Lecture Notes in Networks and Systems 999

Nenad Filipović *Editor*

Applied Artificial Intelligence 2: Medicine, Biology, Chemistry, Financial, Games, Engineering

The Second Serbian International Conference on Applied Artificial Intelligence (SICAAI)

Springer

Lecture Notes in Networks and Systems 999

The series "Lecture Notes in Networks and Systems" publishes the latest developments in Networks and Systems—quickly, informally and with high quality. Original research reported in proceedings and post-proceedings represents the core of LNNS.

Volumes published in LNNS embrace all aspects and subfields of, as well as new challenges in, Networks and Systems.

The series contains proceedings and edited volumes in systems and networks, spanning the areas of Cyber-Physical Systems, Autonomous Systems, Sensor Networks, Control Systems, Energy Systems, Automotive Systems, Biological Systems, Vehicular Networking and Connected Vehicles, Aerospace Systems, Automation, Manufacturing, Smart Grids, Nonlinear Systems, Power Systems, Robotics, Social Systems, Economic Systems and other. Of particular value to both the contributors and the readership are the short publication timeframe and the world-wide distribution and exposure which enable both a wide and rapid dissemination of research output.

The series covers the theory, applications, and perspectives on the state of the art and future developments relevant to systems and networks, decision making, control, complex processes and related areas, as embedded in the fields of interdisciplinary and applied sciences, engineering, computer science, physics, economics, social, and life sciences, as well as the paradigms and methodologies behind them.

Indexed by SCOPUS, INSPEC, WTI Frankfurt eG, zbMATH, SCImago.

All books published in the series are submitted for consideration in Web of Science.

For proposals from Asia please contact Aninda Bose (aninda.bose@springer.com).

Nenad Filipović
Editor

Applied Artificial Intelligence 2: Medicine, Biology, Chemistry, Financial, Games, Engineering

The Second Serbian International Conference on Applied Artificial Intelligence (SICAAI)

Springer

Editor
Nenad Filipović
Faculty of Engineering
University of Kragujevac
Kragujevac, Serbia

ISSN 2367-3370 ISSN 2367-3389 (electronic)
Lecture Notes in Networks and Systems
ISBN 978-3-031-60839-1 ISBN 978-3-031-60840-7 (eBook)
https://doi.org/10.1007/978-3-031-60840-7

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Contents

Implementation of Hybrid ANN-GWO Algorithm for Estimation of the Fundamental Period of RC-Frame Structures

Filip Đorđević(✉) and Marko Marinković

Faculty of Civil Engineering, The University of Belgrade, Bulevar Kralja Aleksandra 73, 11000 Belgrade, Serbia
{fdjordjevic,mmarinkovic}@grf.bg.ac.rs

Abstract. The fundamental period (T_{FP}) of vibration is one of the most important parameters in structural design since it is used to assess the dynamic response of the structures. It is the time taken by a structure or system to vibrate back and forth in its most natural way, without any external forces applied. Simultaneously, T_{FP} depends on the mass distribution and stiffness of the structure, which is largely influenced by infill walls in RC frame structures, and which is why their careful design is necessary. This study aims to develop a fast, accurate, and efficient machine learning (ML) method for the prediction of the fundamental period of masonry-infilled reinforced concrete (RC) frame structures. Hybridization of the stochastic gradient descent (SGD) based artificial neural network (ANN), and meta-heuristic grey wolf optimization (GWO) algorithm is proposed as an effortless computational method. This approach provided even more reliable solutions than the robust second-order procedure based on single ML models. A total of 2178 samples of infilled RC frames were collected from available literature, where the number of storeys (*NoSt*), number of spans (*NoSp*), length of spans (*LoSp*), opening percentage (*OP*), and masonry wall stiffness (*MWS*) were considered as input parameters for predicting the output T_{FP} results. The accuracy and exploration efficiency of the proposed ANN-GWO paradigm have demonstrated superiority over existing seismic design codes and other conventional ML methods.

Keywords: Earthquake engineering · Machine learning · Artificial neural network · Grey wolf optimization · Infill frames

1 Introduction

Fundamental period of a structure is a starting point of every building design process in civil engineering. Therefore, its determination is of utmost importance. Its value determines the seismic load that will be used in the design of a building, thus it is essential to make a good estimation of its value. In the case of RC frame structures with masonry infill walls, this is a complex task. Infill walls are quite stiff and it influences the fundamental period significantly. In recent years, Artificial Intelligence (AI) techniques have been used increasingly in various engineering fields, including structural engineering [1]

N. Filipović (Ed.): AAI 2023, LNNS 999, pp. 1–6, 2024.
https://doi.org/10.1007/978-3-031-60840-7_1

and more specifically earthquake engineering [2, 3]. Existing studies are mostly based on shallow machine learning or Deep-Learning (DL) algorithms for solving regression, classification or optimization problems. This study aims to develop a fast, accurate, and efficient ML method for the prediction of the fundamental period of masonry-infilled (RC) frame structures.

2 Material and Methods

2.1 Database Description

A database consisting of 2178 masonry-infill RC frame structures was collected from the available literature FP4026 Research Database [4]. Table 1 contains information on the distribution of input and output variables, i.e. *NoSt*, *NoSp*, *LoSp*, *OP*, *MWS*, and T_{FP}. This is important because of the possibility to reproduce the results obtained in this study, but also to provide data normalization to the range $[-1 \div 1]$, which was done by using the MinMaxScaler function in the Python environment.

Table 1. Distribution description of input and output variables.

