

Tuning Rules for Fractional Order PID Controller Using Data Analytics

P. R. Varshini¹, S. Baskar, M. Varatharajan and S. Sadhana

Department of EEE, Thiagarajar College of Engineering, Madurai, Tamilnadu, 625015, India

*Corresponding Author: P. R. Varshini. Email: varshinipapers541@gmail.com

Received: 08 October 2021; Accepted: 16 November 2021

Abstract: Flexibility and robust performance have made the FOPID (Fractional Order PID) controllers a better choice than PID (Proportional, Integral, Derivative) controllers. But the number of tuning parameters decreases the usage of FOPID controllers. Using synthetic data in available FOPID tuners leads to abnormal controller performances limiting their applicability. Hence, a new tuning methodology involving real-time data and overcomes the drawbacks of mathematical modeling is the need of the hour. This paper proposes a novel FOPID controller tuning methodology using machine learning algorithms. Feed Forward Back Propagation Neural Network (FFBPNN), Multi Least Squares Support Vector Regression (MLSSVR) chosen to design Machine Learning based Optimal Tuner (MLOT) can handle the interdependency between the controller parameters and multiple outputs for multiple inputs. The proposed tuner finds application in the control of power and energy systems. It can accomplish tracking, disturbance-rejection, and robustness controller performances, thus making FOPID controller design easier and accurate. Comparisons with existing FOPID tuning rules show better controller performances and easy tuning. Thus, this paper addresses a unique, real-time, model-free, easily tunable FOPID tuning methodology satisfying plant requirements.

Keywords: Machine learning; data analytics; support vector regression; controller tuning rule; multi least squares support vector regression; fractional order PID controller

1 Introduction

In most of the process control industries, control loops are of Proportional Integral Derivative (PID) type [1]. A Greater number of failures encountered in industrial controllers are due to poor tuning of PID loops. From literature, it is found that PIDs still provide underperformance in most of the process control loops [2,3]. Fractional Order PID (FOPID) controllers introduced by Podlubny [4] in 1994 are more flexible than PID controllers owing to the two additional parameters, the order of integrator ' λ ' and differentiator ' μ ' [5,6]. These parameters provide robust stability and a constant phase around the gain crossover frequency. These advantageous features are seldom achieved in PID controllers.



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Due to design flexibility, the FOPID controllers find applications especially in systems having nonlinear dynamics [7–10], systems with long-dead time [11], higher-order systems [12], and also unstable systems.

Even though the FOPID controller is found to outperform the conventional PID controllers, design of FOPID controller can be more difficult as it involves five tuning parameters namely, proportional gain constant ' K_p ', integral time constant ' T_i ', derivative time constant ' T_d ', order of integrator ' λ ' and order of differentiator ' μ '. On the other hand, due to five tuning parameters, FOPID controller can satisfy five different performance specifications with higher cost/benefit ratio. To increase the applicability of FOPID controller, these parameters should be made easily accessible like PID controllers.

Determination of FOPID controller parameters is achieved using many optimization algorithms [13] and artificial intelligence techniques such as fuzzy, neural networks. These techniques require algorithm execution time and initial ground work like determining membership functions, training of algorithms. A single tuning rule without the need of any mathematical calculations and algorithm execution exists; the computation of controller parameters will be much easier.

In last two decades, tuning rules such as Ziegler-Nichols (ZN), Cohen-Coon and Kappa–Tau are the classical empirical tuning rules for FOPID control parameters. FOPID tuning rules based on optimal load disturbance rejection [14], followed by optimal set point tracking are given in [15,16]. The FOPID tuning rules for integral and unstable processes are also available [17] based on statistical polynomial curve fitting. The statistical polynomial curve fitting requires a representative model having accurate initial estimate of the parameter set. These methods involve very small dataset followed by interpolation of data for curve fitting. Also, these methods offer separate equation for each controller parameter which make the controller design tedious.

In addition to the above said drawbacks, these FOPID tuning rules are devised using very little information on system dynamics, also require prior assumptions, approximations and fail to include robustness. Hence, a more accurate, flexible, easily accessible tuning rule, without the need of any mathematical formulations or initial estimation of parameter set is need of the hour.

Hence to overcome the above-said drawbacks, Machine Learning based Optimal Tuner (MLOT) is proposed in this paper. Machine Learning Algorithm (MLA) is advantageous since, it is the strongest predictive modeling for linear as well as nonlinear patterns, with lesser assumptions supplying a single predictive model [18–20].

In this proposed work, the optimal FOPID controller dataset is generated to achieve two-different performance specifications, Set-Point Tracking (SPT) and Load Disturbance Rejection (LDR) for various First Order Plus Dead Time (FOPDT) systems using Covariance Matrix Adaptive Evolutionary Strategy (CMA-ES) [21].

Data analytics is performed on SPT and LDR datasets to remove outliers if any and to identify the most suitable MLA. Thus, identified MLA accomplishes the task of MLOT. Two MLAs, Feed Forward Back Propagation Neural Network (FFBPNN) [22,23] and Multi-output Least-squares Support Vector Regression Machines (MLSSVR) [24,25] have been identified and MLOT-FFBPNN, MLOT-MLSSVR is devised using the two algorithms. To test the proficiency of the chosen MLA, MLOT is also constructed using Multi-Variate Regression (MVR) [26,27]. The proposed system can be applied to systems with long dead time, higher order systems which is the advantage.

The proposed MLOT is justified in terms of performance specifications, statistical variations in controller parameters obtained from MLAs by comparing with the Tuning Rules (TR) given by Padula and Visioli.

The authors claim, the following points as the novelty of this proposed work.

