simulates their interactions within a herd, including position updates, defense against predators, and escape mechanisms. HOA balances exploration (searching diverse areas) and exploitation (refining solutions in promising regions) to find optimal solutions. The algorithm works similar to behavioral patterns of hippos in different situations or phases. The pseudocode of HOA with equations is shown in Fig 4

5.1 Steps for Designing the Controller

The following steps outline the process for designing the controller:

1. Define the plant model with suitable patient parameters and the value of $\gamma = 2$ and plot the ellipses for the closed loop function.
2. Select the range of (K_p, K_i) from the region outside yet bounded by the ellipses. And choose a range for λ preferably between 0 and 2.
3. Run HO algorithm in MATLAB on the specified range with number of search agents = 25 and 250 iterations.
4. If the code stops after giving error, check the value of K_p, K_i and λ in the simulation in Simulink. One or more of these parameters have made the simulation unstable so edit the range accordingly and rerun the algorithm.

Algorithm 1 Hippopotamus Optimization Algorithm (HO)

Start HO

Define an optimization problem
Set the maximum number of iterations (T) and number of hippopotamus (N)
Generate the initial position of all hippopotamus and objective function evaluation for this initial population based on the equation : $\chi_i : x_{i,j} = lb_j + r \cdot (ub_j - lb_j)$
for $t = 1$ to T **do**
  Update dominant hippopotamus position based on objective function value criterion
  Phase 1: The hippopotamus's position update in the river or pond **(Exploration Phase)**
  for $i = 1$ to $N/2$ **do**
    Calculate the new position for ith hippopotamus using equations :

$$\chi_i^{M_{hippo}} : x_{i,j}^{M_{hippo}} = x_{i,j} + y_1 \cdot (D_{hippo} - I_1 x_{i,j})$$

$$\text{and} \quad X_i^{FB_{hippo}} : x_{i,j}^{FB_{hippo}} = \begin{cases} x_{i,j} + h_1 \cdot (D_{hippo} - .I_2 \cdot MG_i), & T > 0.6 \\ x_{i,j} + k_2 \cdot (M_{Gi} - D_{hippo}), & T < 0.6 \text{ and } r_6 > 0.5 \\ lb_j + r_7 \cdot (ub_j - lb_j), & \text{otherwise} \end{cases}$$

    Update position of ith hippopotamus using equations:

$$\chi_i = \begin{cases} \chi_i^{M_{hippo}}, & F_i^{M_{hippo}} < F_i \\ \chi_i, & \text{otherwise} \end{cases} \quad \text{and} \quad \chi_i = \begin{cases} \chi_i^{FB_{hippo}}, & F_i^{FB_{hippo}} < F_i \\ \chi_i, & \text{otherwise} \end{cases}$$

  end for
  Phase 2: Hippopotamus defense against predators **(Exploration Phase)**
  for $i = 1 + N/2$ to N **do**
    Generate random position for predator using equation: $Predator_j = lb_j + r_8 \cdot (ub_j - lb_j)$
    Calculate the new position for ith hippopotamus using equation:

$$\chi_i^{\mathrm{HippoR}} = \begin{cases} \mathbf{RL} \oplus \mathrm{Predator}_j + \frac{f}{c - d \cdot \cos(2\pi g)} \cdot \frac{1}{\mathbf{D}}, & F_{\mathrm{Predator}_j} < F_i, \\ \mathbf{RL} \oplus \mathrm{Predator}_j + \frac{f}{c - d \cdot \cos(2\pi g)} \cdot \frac{1}{2.\mathbf{D} + r_9}, & F_{\mathrm{Predator}_j} \geq F_i, \end{cases}$$

    Update the position of ith hippopotamus using equation : $\chi_i = \begin{cases} \chi_i^{HippoR} & F_i^{HippoR} < F_i \\ \chi_i & F_i^{HippoR} \geq F_i \end{cases}$
  end for
  Phase 3: Hippopotamus Escaping from the Predator **(Exploitation Phase)**
  for $i = 1$ to N **do**
    Calculate new bounds of variables decision using equation : $\mathrm{lb}_j^{\mathrm{local}} = \frac{\mathrm{lb}_j}{t}, \quad \mathrm{ub}_j^{\mathrm{local}} = \frac{\mathrm{ub}_j}{t}$
    Calculate the new position for ith hippopotamus using the equation:

$$x_{i,j}^{\mathrm{HippoE}} = x_{i,j} + r_{10} \cdot \mathrm{lb}_j^{\mathrm{local}} + s_1 \cdot \left(\mathrm{ub}_j^{\mathrm{local}} - \mathrm{lb}_j^{\mathrm{local}}\right)$$

    Update the position of ith hippopotamus using equation : $\chi_i = \begin{cases} \chi_i^{HippoE} & F_i^{HippoE} < F_i \\ \chi_i & F_i^{HippoE} \geq F_i \end{cases}$
  end for
  Save the best candidate solution found so far.
end for
Output the best solution of the objective function found by HO

End HO.

