Acústica 2020
XI Congreso Ibérico de Acústica
51º Congreso Español de Acústica
TECNIACÚSTICA 2020

MULTI-MODAL TUNING OPTIMIZATION OF VIBRATING BARS WITH SIMPLIFIED UNDERCUTS USING AN EVOLUTIONARY ALGORITHM

Filipe Soares[1], José Antunes[1], Vincent Debut[2]

[1] Instituto Superior Técnico – Centro de Ciências e Tecnologias Nucleares – Laboratório de Dinâmica Aplicada
{filipe.soares@ctn.tecnico.ulisboa.pt, jantunes@ctn.tecnico.ulisboa.pt}

[2] Instituto de Etnomusicologia, Música e Dança, Faculdade de Ciências Sociais e Humanas, Universidade Nova de Lisboa, 1069-061 Lisboa, Portugal
{vincentdebut@fcsh.unl.pt}

Resumo

Apresentamos um método para afinar as frequências naturais de uma barra genérica para um instrumento de percussão de lâminas (e.g. marimba, vibrafone) para um conjunto de frequências alvo predefinidas usando um algoritmo de otimização global. A barra é modelada como uma viga unidimensional livre nas duas extremidades e suas frequências naturais são calculadas através de elementos finitos. O corte de perfil é composto por uma série de cortes retangulares, que além de reduzir a dimensão do problema de otimização para apenas alguns pares de variáveis (altura e comprimento de cada corte), também gera formas fáceis de fabricar. Adicionalmente, dois termos de penalidade são adicionados à função objectivo, para (1) minimizar a quantidade de material removido e (2) minimizar mudanças abruptas na altura do perfil, de forma a aliviar a complexidade a acelarar o processo de manufactura. Os resultados ilustram o efeito das diferentes penalidades nas soluções obtidas e, adicionalmente, perfis optimizados para várias afinações exiginentes e pouco ortodoxas são comparados com resultados previamente publicados, demonstrando os beneficios do corte simplificado.

Palavras-chave: multi-modal, afinação, barra, optimização, evolutivo.

Abstract

We present a method to tune the natural frequencies of a generic bar for a mallet percussion instrument (e.g. marimba, vibraphone) to a set of predefined target frequencies using a global optimization algorithm. The bar is modelled as a one-dimensional beam free at both ends and its natural frequencies are calculated via 1-D finite elements. The undercut is made up of a series of rectangular cuts, which aside from reducing the dimension of the optimization problem to a few pairs of variables (height and length of each cut) also generates shapes that are easy to manufacture. Moreover, two penalty terms are added to the objective function, in a weighted manner, to (1) minimize the amount of extracted material and (2) minimize abrupt changes in profile height, aimed to alleviate the complexity and accelerate the manufacturing process. The results illustrate the effect of the different penalties on the solutions obtained. Additionally, optimized shapes for various unorthodox and demanding tuning targets are presented and compared to previously published results, illustrating the benefits of the simplified undercut model.

Keywords: multi-modal, tuning, bar, optimization, evolutionary.

PACS nº 43.40.Cw, 43.75.Kk

1 Introduction

Unlike a vibrating string, the natural frequencies of a uniform rectangular bar are not harmonically related (i.e. their frequencies are not integer multiples of the lowest/fundamental frequency), a feature common to tonal musical instruments (Figure 1).

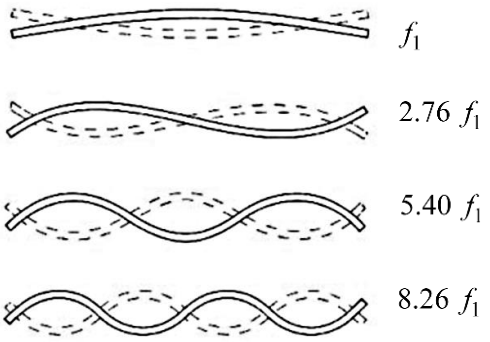

Figure 1 - First four transversal modes of vibration of a rectangular bar with uniform cross-section and their corresponding frequency ratios [1].

Hence, the tuning of idiophone bars, such as marimbas or vibraphones, is generally pursued via an undercut on the bars cross-section, as seen in Figure 2, such that the frequencies of the first few natural modes of vibration become harmonically related. Typical tuning ratios for different instruments are (1:4:10), (1:4:9) or (1:3:6). The aim of this work is to use global optimization techniques to find undercut shapes that are suitable for a generic bar to be tuned to a set of specified target frequencies.

Figure 2 - Typical undercut of a marimba/vibraphone bar.

A handful of studies can be found in literature on the tuning of idiophone bars (see for example [2], [3], [4], [5] and [6]). The studies in [7], [8] and [9] additionally report on experimental results using various tuning approaches. In [6], the authors use a genetic algorithm on a 3D finite element model, where the variables specify bar thicknesses on a 2-D mesh. This leads to an optimization problem with hundreds of variables (computationally expensive) and solutions with exceedingly complex geometries. In [3], which similarly uses global optimization techniques (simulated annealing), the authors use a 1-D model and reduce the dimension of the problem by representing the shape of the undercut in terms of a set of orthogonal shape functions (Fourier/Chebyshev). The latter approach has the merit of enabling a wide variety of undercut shapes using only a few variables and gives it the ability of finding optimized shapes, with smooth profile changes, for very diverse tuning ratios. However, in both cases, the resulting undercut shapes are curvilinear or present exceeding complex geometries, and hence are significantly more difficult to manufacture. With the aim of developing a similarly versatile model, which accounts for the ease of manufacture we focus here on rectilinear undercuts.