- a) A Universal tuner for FOPID controllers using MLA is proposed.
- b) Data analysis with R studio[®] is used to identify appropriate MLAs.
- c) Among the two MLAs, MLSSVR is chosen and verified using statistical analysis.
- d) Better tracking, disturbance rejection, and robustness performances were achieved.
- e) Proposed MLOT is applicable to the FOPDT system and also to any higher-order systems.

The organization of the paper is as follows. Section 2 describes the proposed methodology. Results and discussions are placed in Section 3 and conclusions are given in Section 4.

2 Materials and Methods

The flow diagram of the proposed methodology is given in Fig. 1. Initially, CMA-ES algorithm generates dataset with SPT and LDR objectives constrained with Maximum Sensitivity (M_s). Optimal SPT dataset and LDR dataset are obtained as $K_p^{sp}, T_i^{sp}, T_d^{sp}, \lambda^{sp}, \mu^{sp}$ and $K_p^{ld}, T_i^{ld}, T_d^{ld}, \lambda^{ld}, \mu^{ld}$ respectively.

The generated dataset is analyzed using R studio for removing outliers and identification of suitable MLAs. MLOT-FFBPNN and MLOT-MLSSVR are formulated with the dataset from the data analysis block which is applicable to any process control system.

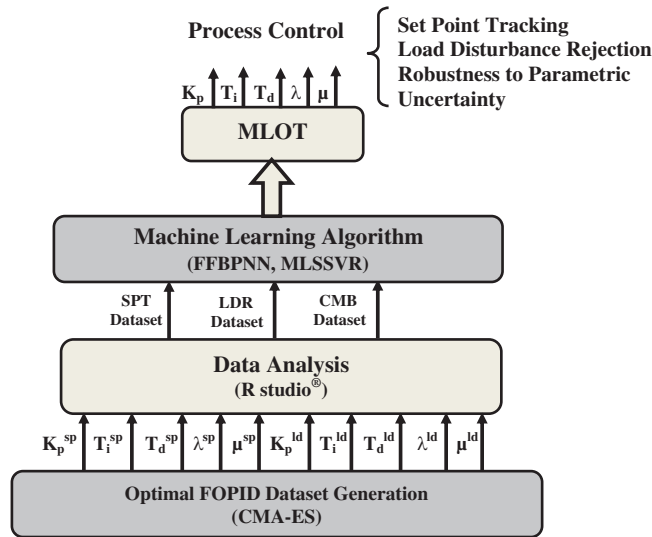


Figure 1: Flow diagram for design of MLOT

2.1 Optimal FOPID Dataset Generation

Optimal FOPID dataset is generated by optimizing the FOPID controller design for a range of FOPDT systems $FOPDT_1, FOPDT_2, \dots, FOPDT_n$. The CMA-ES algorithm is used for optimization with seven different maximum sensitivity values, $M_s = [1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2.0]$ under two design objectives, SPT and LDR. The FOPDT system is in the form of G_p as given in Eq. (1) with process gain ‘ K ’, time constant ‘ T ’, dead time ‘ L ’ and also, normalized dead time, $\tau = L / (L + T)$. The τ is associated with the dynamic behavior of a FOPDT system.

$$G_p(s) = \frac{K}{T_s + 1} e^{-Ls} \tag{1}$$

Hence, the ‘n’ number of different FOPDT systems (FOPDT₁, FOPDT₂, ... FOPDT_n) are produced by varying τ as $\tau_1, \tau_2, \dots, \tau_n$ between the range [0.3, 0.8]. The structure of FOPID controller used is given in Eq. (2) and Oustaloup approximation in Eq. (3)

$$G_c(s) = K_p \left(\frac{T_i s^\lambda + 1}{T_i s^\lambda} \right) \left(\frac{T_d s^\mu + 1}{\frac{T_d}{N} s + 1} \right), 0 \leq \lambda, \mu \leq 2 \quad (2)$$

$$s^\nu \cong k \prod_{n=1}^N \frac{1 + \frac{s}{\omega_{z,n}}}{1 + \frac{s}{\omega_{p,n}}}, \nu > 0 \quad (3)$$

The FOPID controller parameters are obtained for ‘n’ different FOPDT systems by minimizing the Integral Absolute Error (*IAE*). CMA-ES [25,26] optimization algorithm is used to minimize *IAE* in Eq. (4) for the two performance objectives namely, SPT to obtain minimum *IAE_{sp}* and LDR to obtain minimum *IAE_{ld}* as in Eq. (5).

$$\underbrace{\min}_0 IAE_{sp/ld} = \int_0^\infty |e_{sp/ld}(t)| dt \quad (4)$$

subject to, $M_s = M$

where, $M = [1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2.0]$

$$M_s = \underbrace{\min}_\omega \left| \frac{1}{1 + L(j\omega)} \right|, \quad (5)$$

The dataset generated from the optimization block will be of the form as given in Tab. 1. Both SPT and LDR objectives are optimized for 160 numbers of different FOPDT systems and hence 160 numbers of SPT and LDR data points are obtained respectively and this is provided in the last column of Tab. 1. The variables $[\tau, M_s]$ are chosen as input parameters while $[K_p, T_i, T_d, \lambda, \mu]$ are chosen as the output parameters.