Variable	Mean	St. Dev.	Min	Max
NoSt [-]	11.50	6.35	1.00	22.00
NoSp [-]	5.76	0.87	2.00	6.00
LoSt [m]	4.77	1.45	3.00	7.50
OP [%]	31.76	28.99	0.00	75.00
MWS [10^5 kN/m]	11.38	7.85	2.25	25.00
T_{FP} [s]	0.83	0.59	0.04	3.01

Two metrics were used for the performance evaluation of the proposed methods including, mean squared error (*MSE*), and coefficient of determination (R^2) [5].

2.2 Artificial Neural Network (ANN)

ANNs are a class of machine learning models that are based on the organization and operation of the human brain. They have a number of key benefits, such as the capacity to learn from and adjust to new data, the capacity to process sizable amounts of data concurrently, and the capacity to model intricate, non-linear relationships between inputs and outputs. This study aims to predict the fundamental period of infill RC frames, using the output results from the optimization GWO technique as a starting point for ANN, in order to increase the security against falling into local minima. Validation of the results was performed by comparing the hybrid ANN-GWO model based on the SGD rule, with single first-order SGD model and second-order based adaptive moment estimation (ADAM) model developed from scratch [5]. The stability of the single model was tested using the 10-cross validation (CV) technique. The activation functions adopted for hidden and output layers are hyperbolic tangent and pure linear, respectively. SGD rule implemented in this work can be mathematically expressed as:

$$w_{i+1} = w_i - \mu \cdot g_i + \eta \cdot \Delta w_{t-1} \tag{1}$$

where $\mu = 0.1$ $[0 \div 1]$, and $\eta = 0.9$ $[0 \div 1]$ are learning rate and momentum as hyperparameters with values adopted according to the recommendations derived from other studies [5], g_i is the gradient calculated after each sample i, and $\Delta w_{t\text{-}1}$ is the weight increment from previous iteration. Based on previous studies of the same task [6–8] where ANNs either with one or multiple hidden layers were adopted, the authors of this work investigated several architectures with a single layer and different numbers of neurons. The adopted splitting strategy considers 80% of the samples for training and the other 20% for the test phase. The most optimal ANN-GWO architecture (5–8-1) contains eight neurons in the hidden layer.

2.3 Grey Wolf Optimization (GWO)

The social structure and foraging habits of grey wolves in the wild served as inspiration for the development of the GWO meta-heuristic optimization algorithm. GWO was first proposed by [9] in 2014. As gray wolves usually live in the pack, it is important to note their social hierarchy, which consists of 4 groups: alpha (α), beta (β), delta (δ), and omega (ω). The alpha wolf as the best manager in the pack is the most dominant member, while the beta, delta, and omega wolves have a lower rank, respectively. The optimization process involves updating the positions of the wolves in the search space based on their initially random location. Grey wolves traditionally have a strict hunting procedure which can be mathematically described as follows:

$$X(t+1) = (X_1 + X_2 + X_3)/3 \tag{2}$$

$$X_1 = X_\alpha(t) - A_1 \cdot D_\alpha, X_2 = X_\beta(t) - A_2 \cdot D_\beta, X_3 = X_\delta(t) - A_3 \cdot D_\delta \tag{3}$$

$$A_i = 2 \cdot a \cdot r_1 - a, C_i = 2 \cdot r_2 \tag{4}$$

$$a = 2 \cdot (1 - iteration/maxiteration) \tag{5}$$

$$D_{\alpha,\beta,\delta} = \left|C_i \cdot X_{\alpha,\beta,\delta}(t) - X(t)\right| \tag{6}$$

where t refers to the current iteration, X is the vector of the grey wolf position, X_1, X_2, and X_3 are predicted position vectors of the alpha, beta, and delta wolves, X_α, X_β, X_δ are relative position vectors of alpha, beta, and delta wolves, A_i and C_i are coefficient vectors, r_1 and r_2 are random vectors in range [0,1], a is a component that linearly decreased from 2 to 0, and $D_{\alpha,\beta,\delta}$ are vectors that depend on the position of the prey. One of the advantages of GWO is that it has a relatively simple algorithm, without the need to calculate function derivatives, which makes it easy to implement and understand. The fundamental principle behind using GWO for training ANNs is to find the weights and biases in a more optimal manner, by reduction of the *MSE* as a fitness function between the predicted and the actual outputs. The main benefit is its superior ability to explore the high-dimensional parameter space compared to some other optimization algorithms. The authors proposed the initial random generating of 30 wolves as search agents. In the training phase, GWO iterates 10 times, followed by only 320 ANN iterations (see Table 2), which is optimal compared to at least 1000 epochs necessary for the convergence of a single conventional model from scratch. The improvement of the positions of search agents is performed by applying the procedure described in the previous section. All search agents were separately considered as vectors with 57 parameters of the neural network that were initially randomly generated in the range $[-5 \div 5]$.

3 Results and Discussion

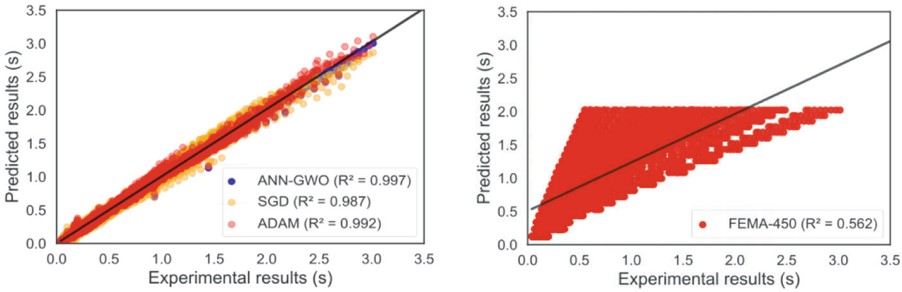
The results presented in Fig. 1, show that in the entire domain the hybrid ANN-GWO algorithm is superior over the pure SGD and ADAM models from scratch, as well as the FEMA-450 [10] design code that proposes the following equation for calculating the fundamental period:

$$T_{FP} = 0.0466 \cdot H_n^{0.9} \tag{7}$$

where H_n is the height of the structure in meters. Table 2 summarizes the performances of the mentioned approaches, whose results are illustrated in Fig. 1. It can be concluded that the hybrid algorithm has a slightly better processing power and faster convergence (320 versus 1000 epochs) even than the second-order ADAM algorithm. A similar conclusion is reached in the case of comparison with SGD from scratch, while the robustness of the hybrid procedure is particularly pronounced in comparison to the results of the FEMA-450 design code.

