The Reconstruction of Blowing Pressure in Pipe Organ Using Machine Learning

Damian Tadeusz Węgrzyn
Polish-Japanese Academy of Information Technology
Warsaw, Poland *damian@wegrzyn.info*

Piotr Wrzeciono
Acoustic Measurement Laboratory, Warsaw University of Life Sciences
Warsaw, Poland *piotr_wrzeciono@sggw.edu.pl*

Alicja Wieczorkowska
Polish-Japanese Academy of Information Technology
Warsaw, Poland *alicja@poljap.edu.pl*

Abstract

The reconstruction of a pipe organ involves determining the blowing pressure. The lack of information about the pressure value may even result in irreversible damage to the pipes, as the adjustment of the sound parameters that depend on the pressure requires changing the physical structure of the pipes. In this paper, we provide a methodology for determining the blowing pressure in a pipe organ, and present a formula describing the air pressure in the pipe foot, depending only on the height of the pipe's cut-up and the fundamental frequency. We apply machine learning to determine the blowing pressure, based on the parameters of only a percentage of pipes. We found that the height of the cut-up and the fundamental frequency allow determining the blowing pressure. The more pipes, the higher the accuracy, but even 10% of pipes can be sufficient.

Keywords: pipe organ, blowing pressure, foot pressure, reconstruction, machine learning.

1. Introduction

Fire can destroy valuable works, as happened in Notre Dame de Paris. The historical pipe organ in St. Elizabeth's Church in Wrocław burned completely, and rebuilding it took years. The data for the reconstruction were obtained by analyzing historical sources and other preserved instruments built by the same organbuilder. However, such data are not always available; sometimes the instrument must be restored from its picture only.

A pipe organ has a fixed relative air pressure, i.e. pressure in the windchest relative to the respective atmospheric pressure of the environment. It is most often measured in millimeters of water gauge (mm H_2O). Relative pressure is a permanent attribute of the organ because all the pipes of the organ are adjusted to the appropriate pressure value, which affects the basic sound parameters of the pipes.

The value of air pressure in the pipe organ is essential not only during its construction but also in the reconstruction process. It is usually impossible to determine its value based on the preserved elements of the wind pressure system. In such a case, it is necessary to change the structure of the (usually antique) pipes, e.g., by cutting them or deforming the pipe's mouth. Thus, incorrect selection of the pressure by the organbuilder may result in the destruction of the instrument due to irreversible damage to the pipes.

In this work, we investigate the determination of blowing pressure to protect the reconstructed instruments and facilitate the reconstruction work of organbuilders. This is important because many damaged instruments require reconstruction, e.g., in the case of fire or warfare. So far, there are no methods to reproduce the pressure value used in a damaged pipe organ, based on the pipes alone.

2. Methodology

The purpose of this study is to create a model that, after receiving selected pipe attributes at the input, will indicate the blowing pressure at the output. To train a model that works in various situations, a sufficiently large set of input data is needed. Since the pressure value for pipe organs does not vary much and is usually in the range of 50-100 mm of water gauge, data from four instruments with different blowing pressures were used.

We generated one million input datasets, where each input dataset simulates one damaged instrument. Each dataset contains data from one instrument only, representing the percentage of randomly selected pipes from all pipes available in this instrument. Creating all possible subsets of k pipes within a single instrument with n pipes would require generating all possible variations without repetition V_n^k. Usually, in a pipe organ, $n >> 1000$. For example, if 50% of the pipes remain (k=500, n=1000), we would need about $3.3 \cdot 10^{1433}$ variations, and with 10% of the instrument remaining (k=100, n=1000), about $6 \cdot 10^{297}$ variations. Therefore, we have not conducted this research on all possible subsets, as such inputs are extreme big data. It is technically difficult to train a model on all variations due to the time required and/or memory limitations. On the other hand, there is no need to train the model on all possible subsets, thus we limited training to one million input datasets.

2.1. Solutions to Key Issues Encountered

Creating a million instruments to be used as input data causes problems. The first one is to ensure the uniqueness of randomly generated instruments, as obtaining unique sets of pipes within one instrument ensures that after dividing these data into randomly chosen train and test sets there is no overlap between them, and no train data are used in tests. This research uses a pseudo-random number generator to draw pipes to be included in datasets. To avoid, on the one hand, the generator falling into periodicity and, on the other hand, possible repetitions of pipe sets, a mechanism of indexing pipes in sets was implemented, ensuring the uniqueness of pipe sets.