Table 1: Optimal FOPID dataset from optimization block

Dataset	Input parameters	Output parameters	Number of data points
SPT	$[\tau^{sp} \quad M_s^{sp}]$	$[K_p^{sp}, T_i^{sp}, T_d^{sp}, \lambda^{sp}, \mu^{sp}]$	160
LDR	$[\tau^{ld} \quad M_s^{ld}]$	$[K_p^{ld}, T_i^{ld}, T_d^{ld}, \lambda^{ld}, \mu^{ld}]$	160

2.2 Data Analysis of Generated Optimal FOPID Dataset

To design MLOT, an MLA that can handle many inputs with many output parameters is required. Before using such MLA, the inter-variable correlation between the output parameters must be determined to obtain an efficient machine learning model. Also, the dataset generated from the optimization block may contain few data points that may differ from other observations termed outliers. Outliers may be due to measurement error which must be discarded to avoid misleading of modeling. Data analysis using R studio[®] is carried out in this work to check and discard any outliers, and also to analyze the relationship between the output parameters of the generated dataset.

2.2.1 Remove Outliers

For a multiple input multiple output model, some data may be suitable in a single dimension but they may become an outlier in a multi-dimension. The Cook’s distance identifies the presence of outliers. These outliers are then removed from the dataset by identifying the data points having standard deviations greater than the mean value.

2.2.2 Cross-Correlation among Output Parameters

The SPT and LDR dataset contains four output variables. The cross-correlation matrix of these four output variables for the SPT dataset after removing outliers is obtained. Positive/Negative large cross-correlation values denote high linear interrelationship among the variables. A Smaller cross-correlation value does not mean that the corresponding parameter is an independent one. For such cases, the cross-correlation plots have to be examined for non-linear relations. Hence, in this work, both the cross-correlation matrix values and the cross-correlation plots have been analyzed for determining the interrelationship among the output parameters before modeling with MLAs.

The SPT and LDR dataset are combined together to obtain the third dataset, the CMB dataset from the generated data points as given in Tab. 2. In the CMB dataset, $\tau^c = [\tau^{sp}, \tau^{ld}]$ while $M_s^c = [M_s^{sp}, M_s^{ld}]$. Also, $K_p^c = [K_p^{sp}, K_p^{ld}]$, $T_i^c = [T_i^{sp}, T_i^{ld}]$, $T_d^c = [T_d^{sp}, T_d^{ld}]$, $\lambda^c = [\lambda^{sp}, \lambda^{ld}]$, $\mu^c = [\mu^{sp}, \mu^{ld}]$.

Table 2: Optimal FOPID dataset along with combined dataset

Dataset	Input parameters	Output parameters	Number of data points (Outliers discarded)
SPT	$[\tau^{sp}, M_s^{sp}]$	$[K_p^{sp}, T_i^{sp}, T_d^{sp}, \lambda^{sp}, \mu^{sp}]$	150
LDR	$[\tau^{ld}, M_s^{ld}]$	$[K_p^{ld}, T_i^{ld}, T_d^{ld}, \lambda^{ld}, \mu^{ld}]$	144
CMB	$[\tau^c, M_s^c]$	$[K_p^c, T_i^c, T_d^c, \lambda^c, \mu^c]$	294

2.3 Development of MLOT

Data analysis results confirm a high cross-correlation among output parameters. Two MLAs, FFBPNN and MLSSVR have been identified to support the highly correlated dataset. FFBPNN and MLSSVR are employed to design the proposed MLOT using the three datasets obtained from the data analysis block.

In FFBPNN, the Levenberg-Marquardt function with Mean Square Error (MSE) minimization is considered. In MLSSVR multi-tasking is achieved by using weight vector, $w_i = w_0 + v_i$, where, $w_i \in \mathfrak{R}$. w_0 is the regular weight vector that determines the output while v_i is dedicated for correlation in output variable. Radial basis function is used as the kernel function and grid search technique in conjunction with Leave one out cross-validation is used to tune the parameters in MLSSVR. Additionally, a Multi-Variate Regression (MVR) model which is an extension of the linear regression model is developed to compare the performances of the developed MLOTs.

The three MLAs are compared based on their statistical parameters such as Correlation Coefficient (CC), Mean Square Error (MSE), Root Mean Square Error (RMSE), and Mean Absolute Error (MAE). The results of MLOTs are also compared with the existing FOPID tuning rule. The MATLAB[®] program to evaluate FOPID controller parameters using the proposed MLOT is available with the authors. This MATLAB[®] program can be used to determine FOPID controller parameters for any FOPID and higher-order systems.

3 Results and Discussions

An Intel® Core™ i7-3632 QM CPU with 2.2 GHz speed and 8 GB RAM computer with 8 logical processors is used to develop the proposed MLOT in this paper. The optimization is carried out with the help of the MATLAB® toolbox while the data analysis is performed using R studio®.

3.1 Optimization/Dataset Generation

The FOPDT system parameters K and T are set to 1. Different FOPDT systems are considered for dataset generation with different $\tau \in [0.3, 0.8]$.

The optimum values of IAE_{sp} obtained from CMA-ES for various FOPDT sample systems are compared with the results obtained from ZN rules and TR given by Padula and Visioli in Tab. 3. The comparison is given for three example systems. It is observed from Tab. 3, IAE_{sp} is minimum for the CMA-ES method, in all three FOPDT sample systems.

Table 3: Comparison of IAE_{sp} obtained from CMA-ES with ZN and TR

Sample system 1	$\tau = 0.35$					
	K_p	T_i	T_d	λ	μ	IAE_{sp}
ZN	0.5528	0.9536	0.3797	1.3607	1.0828	1.8327
TR	1.0281	0.9829	0.1698	1	1.2000	1.4736
CMA-ES	1.3571	1.0544	0.1701	1	1.1006	0.7868
Sample system 2	$\tau = 0.4212$					
	K_p	T_i	T_d	λ	μ	IAE_{sp}
ZN	0.6762	1.1426	0.3619	1.3198	1.0841	2.7109
TR	0.5534	0.9746	0.3566	1	1.2000	2.6390
CMA-ES	0.7867	1.1045	0.3150	1	1.1106	1.4562
Sample system 3	$\tau = 0.6710$					
	K_p	T_i	T_d	λ	μ	IAE_{sp}
ZN	0.9315	2.0084	0.4172	1.3244	0.9105	4.7749
TR	0.2911	0.9676	0.7492	1	1.2000	4.8330
CMA-ES	0.5107	1.2859	0.5364	1	1.1160	2.6707

The graphs in Fig. 2 confirm that the optimal controller parameters obtained using the CMA-ES algorithm are found to vary smoothly for τ in both the design objectives.