2 Model Description

The bars can be modelled as a beam, free at both ends. In many cases, particularly for marimbas, the slender beam assumption made by the simpler Euler-Bernoulli beam model may incur some errors at high order modes. Hence, we use here the Timoshenko's beam equations [10] where the transverse displacement $y(x,t)$ and slope $\phi(x,t)$ of small amplitude vibrations are described by

$$\begin{aligned} \rho A(x)\frac{\partial^2 y}{\partial t^2} + kGA(x)\left(\frac{\partial \phi}{\partial x} - \frac{\partial^2 y}{\partial x^2}\right) = 0 \\ \rho I(x)\frac{\partial^2 \phi}{\partial t^2} - EI(x)\frac{\partial^2 \phi}{\partial x^2} + kGA(x)\left(\phi - \frac{\partial y}{\partial x}\right) = 0 \end{aligned} \tag{1}$$

where $A(x) = bH(x)$ and $I(x) = bH^3(x)/12$ are the cross-sectional area and second moment of inertia, respectively (with b the bar's width and $H(x)$ its local thickness); ρ is the density, E is the Young's modulus, $G = 2/[2(1+\nu)]$ is the shear modulus (ν is the Poison's ratio) and k is a geometric factor for the shear energy (equal to $5/6$ for rectangular cross-sections).

2.1 Undercut Model

For simplicity, we define a symmetric undercut consisting of a series of rectangular cuts, as shown in Figure 3. This undercut is defined by $2N$ degrees of freedom (where N are the number of cuts), namely their lengths, λ_n, and the associated local heights, h_n.

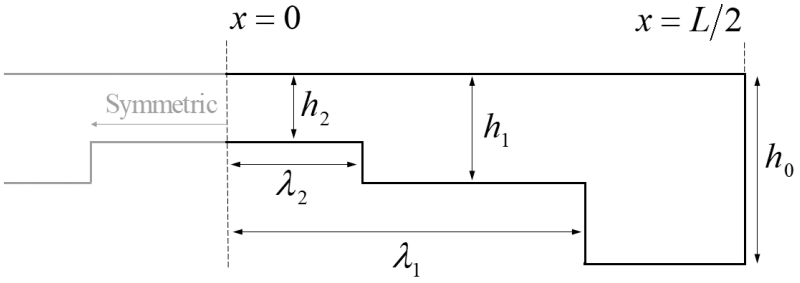

Figure 3 - Schematic description of the simple rectangular undercut ($N = 2$).

This choice of geometry leads to a set of upper and lower bounds for each variable:

$$\begin{aligned} &\text{Lengths:} \quad 0 \le \lambda_n \le L/2 \\ &\text{Depths:} \quad h_{\min} \le h_n \le h_0 \end{aligned} \quad \text{for } n = 1, 2, 3 \ldots N \tag{2}$$

where $h_{\min}$ is a feasible minimum thickness, defined a priori, motivated by the need for structural strength. The upper limit on the variables λ_n could also be set below $L/2$, if there is a desire to fix the height of the beam near its free ends. Then, the cross-sectional profile $H(x)$ is given by

$$H(x) = \begin{cases} h_n & \text{if} \quad \lambda_{n+1} < x \le \lambda_n \quad \text{for } n = 1, 2, 3 \ldots N \\ h_0 & \text{otherwise} \end{cases} \tag{3}$$

letting $\lambda_{N+1} = 0$.

2.2 Finite Element Model

Using the finite element method, we can then discretize the spatial domain into N_e beam elements (Timoshenko) of the same length, and formulate the system in the generic eigenvalue form

$$\left([\mathbf{K}] - \omega_m^2[\mathbf{M}]\right)\{\phi_m\} = 0 \tag{4}$$

where ω_n and $\phi_n(x)$ are the natural frequencies and mode shapes to be calculated, respectively.

Given that the profiles contain discontinuities, an approximation of the elemental profile height H_i (Figure 4) must be made for the elements in which discontinuities fall. Because, from (1), the modal frequencies of a homogeneous bar depend on the ratio $I(x)/A(x) \sim H^2(x)$, we choose to interpolate H_i using the following quadratic weighting:

$$H_i = \sqrt{\frac{h_{n-1}^2 \Delta x_1 + h_n^2 \Delta x_2}{(\Delta x_1 + \Delta x_2)}} \tag{5}$$

The cross-sectional area and second moment of inertia within such element i are readily obtained with $A_i = bH_i$ and $I_i = bH_i^3/12$, respectively. Some errors will inevitably occur using such interpolations, but ultimately, a sufficiently large number of elements N_e in the FE discretization will reduce them efficiently.

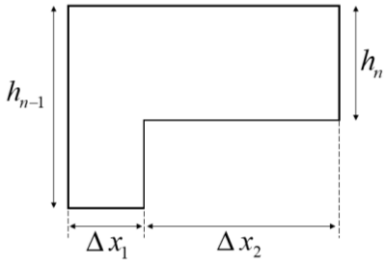

Figure 4 – Illustration of the finite beam element i, where a cross-sectional discontinuity occurs.