Table 2. Performance indicators of the proposed hybrid and single ML models

Measure	R^2				MSE $(\cdot 10^{-3} s^2)$			
Method	ANN-GWO	SGD	ADAM	FEMA-450	ANN-GWO	SGD	ADAM	FEMA-450
Iteration	320	1000	1000	–	320	1000	1000	–
Architecture	5-8-1			–	5-8-1			–
Train	0.997	0.987	0.992	–	1.178	4.735	2.771	–
Test	0.997	0.988	0.991	–	1.189	4.894	3.194	–
All	0.997	0.987	0.992	0.562	1.179	4.767	2.856	247.690



(a) ANN-GWO, SGD and ADAM models (b) FEMA-450 design code

Fig. 1. Performance comparison of the proposed hybrid ANN-GWO model with SGD/ADAM methods, and provisions of FEMA-450 design code on all data.

4 Conclusions

This paper suggests the implementation of a meta-heuristic GWO algorithm to enhance the processing power of the ANN-SGD, used for the prediction of the fundamental period of RC frame structures. Proof that the hybrid algorithm outperforms traditional procedures is made by comparing it with the first-order ANN-SGD and the second-order ANN-ADAM algorithm made from scratch, as well as seismic design codes. ANN-GWO provided a faster convergence, and a higher coefficient of determination values, making it more accurate even than the second-order algorithm. In addition, GWO enables more successful avoidance of falling into a local minimum, which is often a problem of single ANN models. Generally, all three approaches are suitable to make predictions of the fundamental period, with some differences in speed. As part of future research, it is necessary to examine the potential application of the GWO algorithm and its hybrid variations on a wider range of problems, in order to uncover new possibilities and validate its effectiveness.

References

1. Đorđević, F.: A novel ANN technique for fast prediction of structural behavior. In: 6th International Construction Management Conference, We Build the Future (2023)
2. Rahmat, B., Afiadi, F., Joelianto, E.: Earthquake prediction system using neuro-fuzzy and extreme learning machine. In: Proceedings of the International Conference on Science and Technology (2018)
3. Celik, E., Atalay, M., Kondiloğlu, A.: The earthquake magnitude prediction used seismic time series and machine learning methods. In: IV International Energy Technologies Conference (2016)
4. Asteris, P.G.: The FP4026 research database on the fundamental period of RC infilled frame structures. Data Br., pp. 704–709 (2016). https://doi.org/10.1016/j.dib.2016.10.002
5. Đorđević, F., Marinković, M.: Advanced ANN regularization-based algorithm for prediction of the fundamental period of masonry-infilled RC frames. Adv. Comput. (2023). submitted for review

6. Asteris, P.G., et al.: Prediction of the fundamental period of infilled RC frame structures using artificial neural networks. Hindawi Publishing Corporation, Comput. Intell. Neurosci., 5104907 (2016). https://doi.org/10.1155/2016/5104907
7. Asteris, P.G., Nikoo, M.: Artificial bee colony-based neural network for the prediction of the fundamental period of infilled frame structures. Neural Comput. Appl., 4837–4847 (2019). https://doi.org/10.1007/s00521-018-03965-1
8. Tran, V.L., Kim, S.E.: Application of GMDH model for predicting the fundamental period of regular RC infilled frames. Steel Compos. Struct., 123–137 (2022). https://doi.org/10.12989/scs.2022.42.1.123
9. Mirjalili, S., Mirjalili, S.M., Lewis, A.: Grey wolf optimizer. Adv. Eng. Softw., 46–61 (2014). https://doi.org/10.1016/j.advengsoft.2013.12.007
10. FEMA-450, NEHRP recommended provisions for seismic regulations for new buildings and other structures, Part 1: Provisions, Washington (DC), Federal Emergency Management Agency (2003)

Development of a Mathematical Model for Balloon Diameter Calculation in Percutaneous Transluminal Angioplasty Using Genetic Programming

Leo Benolić[1,2(✉)]

[1] Bioengineering Research and Development Centre (BioIRC), Prvoslava Stojanovica 6, 3400 Kragujevac, Serbia
leo.benolic@gmail.com
[2] Faculty of Engineering, University of Kragujevac, Sestre Janjica 6, 34000 Kragujevac, Serbia

Abstract. This paper describes the development of a mathematical model using genetic programming to calculate the diameter of a percutaneous transluminal angioplasty (PTA) balloon dilatation catheter for a given pressure and balloon size. The dataset used for the study was provided by Boston Scientific, and the genetic programming algorithm was implemented in Python using parallel computing. The results showed high levels of accuracy, with R2 values of 0.99989 and 0.99954 for the best and parsimonious models, respectively. The developed model can be useful for *in-silico* simulations of angioplasty surgery and can contribute to improving the effectiveness of the PTA balloon dilatation catheter procedure. This study demonstrates the potential of machine learning techniques for optimizing medical device performance and design. Further work could investigate the use of other machine learning techniques and larger datasets to enhance the accuracy and generalizability of the models.

