OPTIMIZATION OF HORIZONTAL CYLINDRICAL TANK LIQUID LEVEL PROCESS CONTROL USING PSO PI CONTROLLER

Parthasarathy, R[1*], Balaji, P[2]
[1*] Student, Dept. of EIE, St. Peter's Institute of Higher Education and Research, Chennai.
[2]Professor and Head, Dept. of E&I, St. Peter's Institute of Higher Education and Research, Chennai.

Abstract : An intelligent optimization method for designing PID controllers based on particle swarm optimization (PSO) is presented in this paper. The conventional gain tuning of PID controller (such as Ziegler-Nichols (ZN) method) usually produces a big overshoot, and therefore modern heuristics approach such as genetic algorithm (GA) and particle swarm optimization (PSO) are employed to enhance the capability of traditional techniques. However, due to the computational efficiency, only PSO will be used in this paper. The performance comparison of the ZNPI and PSO based PI controllers are compared based on performance indices like maximum peak overshoot, settling time, Integral Square Error (ISE) and integral absolute error (IAE). The proposed PSO based PI controller is tested on the chosen Horizontal Cylindrical Tank level system and better controller performance can be envisaged by in the proposed methods than that of the ZNPI controller.

Keywords: Horizontal Cylindrical Tank, Optimization, Particle Swarm Optimization, PI Controller,

I. INTRODUCTION

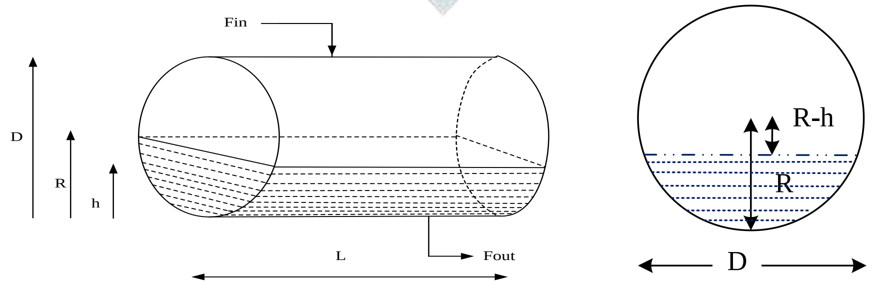
Particle swarm optimization (PSO) is a population based stochastic optimization technique developed by Dr. Eberhart and Dr. Kennedy in 1995, inspired by social behaviour of bird flocking or fish schooling's shares many similarities with evolutionary computation techniques such as Genetic Algorithms (GA). The system is initialized with a population of random solutions and searches for optima by updating generations. However, unlike GA, PSO has no evolution operators such as crossover and mutation. In PSO, the potential solutions, called particles, fly through the problem space by following the current optimum particles. PSO algorithm in finding optimal values follows the work of this animal society. Particle swarm optimization consists of a swarm of particles, where particle represent a potential solution. [1-4]

In past several years, PSO has been successfully used across a wide range of application fields as well as in specific applications focused on a specific requirement for the two reason following. First it is demonstrated that PSO gets better results in a faster, cheaper way compared with other methods and second reason that PSO is attractive is that there are few parameters to adjust. One version, with slight variations, works well in a wide variety of applications [5-7].

The present work deals with the design of controller for hemispherical tank system. The contribution of this work consists mainly in the design of K_P, K_i, and K_d, values are found using three types of Particle swarm optimization techniques to design the PID controller and compared with conventional one. The development and implementation of the proposed system and controllers was done using MATLAB/Simulink.

II. HORIZONTAL CYLINDRICAL TANK SYSTEM

The horizontal tank such as oil, the chemical liquid in its surge drum level control system has shown Fig. 1. The purpose of the surge vessel is to smooth variations in the flow from process one and maintain a relatively constant flow rate to process two. The level can vary substantially from the set point, as long as the vessel does not overflow or go dry. The main object is to vary the manipulated flow rate (the outlet flow from the vessel) as little as possible while satisfying level constraints. Surge vessels are used to help reduce the effect of flow rate variations between interconnected process units. It is necessary to maintain tight level control in a surge vessel.



(a) (b)

Fig. 1 Structure of horizontal cylindrical Tank

The mathematical model of the horizontal cylindrical tank liquid level system considered for the study is expressed as,

Let R, be the radius of the cross-section.
h, be a level of liquid inside the tank.
D, be the diameter of the cross-section.
L, be the height of the tank.