2.3 Objective Function

The objective function must include the mistuning between the several target frequencies ω_m^* and the frequencies estimated with the finite element model $\bar{\omega}_m(\boldsymbol{\lambda}, \mathbf{h})$. Here we choose to formulate the multiple tuning deviations in a single objective function describing the average squared error in percentage

$$\varepsilon(\boldsymbol{\lambda}, \mathbf{h}) = 100 \cdot \frac{1}{M} \sum_{m=1}^{M} \left(\frac{\bar{\omega}_m - \omega_m^*}{\omega_m^*} \right)^2 \tag{6}$$

where M is the number of defined targets, and m indicates the associated mode of vibration. Additionally, it can be useful to monitor the tuning deviations in *cents*, given by

$$\mathrm{E}_m(\boldsymbol{\lambda}, \mathbf{h}) = 1200 \cdot \log_2 \left(\frac{\bar{\omega}_m}{\omega_m^*} \right) \tag{7}$$

2.4 Geometric Penalties

In the spirit of designing a manufacturing aimed algorithm, we can add penalty terms, of various types, to the objective function to narrow the parameter space and generate more manufacture friendly geometries. Here we present two examples to: (1) minimize the amount of extracted material and (2) minimize abrupt changes of profile (smoothness of profile).

2.4.1 Volumetric Penalty

It can be generally said, particularly for CNC operated mills, that the time spent on carving a metal bar is proportional to the volume of material to be extracted. Hence, it would be resourceful to try and

minimize the amount of extracted material from the bar, in parallel to minimizing the deviations to the target frequencies. The percentage of volume of extracted material, with respect to the original bar's volume (with uniform profile), is given by

$$V(\boldsymbol{\lambda},\mathbf{h}) = 100 \cdot \frac{2}{h_0 L} \int_0^{L/2} \left(h_0 - H(x)\right) dx \tag{8}$$

Then, a combined objective function might be formulated as follows

$$\mathrm{E}(\boldsymbol{\lambda},\mathbf{h}) = (1-\alpha)\varepsilon + \alpha V \tag{9}$$

where $0 \le \alpha \le 1$ is a weighting factor on the volumetric penalty, i.e. if $\alpha = 0$ the volumetric constraint is not accounted for, if $\alpha = 1$ only the volumetric condition is accounted for.

2.4.2 Smoothness Penalty

Another example would be to penalize abrupt changes in the bar's height. This can be beneficial in: (1) increase model accuracy, as geometries with sharp discontinuities might be less reliable, possibly due to unaccounted 2-D effects [8] ; and (2) generating more aesthetically pleasing shapes with smoothly changing heights. Such a penalty can be formulized, for example, as a "profile roughness" to be minimized

$$S(\boldsymbol{\lambda},\mathbf{h}) = 100 \cdot \frac{1}{N} \sum_{n=1}^{N} \frac{\left|h_{n-1} - h_n\right|}{h_0} \tag{10}$$

which effectively represents an average height change per discontinuity, with respect to the original height h_0 . A combined objective function can be formulated as in (9).

3 Global Optimization: Evolutionary Algorithm

The general scope of an evolutionary algorithm relies on an attempt to mimic Darwinian evolution following the motto "*survival of the fittest*". We start the analogy by defining:

- Genes: the values of the variables in a solution $(\boldsymbol{\lambda},\mathbf{h})$;
- Individual: one solution, i.e. a defined set of variables $\boldsymbol{\lambda}$ and $\mathbf{h}$, whose fitness is characterised by the objective function;
- Population: A group of solutions.

The algorithm consists of a group of solutions that change at each iteration, where typically the best solutions "survive", "mate" and "mutate". The recipe of the algorithm developed here follows the typical loop of an evolutionary algorithm, consisting of 4 main steps:

- Elitism: a small group of the best solutions in one generation proceed unchanged to the next.
- Selection: consisting in the choice of individuals within the population that are going to "mate". Typically, the fittest solutions are more prone to be chosen.
- Crossover (mating): simulating natural reproduction, this step typically combines different genes of two parents (selected solutions) to generate children, whose genes are a crossover between the parents'.
- Mutation: simulating genetic mutation, this step consists in generating solutions with pseudo-random variations in their genes.

In sum, there are two qualitatively different aims in this process. One is that of "exploitation", i.e. finding local minima or improving an already obtained solution. The second one is that of "exploration" which aims at exploring different regions of the parameter space avoiding the algorithm to be stuck in local minima. The latter is carried mostly by the mutation step while the former is given by the selection/crossover steps.

3.1 Roulette Selection Operator

Here we choose to use the so-called roulette selection operator. It is a fitness proportional probabilistic method where the probability of an individual to be chosen to mate is given, in our case, by

$$p_i = \frac{1}{\varepsilon_i}\left(\sum_{j=1}^{N_{pop}} \frac{1}{\varepsilon_j}\right)^{-1} \tag{11}$$

where N_{pop} is the number of individuals in the population and ε_i is the evaluation of the objective function on individual i. The selected individuals are drawn from random numbers $r = [0, 1]$, as illustrated in Figure 5.