Fig. 4. Pseudocode of Hippopotamus Optimization Algorithm.

5. If the final result is achieved at the boundary of the specified region then increase the range and rerun the algorithm.

5.2 Tuned Parameters by HOA

HOA was used to find the suboptimal parameters by simulating the controller in SIMULINK, which minimized the cost function (ITAE and IAE). Its range was chosen from the boundary made by the ellipses taking γ=2 with some additional regions around it too, in cases the algorithm reaches the boundary of the area provided. Table 2 lists the tuned parameters by HOA.

Table 2. Tuned Parameters of FOPI Controller using HOA.

Patient Type	Cost Function	K_p	K_i	λ
Sensitive	IAE	-0.12644	-0.0022915	1.0505
	ITAE	-0.103	-0.0023486	1.0295
Nominal	IAE	-1.6097	-0.009579	1.0919
	ITAE	-1.0323	-0.012667	1.0169
Insensitive	IAE	-3.9225	-0.023564	1.0627
	ITAE	-3.6109	-0.023459	1.0562

6 Simulation Results and Discussion

The set point is taken as -30mmHg to evaluate the effectiveness of the proposed controller. For the sensitive, nominal and insensitive patient models, the step responses of the FOPI controller tuned by HOA in comparison to the FOPID controller tuned by Genetic Algorithm (GA) [11] are shown in Fig 5, Fig 6 and Fig 7 respectively. And their respective control efforts are shown in Fig. 8, Fig. 9 and Fig. 10.

From the figures, it is evident that the controller satisfied the design requirements with much less steady-state error, settling time and integral errors as shown in table 3. The settling time for sensitive patient model was reduced by 35% and for nominal ITAE, the settling time was reduced by 80%. Sensitive patients have the least settling time while insensitive patients take longest to settle. In the case of insensitive patients and nominal IAE, the steady-state error for the FOPID (GA) controller was more than ±5% of the setpoint -30 mmHg, so their settling time was not taken. ITAE cost function was more optimized (50% - 70%) for all the patient types when compared to the optimization IAE cost function(25% - 45%). The control effort of the FOPI (HOA) controller when compared to the FOPID (GA) controller is smaller in the order of magnitudes at t=1 sec.

This approach balances performance and stability in the presence of model uncertainties, making it particularly useful for designing FOPI controllers in

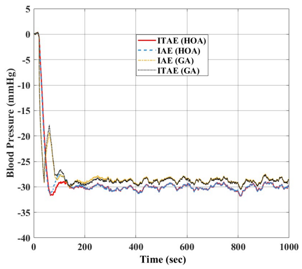

Fig. 5. Sensitive Patient Results.

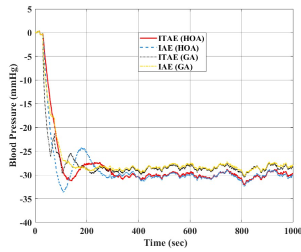

Fig. 6. Nominal Patient Results.

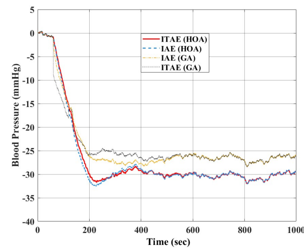

Fig. 7. Insensitive Patient Results.

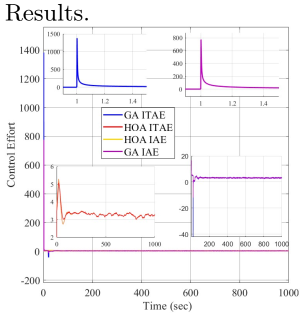

Fig. 8. Control Effort for Sensitive.

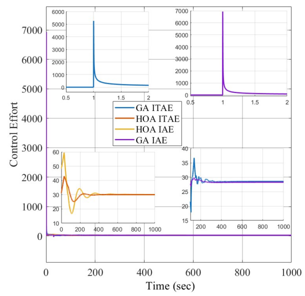

Fig. 9. Control Effort for Nominal.