3.2 Data Analysis

The input and output variables of the SPT and LDR dataset are given in Table I for developing the MLAs. Based on Cook's distance, outliers present in the Kp parameter of the SPT dataset are identified and denoted by the red colour '+' (plus) symbol as shown in Fig. 3a data points after removing outliers are given in Fig. 3b. Similarly, outliers are removed in LDR dataset also.

The cross-correlation matrix of the four output variables for the SPT dataset after removing outliers is given in Tab. 4. The diagonal plots in Fig. 4 represent the histogram of the data distribution for each output variable. The lower diagonal plots represent the variation between the output variables. The upper diagonal

represents their cross-correlation values similar to that tabulated in Tab. 4. The font size is larger for large cross-correlation values while font size is smaller for lower values of cross-correlation in Fig. 4. The red-colored ‘***’ represents very large cross-correlation values among all.

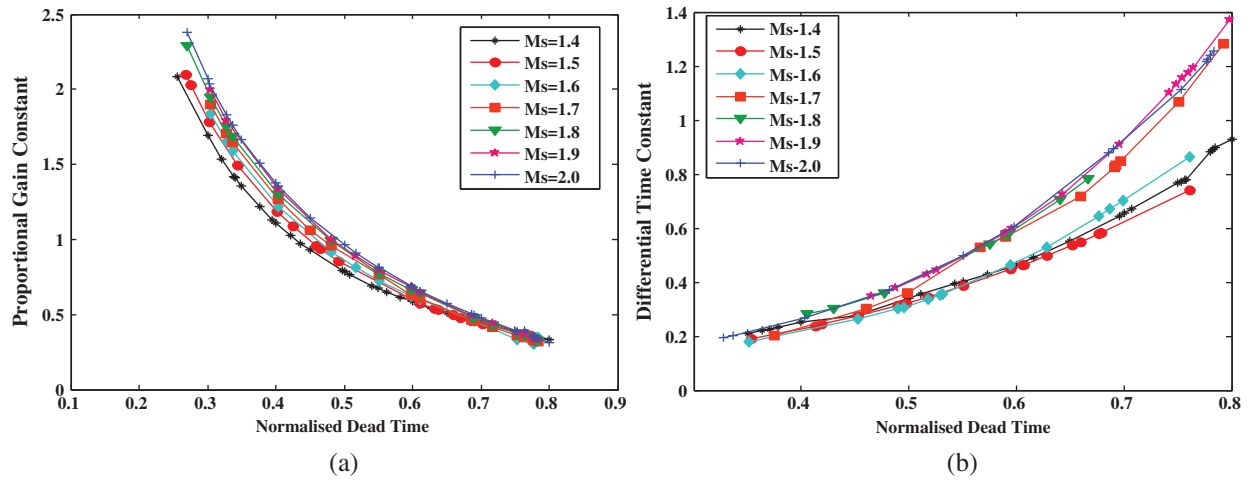


Figure 2: Parameter variation for various FOPDTs. (a) $K_p^{SP}Vs \tau$ for SPT (b) $T_d^{ld}Vs \tau$ for LDR

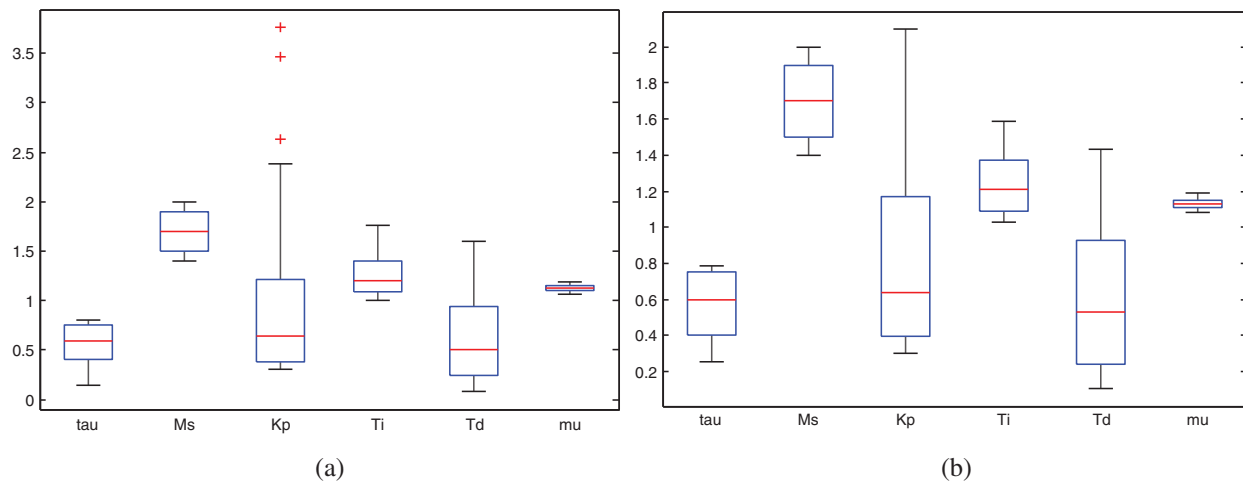


Figure 3: Box plot for SPT dataset. (a) With outliers (b) without outliers

Table 4: Cross correlation matrix for SPT dataset

Output variables	K_p^{SP}	T_i^{SP}	T_d^{SP}	μ^{SP}
K_p^{SP}	1.0000	-0.6503	-0.7461	0.0666
T_i^{SP}	-0.6503	1.0000	0.8020	0.0609
T_d^{SP}	-0.7461	0.8020	1.0000	-0.0047
μ^{SP}	0.0666	0.0609	-0.0047	1.0000