Keywords: Peripheral Artery Disease · PTA balloon · pressure · genetic programming · symbolic regression · machine learning · in-silico simulations

1 Introduction

Peripheral Artery Disease (PAD) is a pervasive vascular condition that affects millions globally. Once termed peripheral vascular disease or lower extremity arterial disease, PAD is now the universally accepted terminology. It is characterized by the narrowing or blockage of arteries, particularly in the legs, due to atherosclerotic occlusions. Systemic atherosclerosis results in these obstructions, with plaque—comprising fat, cholesterol, and calcium deposits—being the primary culprit [1–3].

Despite its prevalence, PAD often remains undetected because many patients are asymptomatic. Intermittent claudication (IC), characterized by leg pain triggered by exercise and relieved by rest, stands out as the primary symptom. Yet, not all PAD patients exhibit this symptom, which occasionally results in misdiagnoses [2, 4].

N. Filipović (Ed.): AAI 2023, LNNS 999, pp. 7–20, 2024.
https://doi.org/10.1007/978-3-031-60840-7_2

Risk factors for PAD span both lifestyle choices and genetic predispositions. These include smoking, diabetes, hypertension, and dyslipidemia. While certain risks like tobacco use and high cholesterol are modifiable, others such as age, gender, and family history are not. Given PAD's widespread impact, which varies across income levels and ethnicities, there's a pronounced need for holistic treatment solutions [2, 4].

Diagnosing PAD is streamlined with tools like the Ankle Brachial Index. Still, advanced imaging, including computed tomography and MRI, is essential for detailed interventional planning. Treatments have evolved over time, focusing on enhancing patient quality of life and minimizing vascular complications. Interventional approaches, particularly percutaneous transluminal angioplasty (PTA), have seen remarkable advancements [1, 3].

PTA is a minimally invasive technique tailored for PAD patients, notably those with intermittent claudication or critical limb ischemia. It boasts a low complication rate and a high success rate. Yet, a significant challenge remains its restenosis rate. Stenting, especially using nitinol self-expanding stents, has risen as a robust alternative, especially for challenging cases. Such stents are known for their radial strength, shape-memory attributes, and minimal foreshortening. Research, including the Dutch Iliac Stent Trial and the Zilver PTX randomized trial, highlights stenting's efficacy, sometimes surpassing PTA. Consequently, while PTA is foundational for treating PAD, modern stenting techniques have enriched the treatment landscape, ensuring improved patient outcomes [5].

Machine learning (ML), a subset of artificial intelligence (AI), has been progressively integrated into various facets of medicine, manifesting its transformative potential. For instance, the Neuro Fuzzy hybrid model (NFHM) provides a cutting-edge approach in blood pressure classification by assimilating neural networks and fuzzy logic, demonstrating an aptitude for accurate hypertension and hypotension diagnoses [6]. Concurrently, in the realm of imaging, 3D convolutional neural networks are being employed to discern types of carotid arteries, albeit with a cautionary note on the vulnerabilities inherent to deep learning models [7]. In another domain, femoral peripheral artery disease (PAD) treatment, an advanced decision-support system has been pioneered using the radial basis function neural network (RBFNN). This system not only exhibited superiority over conventional models but also elucidated the impressive strides AI is making in augmenting clinical decision-making [8]. Complementing these advancements, Mistelbauer et al. unveiled a semi-automatic technique adept at identifying vessels in challenging PAD scenarios, reinforcing the versatility and efficacy of ML-driven methods across diverse medical applications [9]. Collectively, these studies underscore the burgeoning role and invaluable contributions of ML algorithms in the expansive landscape of medical diagnostics and therapeutics.

Genetic programming (GP), a specialized branch of artificial intelligence, has seen remarkable applications across the vast medical landscape. A study by [10] illustrated GP's capability in epidemiology, where it was employed to derive symbolic expressions that accurately estimated the epidemiology curve for the COVID-19 pandemic in the U.S. Similarly, GP's potential in radiology was demonstrated in study [11], where it aided in interpreting cervical spine MRI images, reaching a remarkable prediction accuracy of 90%. Beyond radiology, the method has also found utility in neurology, with research [12] utilizing Cartesian Genetic Programming to detect subtle symptoms of Parkinson's disease by analyzing patients' pen movements during a figure copying test. In the realm of bioinformatics, Hu [13] employed Linear Genetic Programming to evolve symbolic models predicting disease risks based on metabolite concentrations in blood, showcasing GP's intrinsic feature selection capability. Lastly, Bartsch [14] harnessed GP in oncology, developing classifiers to predict recurrence in Nonmuscle-Invasive Urothelial Carcinoma based on gene expression profiling. These diverse studies exemplify GP's profound versatility and potential in reshaping diagnostic, predictive, and epidemiological paradigms in medicine.

In the realm of angioplasty, the precise estimation of a balloon's current diameter based on its nominal diameter and pressure is paramount. The overarching aim of this study is to develop a mathematical model that can accurately predict the current diameter of an angioplasty balloon given its nominal diameter and the pressure applied.

The rationale behind striving for a mathematical model as opposed to a software-centric solution is multifold. Firstly, mathematical models offer universal applicability, uninfluenced by the constraints of programming languages. Secondly, they are immune to the limitations imposed by specific operating systems or hardware configurations. This transcends the barriers of technological specificity, allowing for a wider spectrum of usability.