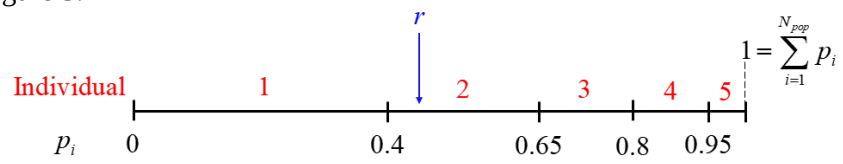

Figure 5 – Illustration of the selection procedure from the fitness proportional probabilistic method.

3.2 Heuristic Crossover Operator

Once selected the parents to mate, one needs an operator that combines the genes (variables) of two parents to form one or more children. Here we chose to use a simple, and commonly used, heuristic crossover operator that generates two children $\mathbf{c}_{1,2}$ from two parents $\mathbf{p}_{1,2}$ by the following

$$\begin{cases} \mathbf{c}_1 = \mathbf{p}_1 + r(\mathbf{p}_2 - \mathbf{p}_1) \\ \mathbf{c}_2 = \mathbf{p}_2 + r(\mathbf{p}_1 - \mathbf{p}_2) \end{cases} \tag{12}$$

where $\mathbf{c}_{1,2}$ and $\mathbf{p}_{1,2}$ are vectors containing the space variables of the solutions and r is a random number between $[0,1]$ with uniform distribution. Unlike many other crossover operators, this one does not retain any of the "genes" given by the parents but instead produces children whose "genes" are somewhere within the range defined by the parents. Moreover, the produced children are symmetrically located with respect to the "average gene values" defined by the parents.

3.3 Mutation Operator

The mutation operator in an evolutionary algorithm generally serves to create diversity in the population and explore new regions of the parameter space. The *Uniform Random Mutation* operator programmed here works in the following way:

- Step 1: Define how many "genes" are to be mutated N_{mut}, via a random pick of an integer between 1 and $2n$ (total number of degrees of freedom).
- Step 2: Select which "genes" will be mutated by randomly picking N_{mut} integers between 1 and $2n$.

- Step 3: Modify selected "genes" with pre-defined mutation strengths $\boldsymbol{\sigma}$, normalized to the variables' bounds, giving $\mathbf{c} = \mathbf{p} + r\boldsymbol{\sigma}$, where r is a vector of size $2n$ with N_{mut} non-zero elements, each composed of a random number between $[-1, 1]$, $\mathbf{c}$ is the mutated children and $\mathbf{p}$ is the parent to be mutated.

3.4 Overview of Procedure

Once defined the three main operators, the algorithm requires four parameters to be defined:

- Size of population N_{pop}, defining the number of individuals in each generation;
- Percentage of elitism P_e, defining the number of individuals that proceed to the next generation unchanged (fittest individuals);
- Ratio of crossover to mutation percentages $r_c = P_c / P_m$, where P_c and P_m are the crossover and mutation percentages, respectively and $P_e + P_c + P_m = 100\%$. This parameter defines if the algorithm will be geared more towards exploitation or exploration.
- Mutation strength σ, which controls the exploratory range of the mutation.

Additionally, a stopping criterion can also be set, typically either a maximum number of iterations and/or a satisfying error tolerance.

4 Results and Discussion

4.1 The Effect of the Volumetric Penalty

Here we present results from a study conducted to evaluate the effects of varying the volumetric penalty parameter α. Naturally, a larger α will emphasize more on reducing the amount of extracted volume at the expense of a less demanding tuning tolerance.
Surely, choosing an appropriate value for parameter α depends strongly on the problem at hand and on the amount of viable solutions. For example, very demanding tuning targets, e.g. with $M > 5$ or awkward tuning ratios, will reduce significantly the amount of viable geometries (reasonably in tune). The volumetric penalty is clearly useful in scenarios where a particular tuning target can be met by various geometries, i.e. in problems where there exist many global minima.
We conducted two statistical studies to evaluate the effect of varying the parameter α. For each value of α, a total of 50 runs of 100 iterations were computed to optimize an aluminum bar with dimensions: $L = 0.35$ m, $b = 0.05$ m and $h_0 = 0.01$ m to:

- Target frequency ratio (1:4:10) with maximum number of cuts $N = 5$;
- Target frequency ratio (1:2:4:8) with maximum number of cuts $N = 4$;

with fundamental frequency $f_1^* = 175$ Hz. The two cases aim to present qualitatively different examples: (1) a less demanding frequency ratio (1:4:10) with larger number of cuts (i.e. many viable solutions possible) and (2) a demanding frequency ratio (1:2:4:8) for a relatively low number of cuts, meaning lower number of viable solutions. For both cases an FE discretization with 150 elements was used. The size of the population was $N_{pop} = 50$, the crossover/mutation percentages were set to 30%/60%, with 10% of elitism. Uniform random mutation was used with mutation strength $\sigma = 0.2$. Each run solved approximately 4500 function evaluations and took approximately 1 minute. Figure 6 shows a statistical summary of the tuning error ε and extracted volume V for the optimal solution found to the (1:4:10) tuning ratio, for various values of parameter α. Analogous results for the (1:2:4:8) tuning ratio are found in Figure 7.
We note that, in both cases, increasing the value of α consistently generates solutions with less extracted volume. We see that, for low values of α, the extracted volume can be decreased without

compromising the tuning errors significantly. In fact, when $\alpha = 0.05$ for both cases, we note a slight improvement of the average tuning error of the solutions obtained, compared with the unpenalized results. This characteristic might be explained by the fact that, in the existence of multiple regions showing potential viable solutions, a lack of a volumetric penalty might divide the population, where several groups of individuals explore different regions, splitting the overall effort. In such scenarios, the volumetric penalty acts as a disambiguation between regions, creating more homogeneous populations, and hence, a faster convergence towards a global optimum.