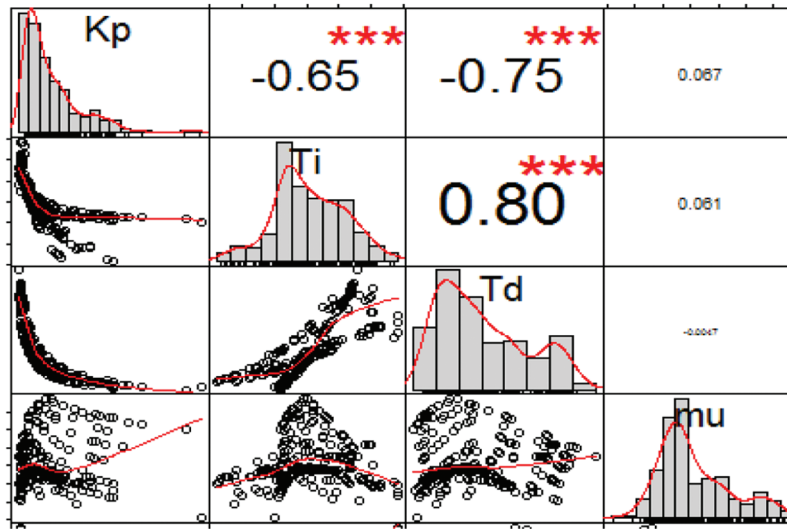


Figure 4: SPT dataset-cross correlation matrix

3.3 Development of MLOT Models

The three MLAs, MLOT-FFBPNN, MLOT-MLSSVR, MLOT-MVR algorithms for SPT, LDR, and CMB datasets are used to develop totally nine MLOT models. The MLOT models are tested using randomly generated data points named testing data points that are not involved in the training and validation phase. Figs. 5a and 5b shows the variation of T_d^{ld} , μ^{ld} with respect to τ^{ld} for MLOT-LDR.

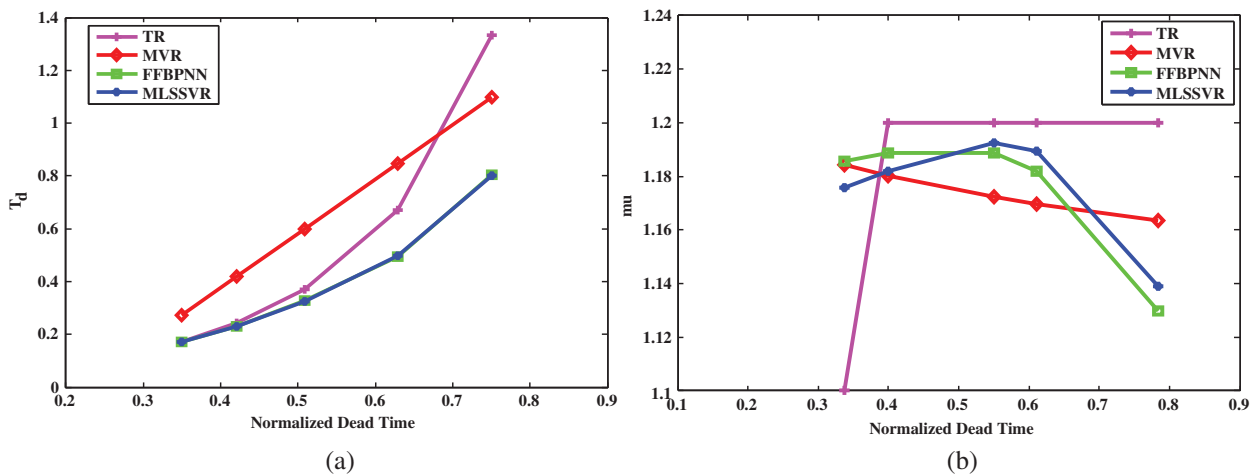


Figure 5: Validation results of three MLOT models compared with results from TR. (a) T_d^{ld} vs τ^{ld} from MLOT-LDR (b) μ^{ld} vs τ^{ld} from MLOT-LDR

Also, it can be observed that the results of testing data points in Fig. 6 follow the graphs of parameter variations similar to the validation data points in Fig. 5. Also, the results from TR show that controller parameters from FFBPNN and MLSSVR are closer to the optimum controller parameters than the results from TR.