III. PID CONTROLLER

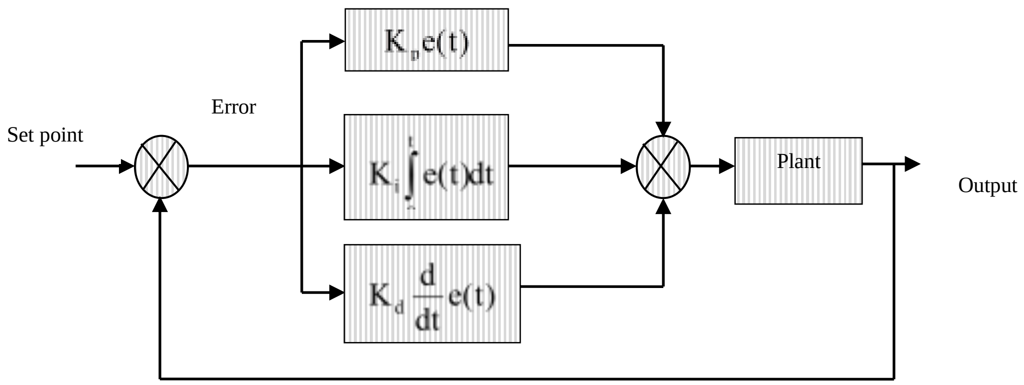


Fig. 2 Basic structure of PID controller

PID - most widely used a type of controller for industrial applications. And exhibit robust performance over a wide range of operating conditions. The three main parameters involved are Proportional (P), Integral (I) and Derivative (D). The proportional part acts on the present value of the error, the integral represent an average of past errors and the derivative can be interpreted as a prediction of future errors based on linear extrapolation, shown in Fig. 3.1.

The PID controller is

$$u(t) = k_e(t) + k_i \int_0^t e(t)dt + k_d \frac{de(t)}{dt} \qquad (1)$$

Where u is the control signal and e is the control error (e = r - y). The reference value is also called the set point. The control signal is thus a sum of three terms: the P-term (which is proportional to the error), the I-term (which is proportional to the integral of the error), and the D-term (which is proportional to the derivative of the error). The controller parameters are proportional gain k, integral gain k_i and derivative gain k_d.

3.1 Tuning of PI Controller

The goal of PI controller tuning is to determine parameters that meet closed loop system performance specifications, and to ensure the robust performance of the control loop over a wide range of operating conditions. Practically, it is often difficult to simultaneously achieve all of these desirable qualities. For example, if the PI controller is adjusted to provide better transient response to set point change, it usually results in a sluggish response when under disturbance conditions. On the other hand, if the control system is made robust to disturbance by choosing conservative values for the PI controller, it may result in a slow closed loop response to a set point change. A number of tuning techniques that take into consideration the nature of the dynamics present within a process control loop have been proposed by Ziegler and Nichols, (1942); Cohen and Coon, (1953); Astrom and Hagglund, (1984); and Atherton, (1993). All these methods are based upon the dynamical behaviour of the system under either open-loop or closed-loop conditions.

IV. PARTICLE SWARM OPTIMIZATION

PSO is an evolutionary computational technique based on the movement and intelligence of swarms looking for the most fertile feeding location. A “swarm” is an apparently disorganized collection (population) of moving individuals that tend to cluster together, while each individual seems to be moving in a random direction. PSO uses a number of agents (particles) that constitute a swarm moving around in the search space looking for the best solution [2,8,9].

Each particle is treated as a point in an n-dimensional space and adjusts its “flying” according to its own flying experience, as well as the flying experience of other particles. Each particle keeps track of its coordinates in the problem space, which are associated with the best solution (fitness) that has been achieved so far. This value is called p_{best}. Another best value called g_{best} is that obtained so far by any particle in the neighbours of the particle. The PSO concept consists of changing the velocity (or acceleration) of each particle toward its p_{best} and the g_{best} position at each time step. Each particle tries to modify its current position and velocity according to the distance between its current position and p_{best}, and the distance between its current position and the *gbest*. At each step n, by using the individual best position, p_{best}, and global best position, g_{best}, a new velocity for the ith particle is updated by,

$$v_i^{k+1} = wv_i^k + c_1 r_1 \times (pbest_i - x_i^k) + c_2 r_2 \times (gbest - x_i^k) \qquad (2)$$

$$x_{t+1} = x_t + x_{t+1} \qquad (3)$$

With regards to (2):

w	=	Inertial Weight
v_i^k	=	current velocity of agent i at iteration k
v_i^{k+1}	=	new velocity of agent i at iteration k+1
c_1, c_2	=	adjustable social acceleration constant (swarm confidence),
r_1, r_2	=	random number between 0 and 1,
x_i^k	=	current position of agent i at iteration k,
$pbest_i$	=	personal best of agent i ,
gbest	=	global best of the population.