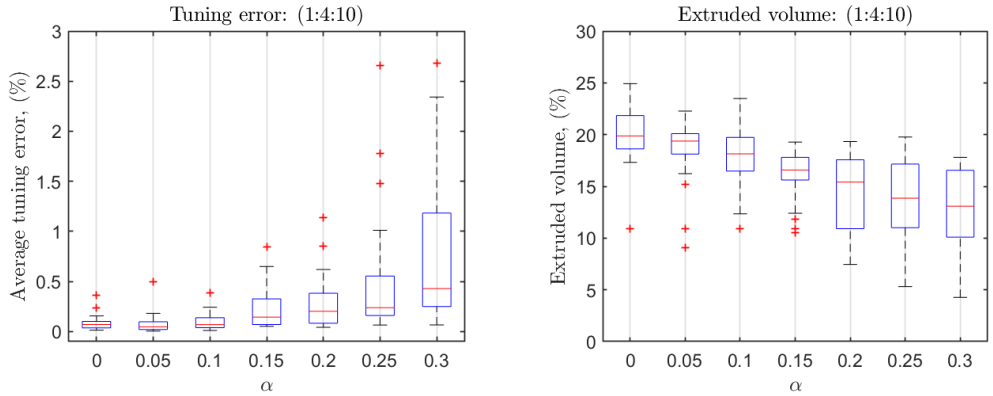

Figure 6 - Statistical summary of the tuning error ε and extruded volume V of the best solutions found to the ratio (1:4:10) at various values of α, after 100 iterations and a total of 50 runs per configuration. The red line is the median, blue box is the 25/75 percentiles, black whiskers the extreme values and red scatters are outliers.

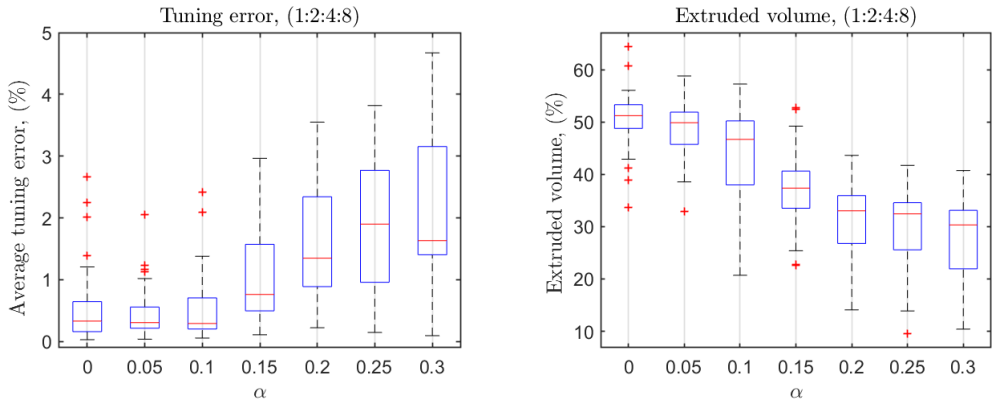

Figure 7 - Statistical summary of the tuning error ε and extruded volume V of the best solutions found to the ratio (1:2:4:8) at various values of α, after 100 iterations and a total of 50 runs per configuration. The red line is the median, blue box is the 25/75 percentiles, black whiskers the extreme values and red scatters are outliers.

In both cases, when $\alpha > 0.1$, the average tuning errors start to increase significantly. In these cases, it is up to the user what would be the best compromise between the reduction of extracted volume and tuning tolerances. If the evolutionary algorithm is used as a stand-alone process, lower values of α are probably a better option. However, if the combined approach is used, larger tuning errors stemming from the evolutionary algorithm with a heavy volumetric penalty can potentially be reduced by the derivative-based approaches a posteriori, setting $\alpha = 0$.

Figure 8 shows some of the optimized profiles found by the algorithm to the target ratios (1:4:10) and (1:2:4:8) with the respective percentage of extracted volume. All presented solutions have average tuning errors $\varepsilon < 0.01\%$. We note that in the pursuit of solutions with less extracted material, the

resulting geometries generally tend to be formed by deeper and more localized extractions. Curiously, these types of geometries are precisely those to be avoided by the "smoothness" penalty term S. This suggest that the two penalties presented may have somewhat of a contradictory nature.