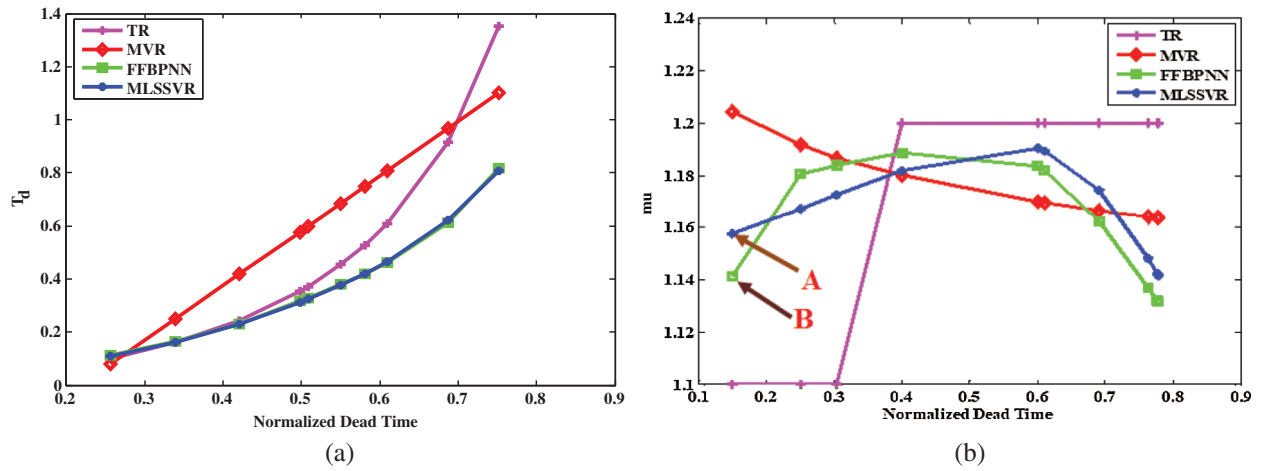


Figure 6: Testing results of three MLOT models compared with results from TR. (a) $T_d^{ld} V_{st}^{ld}$ from MLOT-LDR (b) $\mu^{ld} V_{st}^{ld}$ from MLOT-LDR

In graph $\mu^{ld} V_{st}^{ld}$ of Fig. 6b for $M_s = 2.0$, the point A indicates the value of μ^{ld} from MLSSVR (blue color) and point B from FFBPNN (green color) for the same value of $\tau^{ld} = 0.15$. But, point A and point B clearly indicate that FFBPNN (green color) does not follow the pattern as given in Fig. 5b when compared to MLSSVR (blue color). This indicates that the MLSSVR algorithm produces better results than FFBPNN.

The statistical performances of proposed MLOTs are better than TR and MVR as given in Tab. 5 for K_p and Tab. 6 for T_d .

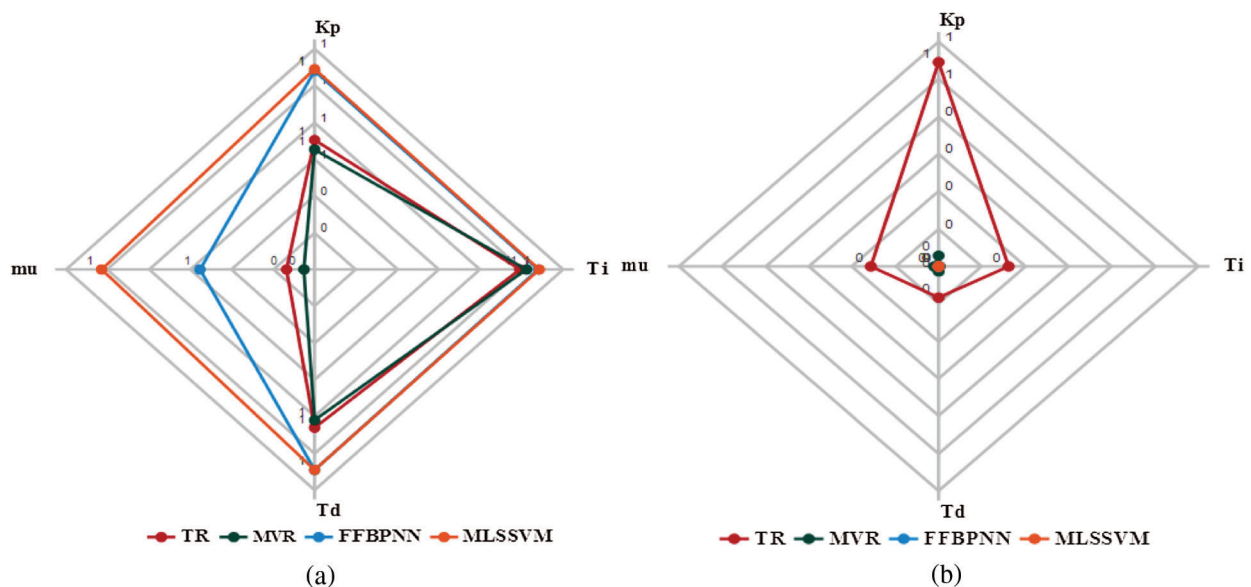
Table 5: Statistical analysis of the proposed MLOTs for K_p

Variables	K_p			
Method	TR	MVR	FFBPNN	MLSSVR
MLOT-SPT				
CC	0.6307	0.6992	0.9999	0.9999
MSE	1.6063	6.2×10^{-6}	0.0003	0.0004
RMSE	1.2674	0.0056	0.0172	0.3522
MAE	0.8444	0.0033	0.0106	0.2801
MLOT-LDR				
CC	0.6433	0.5974	0.9883	0.9966
MSE	0.6540	0.0339	0.0009	0.0003
RMSE	0.2584	0.1842	0.0292	0.0162
MAE	0.8125	0.1671	0.0168	0.0113
MLOT-CMB				
CC	–	0.6892	0.9914	0.9944
MSE	–	0.0579	0.0015	0.0010
RMSE	–	0.2406	0.0386	0.0311
MAE	–	0.2049	0.0263	0.0236