For (3):

x_i^{k+1} = position of agent iat the next iteration k+1,

The parameter 'W' in Equation (2) is inertia weight that increases the overall performance of PSO. It is reported that a larger value of 'W' can favour higher ability for global search while lower value of W implies a higher ability for local re-search. To achieve a higher performance, we linearly decrease the value of inertia weight W over the generations to favour global re-search in initial generations and local re-search in the later generations. The linearly decreasing value of inertia is expressed in Equation (4).

$$w = w_{max} - iter * \frac{w_{max} - w_{min}}{iter_{max}} \quad (4)$$

Where $iter_{max}$ is the maximum of iteration in evolution process, w_{max} is maximum value of inertia weight, w_{min} is the minimum value of inertia weight, and iter is current value of iteration.

Once the particle computes the new x_t it then evaluates its new location. If fitness (x_t) is better than fitness (p_{best}) , then p_{best} = x_t and fitness (p_{best}) = fitness (x_t), in the end of iteration the fitness (g_{best}) = the better fitness (p_{best}), and $g_{best} = p_{best}$.

The PSO algorithm method has been implemented as M file by MATLAB which is interconnected to the Simulink model , where the PID controller parameters are computed and fed to the GUI of the controller. The optimization performed with this initial parameter, number of particles 30, number of dimensions 3, maximum iteration 50, C_1=1, C_2=3, with the objective function ITAE or ISE. The initial values of three parameters K_p, K_i and K_d, of the PID controller will be generated in PSO program and submitted and running the simulation automatically then compute the objective function ITAE and go back with value of ITAE to PSO program to improve the value of K_p, K_i and K_d, , and go on. In the end of iteration the parameters of the PID controller K_p, K_i, K_d has been obtained directly according to the minimum value of objective function ITAE. Fig. 3, shows the flowchart of PSO based PID tuning algorithm

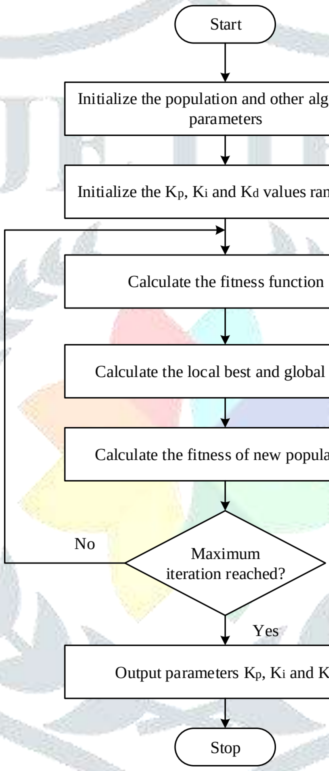


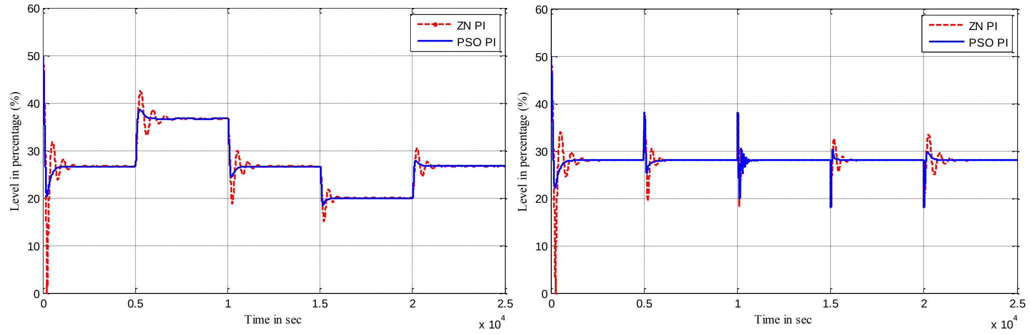
Fig. 3 flow chart of PSO based PID tuning

The design steps of PSO based PID controller

1. Initialize the algorithm parameters like a number of generations, population, inertia weight, cognitive and social coefficients.
2. Initialize the values of the parameters K_p, K_i and K_d randomly.
3. Calculate the fitness function of each particle in each generation.
4. Calculate the local best of each particle and the global best of the particles.
5. Update the position, velocity, local best and global best in each generation.
6. Repeat the steps 3 to 5 until the maximum iteration reached or the best solution is found.