The choice of Genetic Programming (GP) as the methodological backbone for this endeavor is grounded in its unique capabilities. Genetic Programming, rooted in the principles of evolutionary processes, possesses the ability to craft mathematical formulas using predefined structures. Instead of manually searching or tweaking parameters, GP autonomously evolves the optimal formula that best represents the underlying relationship between the variables. This evolution-inspired approach aids in circumventing the challenges associated with traditional mathematical modeling, delivering a solution that is both accurate and robust.

In light of these considerations, this study hypothesizes that through the application of Genetic Programming, we can derive a universal formula that, when fed with the nominal diameter and current pressure of a balloon, can predict its current diameter with high precision.This model can be useful for *in-silico* simulations of angioplasty surgery.

2 Materials and Methods

In this study, we utilized a dataset provided by the manufacturer Boston Scientific [15], which contains the diameter to pressure ratio for various sizes of the PTA balloon dilatation catheter. The dataset comprises balloon diameters ranging from 3 mm to 14 mm and various lengths, spanning from 40 mm to over 130 mm. During preprocessing, it was observed that for some balloons, the actual diameter at nominal pressure might slightly deviate from the designated diameter. For instance, a balloon labeled as 12 mm might not exactly measure this value at its nominal pressure. For the purpose of model approximation, our dataset was divided into two segments: 70% for training and 30% for validation/testing.

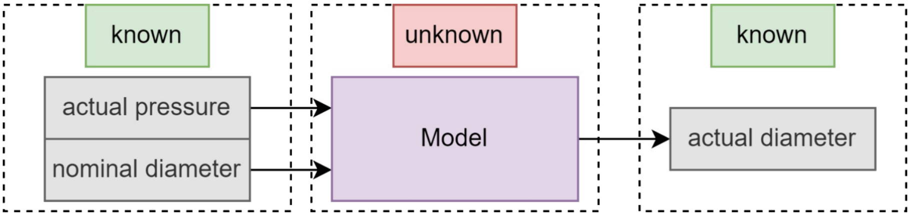


Fig. 1. Black-box model

In the evolving domain of machine learning, algorithms operate on the principle of learning patterns and relationships from data without explicit programming. Such algorithms can often be visualized as a 'black box' where inputs are transformed into outputs based on learned patterns, but the exact inner workings and relationships might not be immediately clear. Figure 1 showcases a typical black box model with two inputs, 'actual pressure' and 'nominal diameter', producing an output, 'actual diameter'.

While this black box representation simplifies the understanding of machine learning models, it is crucial to have methods that can deduce the nature of relationships inside this box. This is where Genetic Programming (GP) comes into play. GP, as an algorithmic approach, evolves and finds the best-fitted formula or relationship to map given inputs to outputs. By using GP, we can potentially unravel the 'equations' or 'formulas' governing the black box, offering us a more transparent view of how inputs are being transformed. This interpretability is especially crucial in fields like medicine, where understanding the rationale behind predictions is as important as the accuracy of the prediction itself.

Genetic Programming (GP) is a machine learning technique that draws inspiration from Charles Darwin's principles of natural evolution. It is a part of the broader family of genetic algorithms, which are metaheuristic methods that mimic the processes of evolution, including mutation, crossover, and reproduction.

In this study, the GP technique was applied using symbolic regression to derive mathematical models from the dataset. GP essentially creates a symbolic mathematical model visualized as a tree. Within this tree, the nodes represent various mathematical functions, while the leaves are either specific variables, such as X0, X1, X2, or constants. To provide a clearer picture, an equation like $y = X0 \times 3.4 + \sin(X1)$ can be depicted as a tree, as demonstrated in Fig. 2.

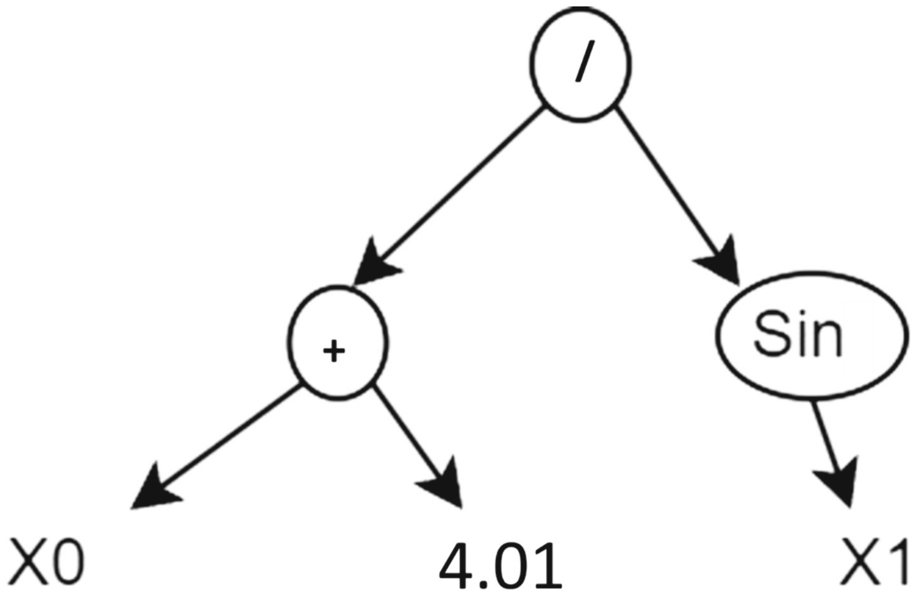


Fig. 2. Example of an equation tree

These mathematical trees are initially established through a process known as "Random Initialization", where the initial population is formed at random. Following this, parents are selected from this population based on specific fitness criteria. Once parents are determined, they undergo a series of genetic operations, with the offspring being produced through processes like mutation, crossover, and reproduction. This new generation, known as the "Offspring", is subsequently evaluated for their accuracy and relevance.