Table 6: Statistical analysis of the proposed MLOTs for T_d

Variables	T_d			
Method	TR	MVR	FFBPNN	MLSSVR
MLOT-SPT				
CC	0.692	0.6956	0.9977	0.9981
MSE	0.093	0.0005	0.0009	0.0004
RMSE	0.305	0.0218	0.0302	0.2268
MAE	0.2061	0.009	0.0161	0.1818
MLOT-LDR				
CC	0.7849	0.7501	0.9951	0.9969
MSE	0.0993	0.0171	0.0004	0.0002
RMSE	0.2650	0.1309	0.0202	0.0158
MAE	0.3561	0.1042	0.0128	0.0096
MLOT-CMB				
CC	–	0.7073	0.9888	0.9905
MSE	–	0.0416	0.0016	0.0013
RMSE	–	0.2039	0.0404	0.0365
MAE	–	0.1676	0.0277	0.0227

The statistical results of controller parameters are given as spider graphs in Figs. 7 and 8. For a better machine learning model, the CC values must be very large and the MSE, RMSE, MAE values must be the least. The CC values for all the output parameters in MLSSVR are larger compared to all other methods. This is shown by the outermost web denoted by red colour in Fig. 7a. Also, the MSE values of MLSSVR results are the least and it is given by the innermost web given in Fig. 7b denoted in red colour.

**Figure 7:** MLOT-LDR dataset. (a) CC (b) MSE

The CC values for the CMB dataset are compared among MLOT-MLSSVR, MLOT-FFBPNN, MLOT-MVR models in Fig. 8a. The MLOT-MLSSVR takes the outermost spider web (red colour) denoting larger correlations among the generated CMA-ES controller parameters and the controller parameters from validation data. Similarly, the results from MLOT-MLSSVR have the least MSE while using the CMB dataset. This is denoted by the innermost spider web given in red colour as shown in Fig. 8b.

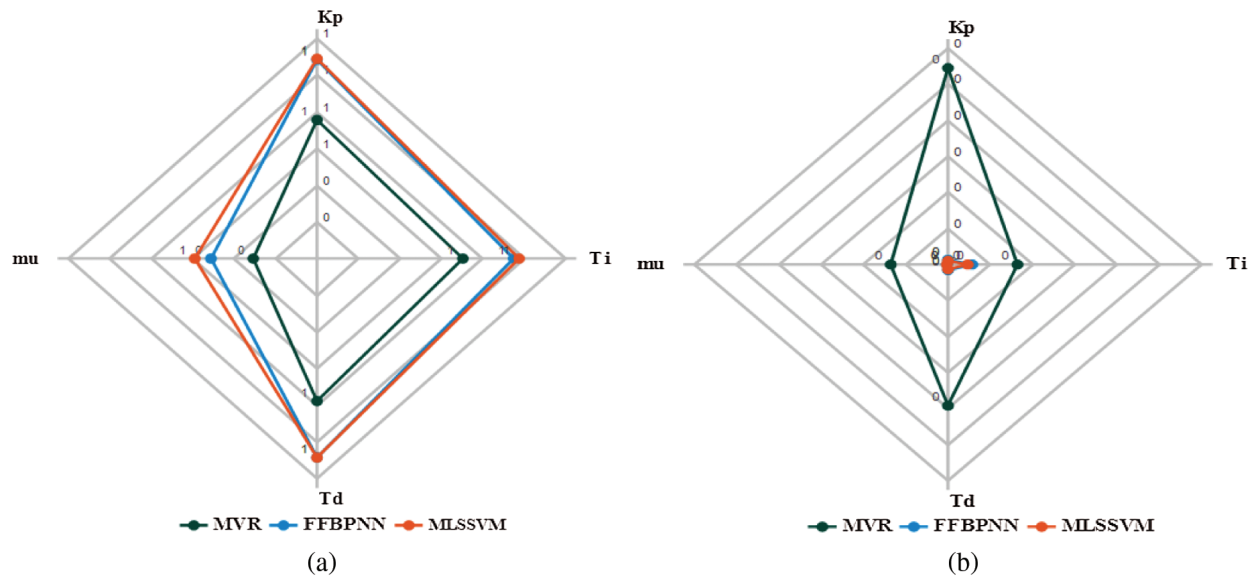


Figure 8: MLOT-CMB dataset. (a) CC (b) MSE

4 Conclusion

This paper proposes a novel FOPID controller tuning methodology using machine learning algorithms. Feed Forward Back Propagation Neural Network (FFBPNN), Multi Least Squares Support Vector Regression (MLSSVR) chosen to design Machine Learning based Optimal Tuner (MLOT) can handle the interdependency between the controller parameters and multiple outputs for multiple inputs. The statistical analysis values reveal that, the CC values of MLSSVR is 0.999, 0.9966, 0.9944 for MLOT-SPT, MLOT-LDR, MLOT-CMB for K_p . The MLOT-MLSSVR using the CMB dataset perfectly captures variations among controller parameters. The graphical analysis of parameter variations and statistical analysis confirms the better results for the proposed MLOT over the FOPID Tuning Rule (TR) given by Padula and Visioli. The proposed MLOT is also applicable for dead time-dominated systems and lag-dominated systems. Pre-processing with large amount of dataset consumes computational time which can be overcome using cloud computing and big data analysis in future.