We encountered various technical problems when developing our software. To speed up the calculations, each regressor we used has been rewritten to a multi-threaded version, using the Executor class and Lambda expressions in Java. Unfortunately, regressors from the Weka library in multi-threaded training require synchronization of model threads and the input data, which significantly increases the training time. Additionally, we had to deal with the RAM usage problem, to avoid program stopping because of running out of memory with the increase of the number of pipe sets. Therefore, RAM monitoring and memory cleaning via manual control of the garbage collector were used.

2.2. Programming Language and Libraries

We chose Java language for its memory management. Firstly, we needed a statically typed language (with variable types assigned before using them), to accurately reserve the space for variables and optimize memory usage. Secondly, we needed a mechanism of immutable objects which ensures that the object remains permanent after its creation, to achieve secure and efficient memory management. In addition, Java provides efficient support for multithreading, which allows parallel execution of different tasks.

The Application Programming Interface (API) of the Weka library, version 3.9.6, was used in the software development process. We chose the Weka platform due to its constantly updated API and a large collection of implemented machine learning (ML) methods. All regression algorithms implemented in Weka were tested, in various configurations and with various hyperparameter settings, to obtain the best solution.

Additionally, we used the Deeplearning4j library, version 1.0.0-M2.1, and the Nd4j sub-module, which allows loading, executing, and retraining TensorFlow models. We used several popular artificial neural network (ANN) models for regression problems, in various hyperparameter configurations. The results obtained using these models were compared with the results obtained using ML algorithms from the Weka library, see Section 4.

3. Our Data

Based on our previous research, we use four input attributes, the values of which were different for each pipe: cut-up height, airflow velocity, fundamental frequency, and air pressure in the pipe's foot. The input data used to train and test models represent either measured or calculated values. The output value is the blowing pressure, measured in millimeters of water gauge; this value is predicted using ML and deep learning. Our data describe 186 pipes representing 20 voices from four complete instruments. The blowing pressure is constant per instrument. In the case of incomplete instruments, when bellows and a significant percentage of pipes are missing, it is difficult to determine the blowing pressure, and the trial-and-error method may lead to the destruction of historical pipes.

The first (measured) attribute is the height h of the pipe mouth's cut-up, in millimeters, see Figure 1. The values of this attribute were taken from data provided in The Diapason [2, 3, 4, 5, 6], a journal devoted to organ and church music.

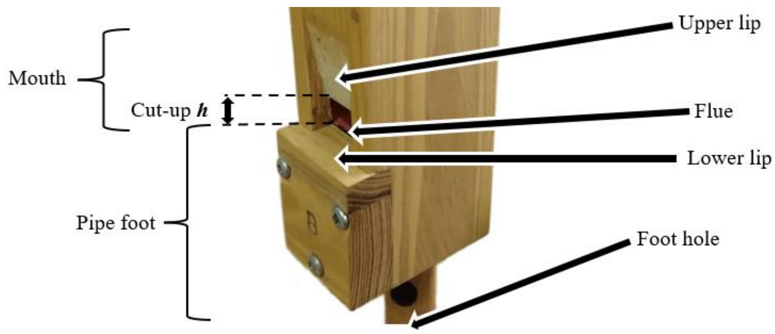

Fig. 1. Construction of a flue pipe with the height of the cut-up h indicated

The second (calculated) input attribute is the fundamental frequency of the pipe's sound. We used a musical interval to calculate the third attribute, which is the ratio of frequencies, constant and equal to $\sqrt[12]{2}$ for consecutive semitones in the twelve-tone equal-tempered scale, and tuned relative to a standard pitch A (440 Hz).

The third (calculated) input attribute is the velocity v of the airflow in a pipe, in meters per second. A constant value of the Strouhal number $S_t = 0.2$ was assumed for the calculations, as we found in our previous work [11] that it is approximately stable for labial pipes. The airflow velocity v in a flue pipe is calculated as

$$v = F_0 \cdot h / S_t \tag{1}$$

where F_0 is the fundamental frequency of the pipe's sound (Hz). In addition, we assume that the pipe was properly voiced, which means that the pipe sounds as intended (without beats). To achieve proper voicing, the edge tone produced in the cut-up must be of the same frequency as the resonator.