From this group, trees that demonstrate exceptional performance are chosen for the next cycle using a technique termed "Tournament Selection". These top-performing trees are the "Survivors" of this round. The algorithm evaluates whether to continue or halt based on certain "Stopping Criteria". If the criteria dictate continuation, the cycle loops back to the parent selection and proceeds as before. If the criteria suggest termination, the process ends. This cycle of evolution persists until a preset number of iterations are reached or other stipulated stopping points are met. A comprehensive depiction of the Genetic Programming process can be observed in Fig. 3.

In order to adjust the behavior of the algorithm, it is necessary to define the hyperparameter rules or at least their range (Table 1).

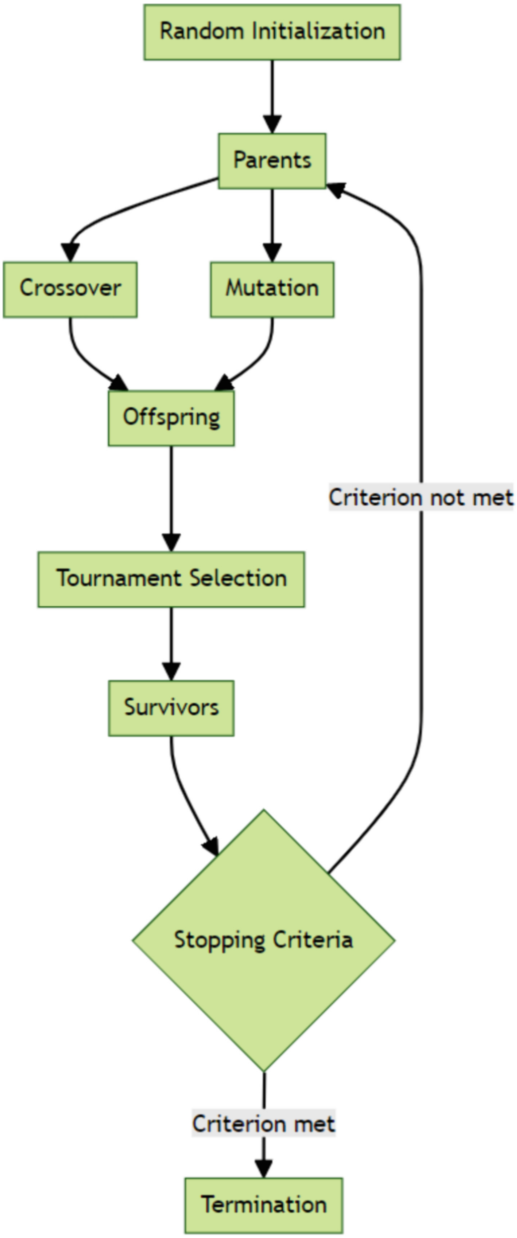


Fig. 3. Example of GP algorithm flow

Model quality estimation:
To gauge the effectiveness of the resulting model, we employ the coefficient of determination, R2. This metric serves as an indicator of how accurately the statistical model aligns with the observed data. Essentially, R2 represents the fraction of variance in the

Table 1. Hyper parameters

	Parameter range	
	min	max
Population	100	550
Generation	150	650
Tournament size	5	20
Crossover probability	0.9	0.97
Subtree mutation probability	0	0.09
Hoist mutation probability	0.01	0.07
Point mutation probability	0	0.09
Sum of probability	0.9	1
Constant range	−5000	5000
The minimal initial tree depth	3	8
The maximal initial tree depth	11	11
Stopping criteria coefficient	0,00001	
Parsimony coefficient	0.001	0.1
Metrics	RMSE	
Function set	add, sub, mul, div, sqrt, log, abs, neg, inv, max, min, sin, cos, tan	

dependent variable captured by the model. The Equation:

$$R^2 = 1 - \frac{SSR}{SST} \tag{1}$$

where: SSR is a Sum of Squared Regression (variation explained by model), and SST is Sum of Squared Total (total variation in data) [17].

3 Results and Discussion

In our exploration of the genetic programming algorithm's application, it became evident that computational efficiency is paramount, especially when dealing with complex datasets and the necessity for multiple algorithm runs. Consequently, the approach we took, as visualized in our diagram (refer to Fig. 4), centered around parallelizing our computations.

The decision to implement our algorithm in Python was bolstered by the availability and robustness of the gplearn 0.4.2 library. It provided us with a solid foundation to build our genetic programming routines. However, as with any computationally intensive task, optimization of runtime was a key challenge.

To address the efficiency concern, we integrated the multiprocessing library, a staple in Python's parallel computing arsenal. By deploying our GP algorithm concurrently on four separate processor cores, we were not only able to maximize our computer's resource utilization but also significantly cut down on training durations. This parallel execution method was visually depicted in a comprehensive diagram, which can be found in the Fig. 4 of this paper.

The system's unique design allowed each core to execute the GP algorithm independently, each initializing with distinct random states and parameters. This strategy proved invaluable, especially when it came to hyperparameter tuning. By repeatedly running the GP algorithm in parallel, we were granted multiple opportunities to refine our hyperparameters in a fraction of the time it would have taken on a single-core execution. This parallel computing approach, as facilitated by Python's multiprocessing library, has proven instrumental in enhancing the efficiency of our genetic programming endeavors. It underscores the importance of leveraging modern computing architectures and libraries to overcome challenges inherent in machine learning and data processing tasks.

