Politechnika Warszawska

WYDZIAŁ ELEKTRYCZNY

Instytut Elektrotechniki Teoretycznej i Systemów Informacyjno-Pomiarowych

Bachelor's diploma thesis

in the field of study Informatyka Stosowana

and specialisation Inżynieria Danych i Multimedia

Optimization of a two-element race car wing using a genetic algorithm and autoencoder

Rafał Szulejko

student record book number 271093

supervisor

Bartosz Chaber, PhD Sc.

WARSAW 2022

Optimization of a two-element race car wing using a genetic algorithm and autoencoder

Abstract

Following the invention of flaps and slats in aircraft, multi-element wings were brought to motorsport to increase effectiveness of front and rear wings. Usage of multi-element wings allows for much larger effective camber of the wing system as a whole compared to a single-element wing, reducing boundary layer separation and increasing overall efficiency. The interaction between used profiles is complex, and requires a delicate balance of airfoil shapes and their relative positions to achieve desired effect. The airfoil generation algorithm was developed using an autoencoder to output a complete airfoil shape using a single-digit number of input parameters. The autoencoder was trained in MATLAB using approximately 1600 airfoils from UIUC Airfoil Database. Then, a genetic algorithm was used to find the combination of parameters of two airfoils generated by this generator for highly negative lift coefficient and high lift to drag ratio. The fitness of each case was calculated using a detailed flow simulation with widely used, open-source OpenFOAM toolbox. During a 20-hour-long optimization process, approx. 1800 wings were analysed. The result is a two-element wing having a 10% larger downforce coefficient than the best member of random initial population at a cost of 6.55% increase of drag coefficient.

Keywords: genetic algorithm, machine learning, aerodynamics, CFD, autoencoders, motorsport, wings, downforce

Optymalizacja podwójnego skrzydła samochodu wyścigowego przy użyciu algorytmu genetycznego i autoenkodera

Streszczenie

Naśladując klapy i sloty stosowane w lotnictwie, wieloelementowe skrzydła zostały wprowadzone do samochodów wyścigowych celem zwiększenia wydajności przednich i tylnych skrzydeł. Wykorzystanie wielu płatów pozwala na większe praktyczne ugięcie w porównaniu z pojedynczym płatem, zmniejsza oderwanie warstwy przyściennej i zwiększa całkowitą wydajność układu. Przepływ spowodowany takim układem płatów jest skomplikowany i uzyskanie korzyści z jego zastosowania wymaga precyzyjnego dopasowania kształtów i wzajemnego położenia płatów. Zaprojektowany został algorytm do generowania kompletnych profili lotniczych przy użyciu kilku argumentów wejściowych wykorzystujący autoenkoder. Trenowanie autoenkodera przeprowadzono w środowisku MATLAB na bazie ok. 1600 profili lotniczych z bazy UIUC. Następnie, wykorzystano algorytm genetyczny w celu znalezienia optymalnej kombinacji parametrów dla dwóch profili, kierując się możliwie największym współczynnikiem docisku oraz największą wydajnością układu płatów. Oceny osobników dokonano na podstawie szczegółowych symulacji przepływu wykorzystując otwartoźródłowy i uznany pakiet OpenFOAM. W trakcie 20-godzinnego procesu optymalizacji przeanalizowano ok. 1800 osobników. Wynikiem optymalizacji jest dwuelementowe skrzydło charakteryzujące się współczynnikiem docisku o 10% większym od najlepszego losowo wygenerowanego kształtu kosztem zwiększenia współczynnika oporu o 6.55%.

Słowa kluczowe: algorytm genetyczny, sztuczna inteligencja, profil lotniczy, autoenkoder, uczenie maszynowe

Załącznik nr 2 do zarządzenia nr 109 /2021
Rektora PW z dnia 9 listopada 2021 r.