The fourth (calculated) input attribute is the air pressure in the pipe's foot p_p. This value differs from the blowing pressure and varies between pipes. We used the following formula to determine the pressure p_p in the foot of a flue pipe, which depends on the pipe mouth's cut-up height h, and the fundamental frequency of the pipe's sound F_0:

$$7.25 \cdot \sqrt[1.401]{p_p} - p_p + 4.38 \cdot F_0^2 \cdot h^2 = 0 \tag{2}$$

This parameter uses both the second and the third parameter, in a non-linear equation.

The pressure calculated using Eq. (2) does not take into account losses in the airflow and indicates an absolute pressure. To determine the value of relative pressure, used in blowing pressure measurement, the ambient pressure must be subtracted from the calculated pressure p_p, and the units converted to millimeters of water gauge.

Solving Eq. (2) is not trivial, but can be done numerically or graphically. Based on the graphical solution, we can find that Eq. (2) has a unique solution, but this is an inaccurate method. In this work, we used the Newton-Raphson (tangent) method as an iterative numerical method for finding the zeros of functions in a given range.

4. Results

We compared all trained models using the following evaluation metrics [7]: Pearson's correlation coefficient (r), mean absolute error (MAE), root-mean-square error (RMSE), relative absolute error (RAE), root relative square error (RRSE), mean absolute percentage error (MAPE), and the accuracy of model predictions (ACC), calculated as a percentage ratio of number of predictions such that $|y_i - x_i| \leq 3$ (where x_i is an actual value and y_i is the predicted value), to the total number of predictions. We used this ACC measure because the admissible error in the blowing pressure is $\leq$ 3 mm H_2O. It is caused by the measurement error made by the organbuilder, and the differences in the height of various components of the wind system relative to the ground.

The most important metric in the assessment was the ACC. The models that performed best were the Random Forest (RF) and the Multilayer Perceptron (MLP). Table 1 presents evaluation results for the best three models, for four percentages of the pipes drawn into the pipe set (simulating the remained pipes in an instrument), namely: 75%, 50%, 30%, and 10% of the full set of pipes.

Table 1. The evaluation of the top three ML algorithms for training on 75%, 50%, 30%, and 10% of all pipes

Remaining pipes	Rate	Model	r	MAE	RMSE	RAE [%]	RRSE [%]	MAPE [%]	ACC [%]
75%	1	Random Forest	0.9748	0.8026	1.8425	13.3937	22.3954	1.04	96
	2	Multilayer Perceptron	0.6917	4.4587	9.3577	74.4061	113.7408	5.01	79
	3	Perceptron	0.5401	4.9927	8.3555	83.3173	101.5594	5.23	74
50%	1	Random Forest	0.9500	0.9240	2.4315	15.4078	31.0516	1.27	93
	2	Multilayer Perceptron	0.8185	4.5038	8.5188	78.3698	108.6010	5.06	84
	3	SMOreg	0.5442	3.3364	6.9143	55.6211	88.2979	4.07	72
30%	1	Random Forest	0.8643	2.0476	4.1983	33.2862	50.2947	2.62	85
	2	Multilayer Perceptron	0.6269	4.8913	8.5585	78.7294	101.8898	5.41	83
	3	Recurrent Neural Network	0.4389	4.5010	8.4934	72.7459	98.9981	5.49	68
10%	1	Multilayer Perceptron	0.4712	4.4961	8.1038	73.3760	100.0896	5.10	84
	2	SMOreg	0.0861	4.4882	9.2748	71.1717	111.8923	4.99	76
	3	Convolutional Neural Network	0.1247	4.5639	9.6412	73.9562	113.2431	5.12	75

The training was repeated several times (up to 50 repetitions for RF) and similar values of evaluation measures were obtained, which confirms the repeatability of the results. We built a RF with 200 decision trees with a maximum depth of 5 and the number of attributes to randomly investigate set to 4, but even a RF with only 10 trees yielded good results, thus we present results for this small RF, which are even better than for 200 trees.

In addition, we analyzed the importance of individual input attributes in our best RF model. The most important one is the cut-up height (58.48%) and the second one is the fundamental frequency (25.56%), which is consistent with organ building knowledge.