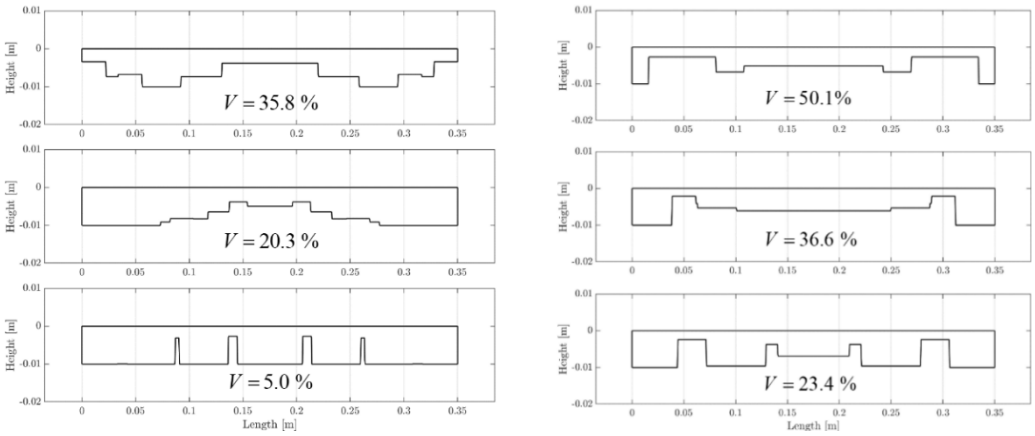

Figure 8 – Optimized undercuts for tuning ratios (1:4:10) on the left and (1:2:4:8) on the right, with various amounts of extracted volume.

4.2 The Effect of the Smoothness Penalty

Analogous studies were conducted for the smoothness penalty, S, shown in Figure 9 and 10. Similar results were found, confirming that it is possible to get, statistically, smoother geometries without loss of tuning accuracy, particularly when $\alpha < 0.2$. Again, above a certain value of α, the tuning accuracy starts to deteriorate, more evidently for the demanding tuning target (1:2:4:8). We notice as well that, for the less demanding target (1:4:10), there appears to exist a limit of improvement for the profile smoothness, beyond which the tuning error would severely deteriorate. This might occur because a certain amount of cut depth is necessary to bring the modal frequencies reasonably close to their targets.

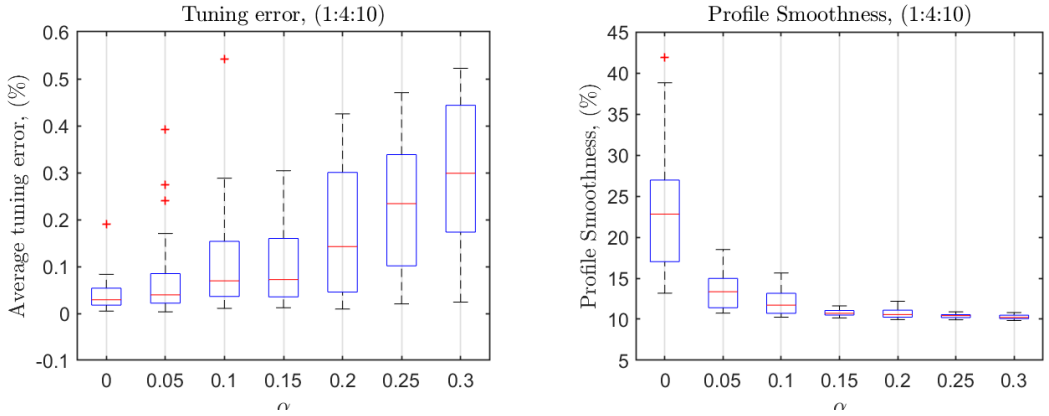

Figure 9 - Statistical summary of the tuning error ε and profile smoothness S of the best solutions found to the ratio (1:4:10) at various values of α, after 100 iterations and a total of 50 runs per configuration. The red line is the median, blue box is the 25/75 percentiles, black whiskers the extreme values and red scatters are outliers.

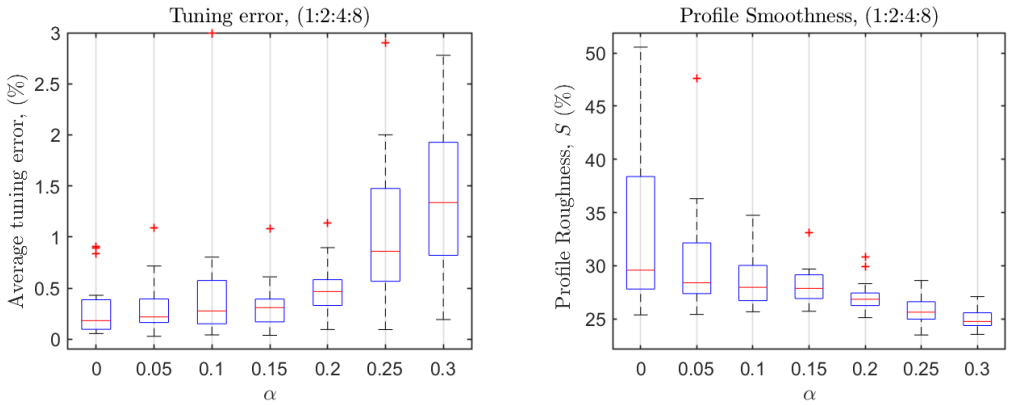

Figure 10 - Statistical summary of the tuning error ε and profile smoothness S of the best solutions found to the ratio (1:2:4:8) at various values of α, after 100 iterations and a total of 50 runs per configuration. The red line is the median, blue box is the 25/75 percentiles, black whiskers the extreme values and red scatters are outliers.