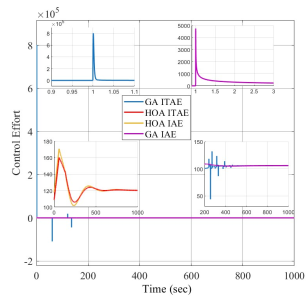

Fig. 10. Control Effort for insensitive.

complex systems like the human body. H_∞ control is crucial for blood pressure regulation due to its inherent robustness, which allows it to handle patient-specific sensitivities, varying physiological responses, and external disturbances. Furthermore, it effectively manages time delays inherent in the response to drugs in the body, optimizes the trade-off between performance (e.g., settling time) and robustness, and maintains stability across a range of conditions, preventing excessive blood pressure fluctuations.

Table 3. Performance Indices for Patient Model

		Integral Error		Steady State Error		Settling Time (s)	
Patient	Cost	FOPID GA	FOPI HOA	FOPID GA	FOPI HOA	FOPID GA	FOPI HOA
Sens.	IAE	2.36e3	1.3e3	-1.27	0.023	81	53
	ITAE	0.59e6	0.23e6	-1.17	0.016	114	78
Nom.	IAE	3.2e3	2.4e3	-1.734	0.103	–	232
	ITAE	0.73e6	0.35e6	-1.458	0.022	583	104
Insens.	IAE	6.2e3	4.0e3	-3.545	0.172	–	429
	ITAE	1.94e6	0.49e6	-3.533	0.170	–	422

7 Conclusion

A FOPI controller is designed to control the mean artrial blood pressure of three different patient models. A H_∞ criteria-based graphical approach was used to find the range for the parameters of the controller and the parameters were tuned by hippopotamus optimization algorithm. The controller used a 2-D graphical approach of a K_p vs K_i graph to get the range of parameters. To get even better results, future research aims to use other techniques to estimate the range of the Kd parameters and fine tune it using HOA to design a FOPID controller.

References

1. Behbehani, K., Cross, R.R.: A controller for regulation of mean arterial blood pressure using optimum nitroprusside infusion rate. IEEE Trans. Biomed. Eng. **38**(6), 513–521 (1991).
2. Sheppard, L.C., Shotts, J.F., Roberson, N.F., Wallace, F.D., Kouchoukos, N.T.: Computer controlled infusion of vasoactive drugs in post cardiac surgical patients. In: IEEE/1979 Frontiers of Engineering in Health Care (IEEE CH1440-7), pp. 280–284 (1979).
3. Slate, J.B., Sheppard, L.C.: A model based adaptive blood pressure controller. In: 6th IFAC Conference on Identification System Parameter Estimation, Washington, DC, pp. 1437–1442 (1982).
4. Rao, P.V.G.K., Subramanyam, M.V., Satyaprasad, K.: Robust Design of PID Controller Using IMC Technique for Integrating Process Based on Maximum Sensitivity. J. Control Autom. Elect. Syst. **26**(5), 466–475 (2015).
5. Enbiya, E., Hossain, E., Mahieddine, F.: Performance of optimal IMC and PID Controllers for blood Pressure control. In: Proceedings of IFMBE, Miami, Florida, pp. 89–94 (2009).
6. Hamamci, S.E.: An algorithm for stabilizing fractional-order time delay systems using fractional-order PID controllers. IEEE Trans. Autom. Control **52**(10), 1964–1969 (2007).
7. Rao, P.V.G.K., Subramanyam, M.V., Satyaprasad, K.: Performance Comparison of PID Controller Tuned using Classical and Genetic Algorithm Methods. Int. J. Appl. Eng. Res. **6**(14), 1757–1766 (2011).
8. Amiri, M.H., Mehrabi Hashjin, N., Montazeri, S., Mirjalili, S., et al.: Hippopotamus optimization algorithm: a novel nature-inspired optimization algorithm. Sci Rep **14**, 5032 (2024).
9. Sondhi, S., Hote, Y.V.: Fractional-Order PI Controller with Specific Gain-Phase Margin for MABP Control. IETE J. Res. (2015).
10. Kumar S.R., Anwar M.N., Verma P., Somanshu S.R.: H_∞ Criterion Based PI Controller Design for DC-DC Step-Up Converters. IEEE Journal of Emerging and Selected Topics in Industrial Electronics, vol. 5, no. 2, (2024).
11. Siva Krishna P., Gopi Krishna Rao P.V.: Fractional-order PID controller for blood pressure regulation using genetic algorithm. Biomedical Signal Processing and Control, vol. 88, Part B, February 2024, 105564.