5. Related Research and Discussion

The air pressure in the foot of the pipe depends mainly on the blowing pressure, but also on the geometric dimensions of the foot hole and the flue. [9] experimentally confirmed the relationship between the air pressure in the foot of the pipe and the size of the foot hole. [1] measured the pressure in the foot of the pipe using a pressure sensor. Pipes with similar geometrical features, used in their research, had pressure values similar to the values in our work, which confirms the correctness of the calculated pressure values.

The influence of changing the blowing pressure on the sound generated by pipes is well-known. The change of the blowing pressure changes the amplitude of the generated sound, its pitch and timbre. [10] confirmed the increase in sound frequency with increasing blowing pressure and vice versa. With the decrease in blowing pressure, the sound becomes darker and duller. If pressure increases, a discontinuity of pressure at the mouth is observed, caused by the centrifugal force of the curvilinear flow.

A fixed value of blowing pressure allows for a strong and clear sound. The blowing pressure cannot change significantly, and if the pressure is too high, overblowing, typical of wind instruments, occurs. [8] also describes the feedback cycle operating regime for pipe blowing, which includes cut-up, blowing pressure, and the flue width. Therefore, tuned pipes are adjusted to a limited blowing pressure range.

6. Conclusions

The issue of restoring pressure in a pipe organ has been so far an unsolvable problem. In most cases, the restored organs are incomplete, with bellows and a high percentage of pipes missing. The proposed solution based on ML and ANN models yields high accuracy and confirms the possibility of determining the blowing pressure. Our methodology is currently the only alternative to the trial-and-error method, that may destroy the historical pipes, as there is no method to calculate the blowing pressure.

Eq. (2) in this work proposes a formula describing the air pressure in the labial pipe's foot depending only on its fundamental frequency and cut-up height, with no other variables. We also confirmed the relationship between flue pipe attributes and blowing pressure. We found that the height of the cut-up is the most important feature. Fundamental frequency is also important, as it affects the proper voicing of a pipe. These two attributes suffice to determine the blowing pressure (the others can be calculated), and their importance is confirmed in RF.

References

1. Außerlechner, H.J., Trommer, T., Angster, J., Miklós, A.: Experimental jet velocity and edge tone investigations on a foot model of an organ pipe. J. Acoust. Soc. Am. 126 (2), 878-886 (2009)
2. McNeil, M.: 1863 E. & G. G. Hook Opus 322 Church of the Immaculate Conception Boston, Massachusetts Part 1. The Diapason 1292 (7), pp. 17-19 (2017)
3. McNeil, M.: The 1864 William A. Johnson Opus 161, Piru Community United Methodist Church Piru, California, Part 1. The Diapason 1305 (8), pp. 16-20 (2018)
4. McNeil, M.: The 1864 William A. Johnson Opus 161, Piru Community United Methodist Church Piru, California, Part 2. The Diapason 1306 (9), pp. 20-25 (2018)
5. McNeil, M.: The 1864 William A. Johnson Opus 161, Piru Community United Methodist Church Piru, California, Part 3. The Diapason 1307 (10), pp. 26-28 (2018)
6. Redman, R.: Louis Debierre Choir Organ, 1884. The Diapason 1298 (1), pp. 18-19 (2018)
7. Shcherbakov, M.V., Brebels, A., Shcherbakova, N.L., Tyukov, A.P., Janovsky, T.A., Kamaev, V.A.E.: A survey of forecast error measures. World Appl. Sciences J. 24, 171-176 (2013)
8. Steenbrugge, D.: Flue organ pipe operating regimes and voicing practices. In: Acoustics 2012, France, Nantes (2012)
9. Steenbrugge, D.: Fluid mechanical aspects of open-and closed-toe flue organ pipe voicing. International J. of Sustainable Construction and Des. 2 (2), 284-295 (2011)
10. Steenbrugge, D., De Baets, P.: Aerodynamics of flue organ pipe voicing. International J. of Sustainable Construction and Des. 1 (1), 162-173 (2010)
11. Węgrzyn, D., Wrzeciono, P., Wieczorkowska, A.: The Dependence of Flue Pipe Airflow Parameters on the Proximity of an Obstacle to the Pipe's Mouth. Sensors 22 (1), 10 (2022)