Figure 11 shows some of the optimized profiles found by the algorithm to the target ratios (1:4:10) and (1:2:4:8) with the respective values of "roughness". For the ratio (1:4:10), satisfying solutions were indeed obtained using the smoothness penalty compared to those typical found without it. For the ratio (1:2:4:8), while the penalty effectively reduced the parameter S, it seems that the demanding tuning ratio requires the geometry to have two abrupt cavities about 15% away from each end, as seen in Figure as well as in Figure 8.

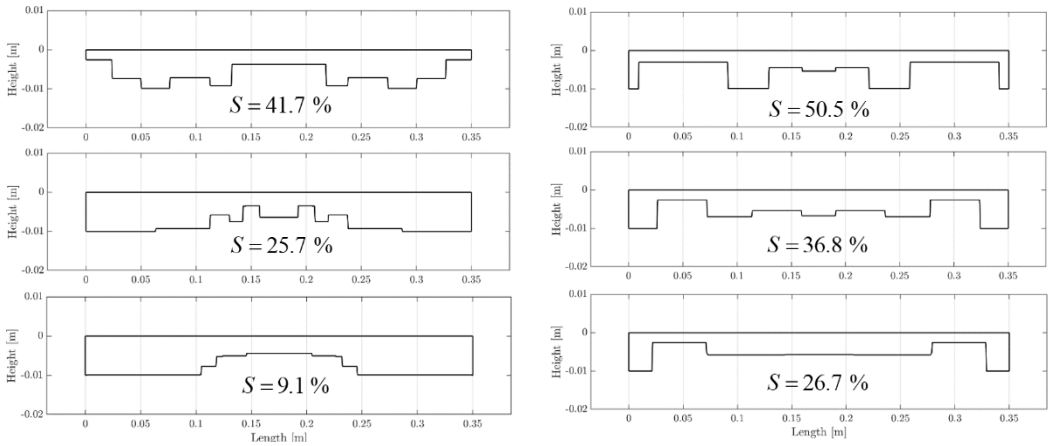

Figure 11 – Optimized undercuts for tuning ratios (1:4:10) on the left and (1:2:4:8) on the right, with various values of "smoothness".

4.3 Optimal Profiles for Various Tuning Ratios

Here we present the resulting optimized profiles for various unorthodox tuning ratios using the present model. For comparative reasons, we follow the target tuning ratios and model parameters found in [3]. Figure 12 shows the resulting optimized profiles for the tuning ratios (1:3:6:12), (1:5:10:15), (1:2:4:8:16), (1:2:5:10) and (1:3:5:7:9) found by [3] (on the left) and found by the present model (on the right). The solutions were calculated using 300 finite elements, used the least number of rectangular cuts and stopped at a criterion of error $\varepsilon < 0.01$ %.