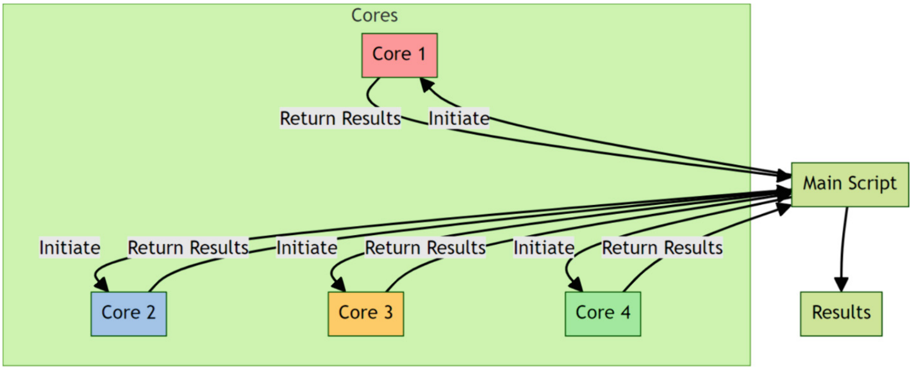


Fig. 4. Example of Multiprocessing Utilization

The results of the study show the curves of the obtained models for 5 mm and 3 mm balloons, as presented in Figs. 1 and 2, respectively. The graphs indicate the actual data points provided by the manufacturer. The deviations observed in the figures are less than 40 microns, which is less than 2% deviation. The accuracy of the models is high with R2 = 0.99989. These results demonstrate the effectiveness of using the genetic programming approach for developing the mathematical model for calculating the balloon diameter for a given pressure and balloon size. The developed model can be used for *in-silico* simulations of angioplasty surgery and can contribute to improving the effectiveness of the PTA balloon dilatation catheter procedure. However, further validation of the model using clinical data is necessary to assess its clinical applicability.

In the discussion, it can be explained that Figs. 5 and 6 show the syntax tree of the best performing model with an R2 of 0.99989 and a slightly inferior model with an R2 of 0.99954, respectively. The difference in performance was achieved by changing the parsimony coefficient, which penalizes models with more parameters during the evaluation process. This was done to avoid increasing the size of the models without any noticeable improvement in accuracy. The use of the parsimony coefficient helped to

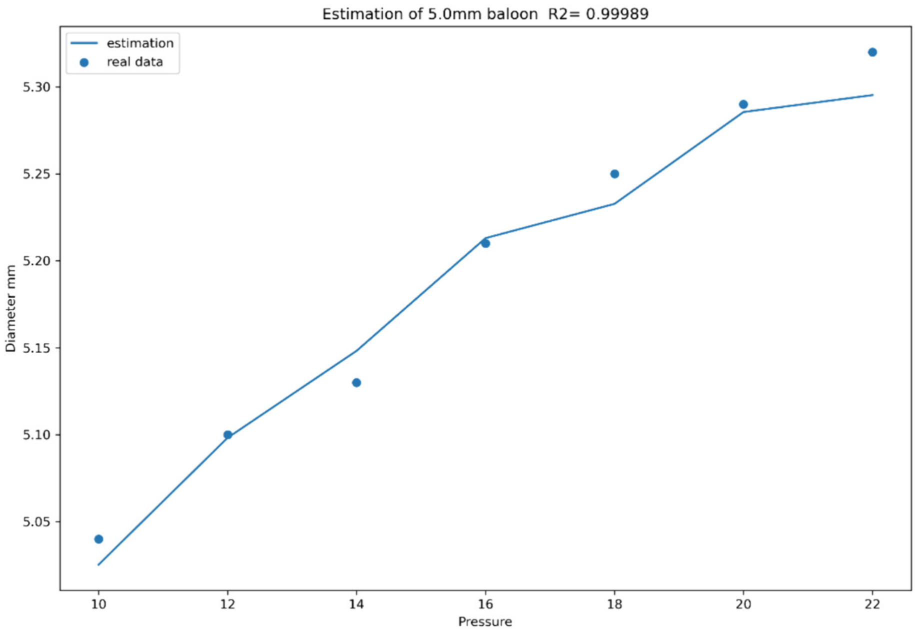


Fig. 5. Estimation of diameter for 5 mm Balloon

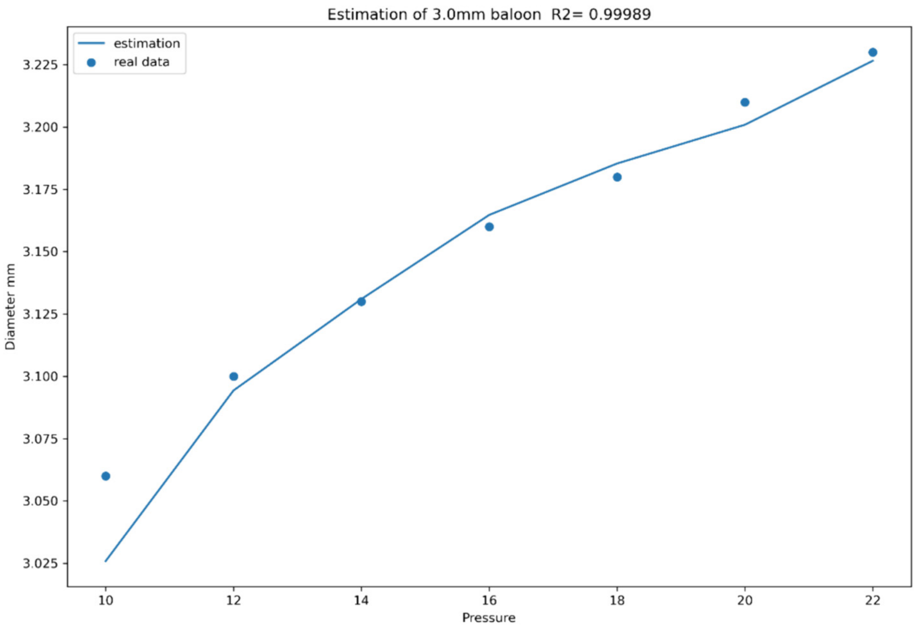


Fig. 6. Estimation diameter for 3 mm Ballon

ensure that the models remained computationally efficient while achieving high levels of accuracy (Fig. 7).