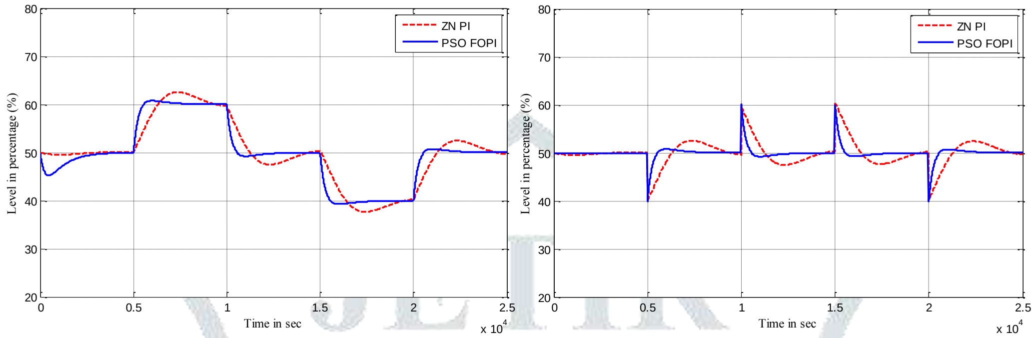
V. PERFORMANCE OF PSO BASED PI CONTROLLER

The performance of Particle Swarm Optimization based PI controller for horizontal cylindrical tank level process is compared with conventional ZN PI.



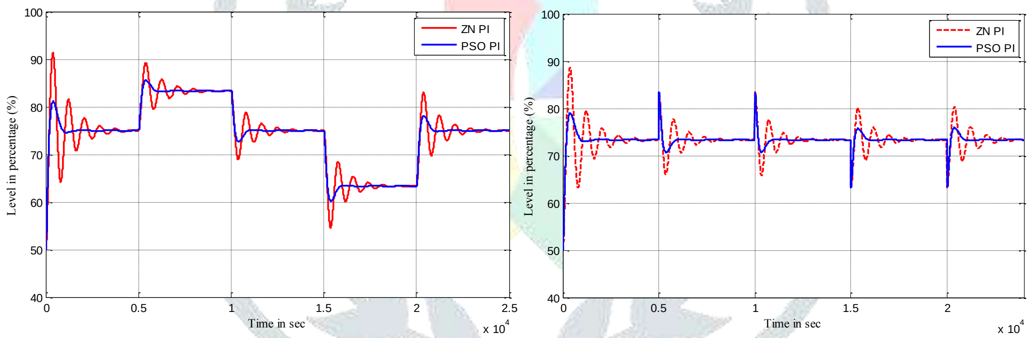
(a) Servo Response (b) Regulatory Response

Fig. 4 Horizontal cylindrical tank level for 10% increment and decrement in load from nominal operating load of 25% using PSO tuned PI controller



(a) Servo Response (b) Regulatory Response

Fig. 5 Horizontal cylindrical tank level for 10% increment and decrement in load from nominal operating load of 50% using PSO tuned PI controller



(a) Servo Response (b) Regulatory Response

Fig. 6 Horizontal cylindrical tank level for 10% increment and decrement in load from nominal operating load of 75% using PSO tuned PI controller

Fig. 4(a) shows the servo response of 10% increment in set point from nominal operating point and 10% decrement in setpoint from the nominal operating point of 25%. Fig. 4(b) shows the regulatory response of both positive and negative load change of 10% at a nominal operating load of 25%. Fig. 5(a) shows the servo response of 10% increment in set point from nominal operating point and 10% decrement in setpoint from the nominal operating point of 50%. Fig. 5(b) shows the regulatory response of both positive and negative load change of 10% at a nominal operating load of 50%. Fig. 6(a) shows the servo response of 10% increment in set point from nominal operating point and 10% decrement in setpoint from the nominal operating point of 75%. Fig. 6(b) shows the regulatory response of both positive and negative load change of 10% at a nominal operating load of 75%. PSO based PI controllers give responses with no oscillations, smaller ISE and IAE. In the case of all PSO PI, it is settling time better than ZN PI controller as given in Tables. 1. The graphical analysis of the performance indices is shown in Figs. 7 and 8.