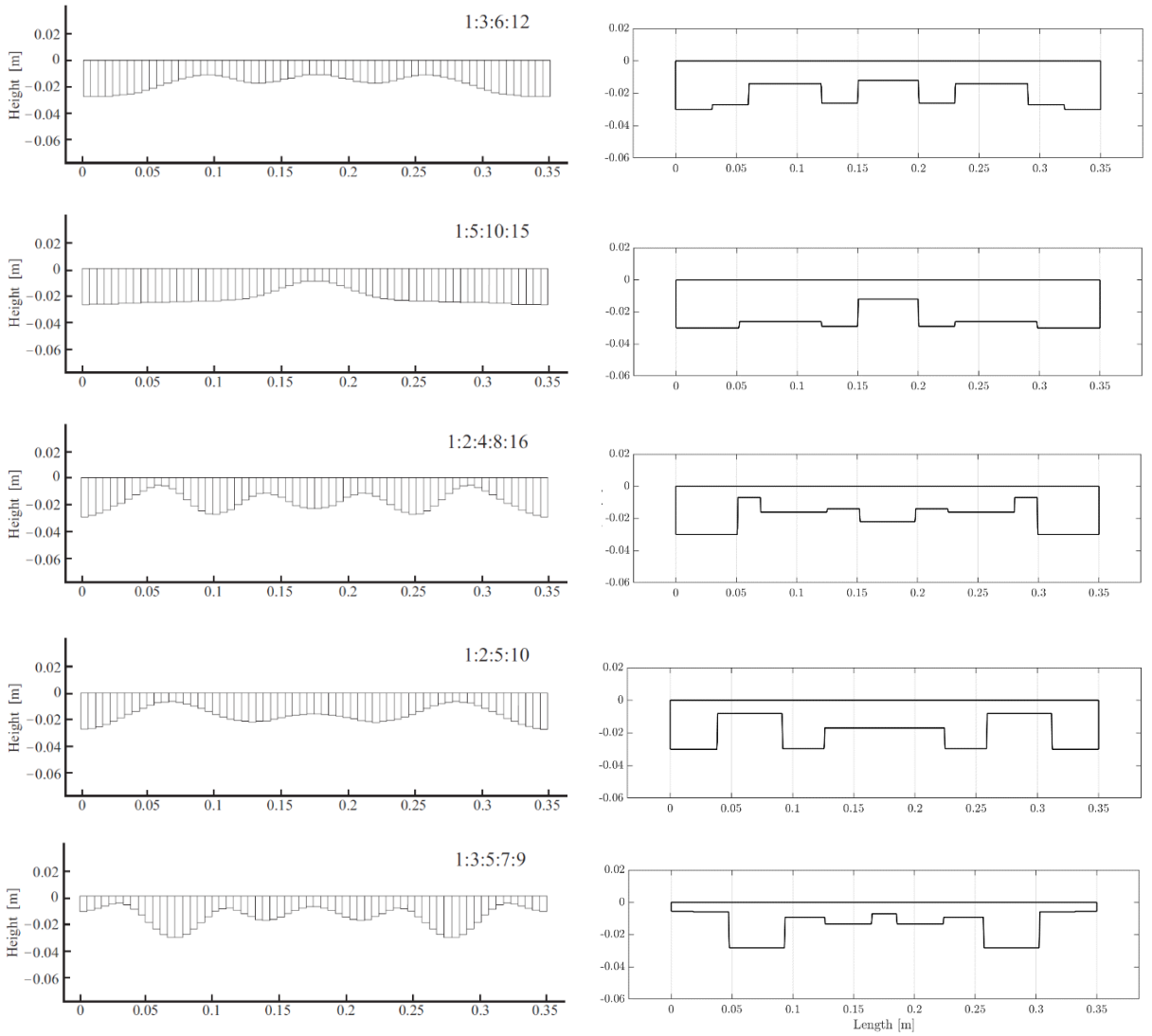

Figure 12 – Optimal shapes for various tuning ratios. Curvilinear profiles based on sine and cosine functions found in [3] (left) and those resulting from the present model (right). Note: relation between vertical and horizontal axis are not to scale.

Noticeably, most of the shapes resulting from the rectangular cut model resemble the shapes of the curvilinear undercuts found in [3]. All shapes appear to be discontinuous versions of their analogous, sinusoidal based, profiles.

These results demonstrate that is it possible, with the current model, to obtain designs with negligible tuning inaccuracies, even for unorthodox and demanding tuning targets. However, it must be noted that the modeling simplifications (1-D model, elemental interpolation for discontinuities, etc.) will surely incur small errors. These can in great part be resolved by applying the optimization procedure on a 3-D finite element model. This would provide a more realistic representation of real bars, as well as providing the possibility of including the frequency of torsional modes in the objective function, at the cost of larger computational times of course. However, since the number of variables is typically small $(n \leq 8)$, computation times should not become unfeasible. Nevertheless, the results show an exciting potential to design mallet instruments with new timbral characteristics and manufacture friendly designs.

5 Conclusion

In the pursuit of designing easily manufacturable bars tuned to specified target frequencies, we present a versatile 1-D model of the undercut based on a series of simple rectangular cuts, capable of achieving geometries for demanding tuning targets (large number of frequency targets and/or unorthodox tuning ratios), with a minimal amount of design variables (typically $n \leq 8$) . We show that, using evolutionary algorithms, with parameters tuned to accelerate convergence of the specified problem, it is possible to find simple optimal undercut geometries to satisfy a wide range of predefined tuning targets at low computational cost. In addition to tuning to target frequencies, we explored two added penalty terms, which successfully generated more appropriate geometries to either (a) minimize volume of extracted material or (b) smoothen abrupt changes in profile. It was found that these two possible additional criteria are, in a sense, opposite and lead to widely different profiles of the tuned bar.

Based on the evolutionary optimization algorithm developed, extensive computational experiments were carried out in order to assess the performance of the evolutionary algorithm as well as to find the most appropriate parameters values for solving this specific optimization problem.

Overall, the proposed bar-profiling concept seems promising, and the optimization strategy developed produced totally satisfying tuning results.

References

[1] L. Henrique, Acústica Musical, Lisboa: Fundação Calouste Gulbenkian, 2002.

[2] I. Bork, "Practical tuning of xylophone bars and resonators," *Applied Acoustics,* vol. 46, pp. 103-127, 1995.

[3] L. Henrique and J. Antunes, "Optimal design and physical modelling of mallet percussion instruments," *Acta Acustica,* vol. 89, pp. 948-963, 2003.

[4] Z. Mingming, Automatic multi-modal tuning of idiophone bars, Curtin: Curtin University, 2011.

[5] F. Bustamante, "Nonuniform beams with harmonically related overtones for use in percussion instruments," *Journal of the Acoustical Society of America,* vol. 90, no. 6, pp. 2935-2941, 1991.

[6] D. Beaton and G. Scavone, "Optimization of marimba bar geometry by 3D finite element analysis," in *Proceedings of ISMA 2019,* Detmold, Germany, 2019.

[7] D. Beaton and G. Scavone, "Measurement-based comparison of marimba bar modal behaviour," in *Proceedings of ISMA 2019,* Detmold, Germany, 2019.

[8] J. Petrolito and K. A. Legge, "Optimal undercuts for the tuning of percussive beams," *Journal of the Acoustical Society of America,* vol. 102, no. 4, pp. 2432-2436, 1997.

[9] E. Laukkanen and R. Worland, "Acoustical effect of progressive undercutting of percussive aluminium bars," in *Proceedings of Meetings on Acoustics,* Tacoma, 2011.

[10] K. F. Graff, Wave Motion in Elastic Solids, Dover Publications, 1991.