Acknowledgement: The author with a deep sense of gratitude would thank the supervisor for his guidance and constant support rendered during this research.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] K. J. Åström and T. Häggglund, PID controllers: Theory, design, and tuning. In: *Automatic Tuning of PID Controllers*, 2nd ed., Research Triangle Park, NC: Instrument society of America, pp. 1–354, 1995.
- [2] A. O'Dwyer, *Handbook of PI and PID controller tuning rules*, 3rd ed., USA: Imperial college press, p. 624, 2009.
- [3] P. Shah and S. Agashe, "Review of fractional PID controller," *Mechatronics*, vol. 38, no. 7, pp. 29–41, 2016.
- [4] I. Podlubny, "Fractional-order systems and fractional-order controllers," *Institute of Experimental Physics, Slovak Academy of Sciences, Kosice*, vol. 12, no. 3, pp. 1–18, 1994.
- [5] C. A. Monje, Y. Chen, B. M. Vinagre, D. Xue and V. Feliu-Batlle, Fractional-order systems and controls: Fundamentals and applications. In: *Advances in Industrial Control*. London: Springer Science & Business Media, Springer, p. 415, 2010.
- [6] S. Das, S. Saha, S. Das and A. Gupta, "On the selection of tuning methodology of FOPID controllers for the control of higher order processes," *ISA Transactions*, vol. 50, no. 3, pp. 376–388, 2011.
- [7] R. V. Yohanandhan and L. Srinivasan, "Decentralised wide-area fractional order damping controller for a large-scale power system," *IET Generation, Transmission & Distribution*, vol. 10, no. 5, pp. 1164–1178, 2016.
- [8] S. Debbarma and A. Dutta, "Utilizing electric vehicles for LFC in restructured power systems using fractional order controller," *IEEE Transactions on Smart Grid*, vol. 8, no. 6, pp. 2554–2564, 2016.
- [9] I. M. Mehedi, U. M. Al-Saggaf, R. Mansouri and M. Bettayeb, "Two degrees of freedom fractional controller design: Application to the ball and beam system," *Measurement*, vol. 135, no. 3, pp. 13–22, 2019.
- [10] H. P. Ren, J. T. Fan and O. Kaynak, "Optimal design of a fractional-order proportional-integer-differential controller for a pneumatic position servo system," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 8, pp. 6220–6229, 2018.
- [11] M. H. Khooban, M. ShaSadeghi, T. Niknam and F. Blaabjerg, "Analysis, control and design of speed control of electric vehicles delayed model: Multi-objective fuzzy fractional-order PI λ D μ PI λ D μ controller," *IET Science, Measurement & Technology*, vol. 11, no. 3, pp. 249–261, 2017.
- [12] A. Asgharnia, A. Jamali, R. Shahnazi and A. Maheri, "Load mitigation of a class of 5-MW wind turbine with RBF neural network based fractional-order PID controller," *ISA Transactions*, vol. 96, no. 8, pp. 272–286, 2020.
- [13] D. S. Acharya and S. K. Mishra, "A multi-agent based symbiotic organisms search algorithm for tuning fractional order PID controller," *Measurement*, vol. 155, no. 9–10, p. 107559, 2020.
- [14] Y. Chen, T. Bhaskaran and D. Xue, "Practical tuning rule development for fractional order proportional and integral controllers," *Journal of Computational and Nonlinear Dynamics*, vol. 3, no. 2, p. 021403, 2008.
- [15] F. Padula and A. Visioli, "Optimal tuning rules for proportional-integral-derivative and fractional-order proportional-integral-derivative controllers for integral and unstable processes," *IET Control Theory & Applications*, vol. 6, no. 6, pp. 776–786, 2012.
- [16] F. Padula and A. Visioli, "Set-point weight tuning rules for fractional-order PID controllers," *Asian Journal of Control*, vol. 15, no. 3, pp. 678–690, 2013.
- [17] F. Padula and A. Visioli, "Tuning rules for optimal PID and fractional-order PID controllers," *Journal of Process Control*, vol. 21, no. 1, pp. 69–81, 2011.
- [18] I. H. Witten, E. Frank, M. A. Hall, C. J. Pal and M. Data, "Practical machine learning tools and techniques". *Data Mining*, vol. 2, pp. 4, 2005.
- [19] B. J. Perry, Y. Guo, R. Atadero and J. W. van de Lindt, "Streamlined bridge inspection system utilizing unmanned aerial vehicles (UAVs) and machine learning," *Measurement*, vol. 164, p. 108048, 2020.
- [20] R. Abdelaziz, M. Elhoseny, A. S. Salama and A. M. Riad, "A machine learning model for improving healthcare services on cloud computing environment," *Measurement*, vol. 119, no. 3, p. 117–128, 2018.
- [21] N. Hansen, The CMA evolution strategy: A comparing review. In: *Towards a New Evolutionary Computation*. Vol. 192. Berlin, Heidelberg: Springer, pp. 75–102, 2006.
- [22] M. H. Beale, M. T. Hagan and H. B. Demuth, Neural network toolbox user's guide. In: *Math Works Inc*. Natick, MA: Ver. 4, 2011.

- [23] D. Svozil, V. Kvasnicka and J. Pospichal, "Introduction to multi-layer feed-forward neural networks," *Chemometrics and Intelligent Laboratory Systems*, vol. 39, no. 1, pp. 43–62, 1997.
- [24] S. Xu, X. An, X. Qiao, L. Zhu and L. Li, "Multi-output least-squares support vector regression machines," *Pattern Recognition Letters*, vol. 34, no. 9, pp. 1078–1084, 2013.
- [25] X. Zhu and Z. Gao, "An efficient gradient-based model selection algorithm for multi-output least-squares support vector regression machines," *Pattern Recognition Letters*, vol. 111, no. 5, pp. 16–22, 2018.
- [26] C. Chatfield and A. J. Collins, Introduction to Multivariate Analysis. In: *Mathematics & Statistics*, 1st ed., Boca Raton: Routledge, pp. 1–248, 1980.
- [27] V. Valério, D. Duarte and J. S. Da Costa, "Tuning of fractional PID controllers with Ziegler-Nichols-type rules," *Signal Processing*, vol. 86, no. 10, pp. 2771–2784, 2006.