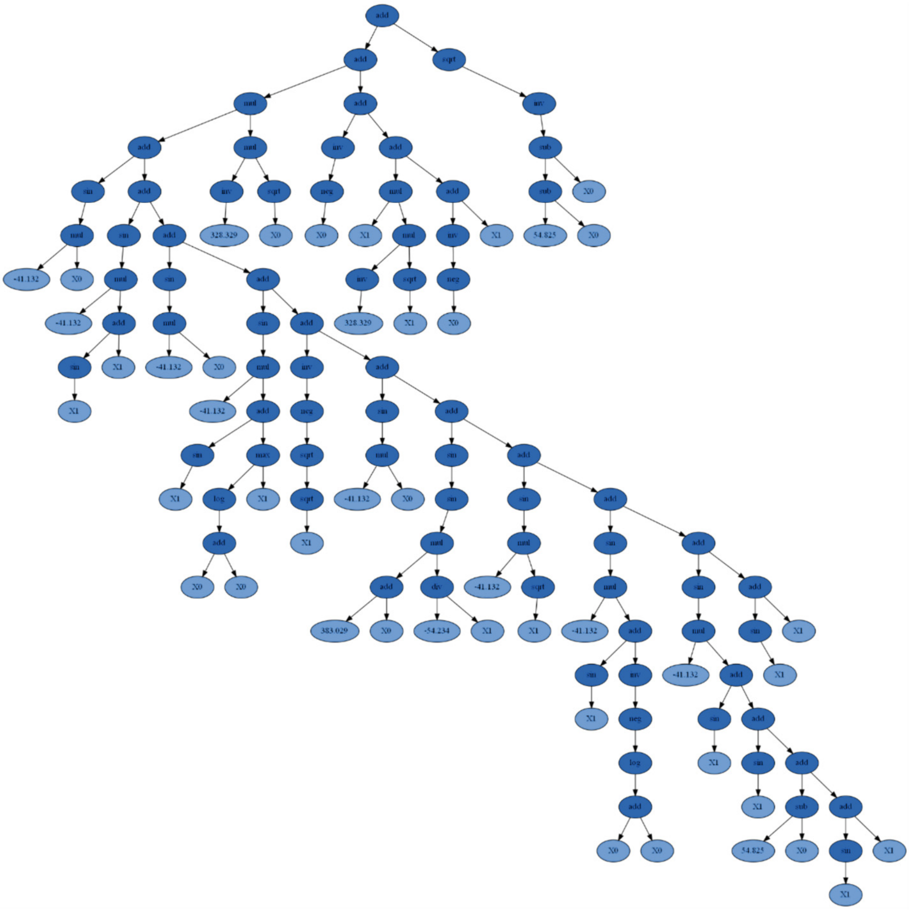


Fig. 7. Syntax tree of the best performing model with an R2 of 0.99989

An interesting feature of the model with an R^2 of 0.99954 (Fig. 8) is its reduced tree size. Although its performance is marginally lower, its simplicity is commendable. Indeed, the formula governing this model is so straightforward that it can be computed even with a basic calculator:

$$= X1 * \frac{sinsin\left(\sqrt{X0}\right) * log(X1^{-1})}{\sqrt{X0 - 48.522}} = X1 * \frac{sinsin\left(\sqrt{X0}\right) * log(X1^{-1})}{\sqrt{X0 - 48.522}} \quad (2)$$

This highlights one of the prime advantages of employing genetic programming. While complex models can achieve remarkable accuracy, simpler models derived from the

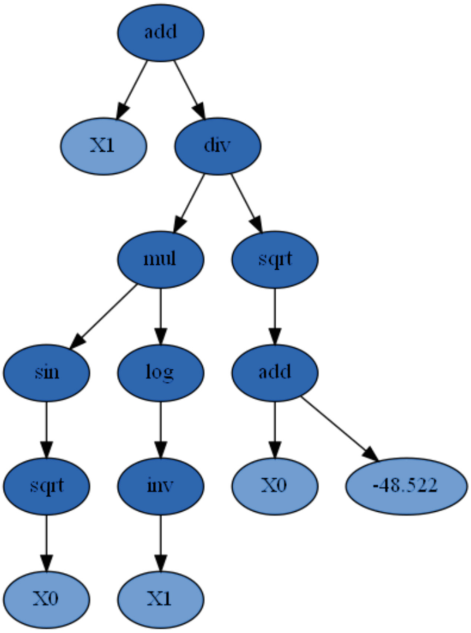


Fig. 8. Syntax tree of smaller model with an R2 of 0.99954

same methodology can be equally valuable, especially when the ease of calculation or resource constraints are considered.

Figures 9 and 10 expand on the applicability of our derived model to a broader range of balloon diameters. Figure 9 specifically showcases the estimated curves for diameters ranging from 3 mm to 8 mm, while Fig. 10 emphasizes on the larger spectrum, depicting diameters from 9 mm to 14 mm.

What is particularly noteworthy here is the inclusion of several non-standard diameters in both figures. These non-standard sizes, often overlooked in conventional representations due to their irregularity or rarity, have been effortlessly predicted using our genetic programming-derived model. The ability to forecast such unconventional diameters exemplifies the robustness and adaptability of our model, making it a comprehensive tool for diverse applications.