Table 1 Performance index for Horizontal cylindrical Tank at various nominal operating points

Nominal operating point	Controller	Servo Response				Regulatory Response			
		10% increment in SP		10% decrement in SP		10% increment in load		10% decrement in load	
		over shoot (%)	Settling time (s)	under shoot (%)	Settling time (s)	over shoot (%)	Settling time (s)	under shoot (%)	Settling time (s)
25%	**ZNPI**	5.94	4460	6.43	4410	8.43	4940	8.13	5340

	PSO PI	4.36	1680	4.73	1760	4.43	4460	4.6	4580
50%	**ZNPI**	25.41	5180	12.3	5240	8.33	5090	8.23	6300
	PSO PI	7.02	4140	8.1	3660	2.6	3220	2.65	2130
75%	**ZNPI**	4.7	3980	5.3	3930	5.46	4040	5.4	5280
	PSO PI	2.08	1353	2.3	1208	2.3	1116	2.26	1229

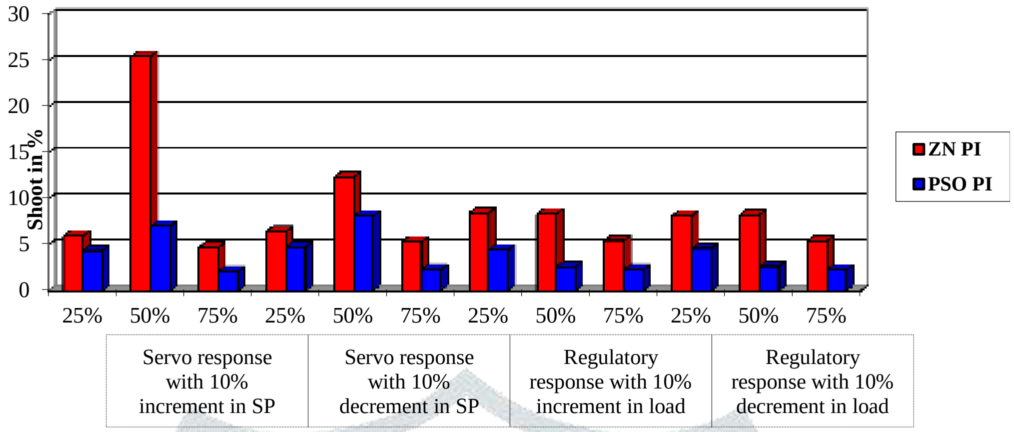


Fig. 7 Graphical analysis of overshoot and undershoot responses of horizontal cylindrical Tank System

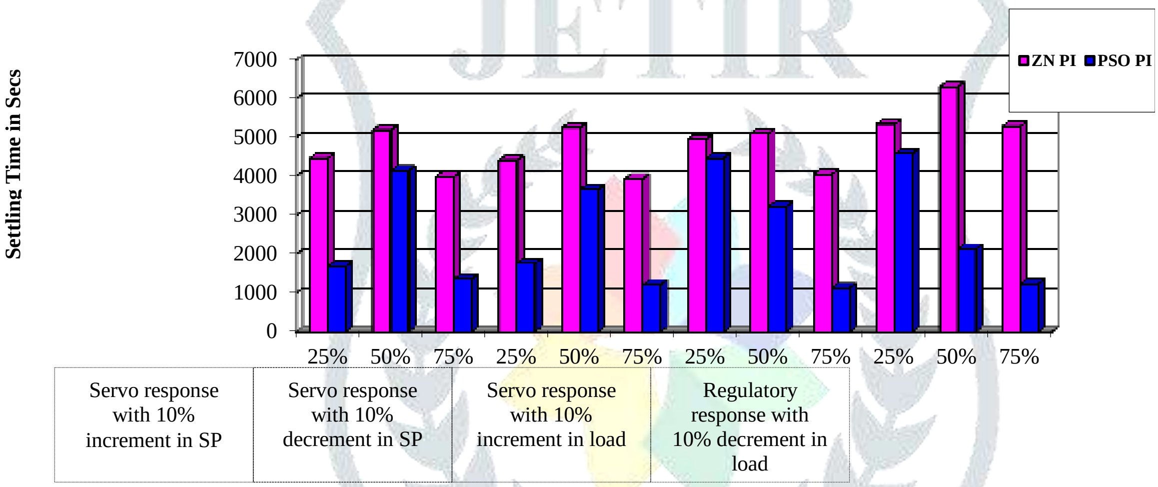


Fig. 8 Graphical analysis of settling time of controllers output response (PV= level in %) of horizontal cylindrical Tank System

VI. CONCLUSION

PSO based PI controller are used to control the level in the horizontal cylindrical tank. It has been shown that the speed of responses of the level control system with and without load interrupt in the tank are fast. In order to appraise the performance of the controller, the proposed controller was done with MATLAB/Simulink. The PSO tuned PI controller offers enhanced process characteristics such as better time domain specifications, smooth reference tracking, supply disturbance rejection, and error minimization compared with ZN PI. In addition, the PSO - PI controller enhanced the flexibility and stability of the horizontal cylindrical tank level process.

REFERENCES

[1] Kennedy, J. and Eberhart, R. C. Particle swarm optimization. Proc. IEEE int'l conf. on neural networks Vol. IV, pp. 1942-1948. *IEEE service center*, Piscataway, NJ, 1995.

[2] Eberhart, R. C. and Kennedy, J. A new optimizer using particle swarm theory. Proceedings of the sixth international symposium on micro machine and human science pp. 39-43. *IEEE service center, Piscataway*, NJ, Nagoya, Japan, 1995.

[3] Eberhart, R. C. and Shi, Y. Particle swarm optimization: developments, applications and resources. *Proc. congress on evolutionary computation 2001 IEEE service center*, Piscataway, NJ., Seoul, Korea., 2001.

[4] M. B. Ghalia, "Particle Swarm Optimization with an Improved Exploration-Exploitation Balance," *iEEE*, vol. 978-14244-2167-1/08/$25.00 ©2008 IEEE, 2008

[5] R. C. Eberhart, Y. Shi, "Comparing inertia weights and constriction factors in particle swarm optimization," *Proceedings of the 2000 Congress on Evolutionary Computation. IEEE Service Center:* Piscataway, NJ, pp. 8488, 2000.

[6] P.Pomeroy, "An Introduction to Particle Swarm Optimization," *http://www.adaptiveview.com/articles/ipsop1.html*, 2003.

[7] J. F. Schutte, J. A. Reinbolt, B. J. Fregly et al., "Parallel global optimization with the particle swarm algorithm," *Int. J. Numer. Meth. Engng.*, vol. 61, pp. 2296-2315, 2004.

[8] F.L. Lewis, D.M. Dawson, C.T. Abdallah, Control of Robot Manipulators, *New York, Macmillan*, 1993.

[9] J. Kennedy, R.C. Eberhart, Y. Shi, Swarm Intelligence, *New York, Morgan Kaufmann*, 2001.

[10] Ho, S. L., S. Yang, G. Ni, et al., "A particle swarm optimization method with enhanced global search ability for design optimizations of electromagnetic devices," *IEEE Trans. Magnetics*, Vol. 42, 1107–1110, 2006.

[11] Schutte, J. F. and A. A. Groenword, "A study of global optimization using particle swarms," *Journal of Global Optimiz.*,Vol. 31, 93–108, 2005.
[12] Zhi-hui Zhan and Jun Zhang Adaptive Particle Swarm Optimization _*Springer-Verlag Berlin Heidelberg 2008 M*. Dorigo et al. (Eds.): ANTS 2008, LNCS 5217, pp. 227–234, 2008.
[13] Zeigler, J. and Nichols, N. "Optimum Settings for Automatic Controllers", *Transactions of ASME*, Vol. 64, pp. 759–768, 1942.
[14] Cohen, G.H. & Coon, G.A., 1953. Theoretical Consideration of Retargeted Control. *Transaction ASME*, 75, pp.827-34.
[15] Astrom, K. J., & Hagglund, T. (1984). Automatic tuning of simple regulators with specifications on phase and amplitude margins. *Automatica*, 20(5), 645–651.