”załącznik nr 8 do zarządzenia nr 42 /2020 Rektora PW
annex no 8 to regulation no. 42 /2020 of the WUT Rector

Politechnika Warszawska

Politechnika Warszawska
Warsaw University of Technology

…….....
miejscowość i data
place and date

Rafał Szulejko
……..
imię i nazwisko studenta
name and surname of the student
271093
……..
numer albumu
student record book number
Informatyka Stosowana
……
kierunek studiów
field of study

OŚWIADCZENIE
DECLARATION

Świadomy/-a odpowiedzialności karnej za składanie fałszywych zeznań oświadczam, że niniejsza praca dyplomowa została napisana przeze mnie samodzielnie, pod opieką kierującego pracą dyplomową.
Under the penalty of perjury, I hereby certify that I wrote my diploma thesis on my own, under the guidance of the thesis supervisor.
Jednocześnie oświadczam, że:
I also declare that:

- niniejsza praca dyplomowa nie narusza praw autorskich w rozumieniu ustawy z dnia 4 lutego 1994 roku o prawie autorskim i prawach pokrewnych (Dz.U. z 2021 r., poz. 1062) oraz dóbr osobistych chronionych prawem cywilnym,
  this diploma thesis does not constitute infringement of copyright following the act of 4 February 1994 on copyright and related rights (Journal of Acts of 2021, item 1062) or personal rights protected under the civil law
- niniejsza praca dyplomowa nie zawiera danych i informacji, które uzyskałem/-am w sposób niedozwolony,
  the diploma thesis does not contain data or information acquired in an illegal way,
- niniejsza praca dyplomowa nie była wcześniej podstawą żadnej innej urzędowej procedury związanej z nadawaniem dyplomów lub tytułów zawodowych,
  the diploma thesis has never been the basis of any other official proceedings leading to the award of diplomas or professional degrees,
- wszystkie informacje umieszczone w niniejszej pracy, uzyskane ze źródeł pisanych i elektronicznych, zostały udokumentowane w wykazie literatury odpowiednimi odnośnikami,
  all information included in the diploma thesis, derived from printed and electronic sources, has been documented with relevant references in the literature section,
- znam regulacje prawne Politechniki Warszawskiej w sprawie zarządzania prawami autorskimi i prawami pokrewnymi, prawami własności przemysłowej oraz zasadami komercjalizacji.
  I am aware of the regulations at Warsaw University of Technology on management of copyright and related rights, industrial property rights and commercialisation.

……
czytelny podpis studenta
legible signature of the student”.

Politechnika Warszawska

Załącznik nr 4 do zarządzenia nr 109 /2021
Rektora PW z dnia 9 listopada 2021 r.

„załącznik nr 10 do zarządzenia nr 42/2020 Rektora PW
annex no. 10 to regulation no. 42/2020 of the WUT Rector

Politechnika Warszawska
Warsaw University of Technology

…….....
miejscowość i data
place and date

Rafał Szulejko
imię i nazwisko studenta
name and surname of the student
271093
numer albumu
student record book number
Informatyka Stosowana
Wydział i kierunek studiów
faculty and field of study

Oświadczenie studenta w przedmiocie udzielenia licencji
Politechnice Warszawskiej
Student declaration on granting a license to the Warsaw University of Technology

Oświadczam, że jako autor/współautor* pracy dyplomowej pt. …..... udzielam/nie udzielam* Politechnice Warszawskiej nieodpłatnej licencji na niewyłączne, nieograniczone w czasie, umieszczenie pracy dyplomowej w elektronicznych bazach danych oraz udostępnianie pracy dyplomowej w zamkniętym systemie bibliotecznym Politechniki Warszawskiej osobom zainteresowanym.
I hereby declare that as the author/co-author of the diploma thesis entitled "……." I grant/do not grant* the Warsaw University of Technology a free, non-exclusive, unlimited in time license to include my diploma thesis in electronic databases and to make my diploma thesis available in the closed library system of the Warsaw University of Technology to those interested.*

Licencja na udostępnienie pracy dyplomowej nie obejmuje wyrażenia zgody na wykorzystywanie pracy dyplomowej na żadnym innym polu eksploatacji, w szczególności kopiowania pracy dyplomowej w całości lub w części, utrwalania w innej formie czy zwielokrotniania.
The license to make the diploma thesis available does not include the consent to use the diploma thesis in any way, especially to copy the diploma thesis in full or in part, to save it in any form or to multiply it.

.....
czytelny podpis studenta
legible signature of the student

* niepotrzebne skreślić
delete as applicable ".

Politechnika Warszawska
Warsaw University of Technology

załącznik nr 10 do zarządzenia
nr /2016 Rektora PW

Oświadczam, że treść pracy dyplomowej w wersji drukowanej, treść pracy dyplomowej zawartej na nośniku elektronicznym (płycie kompaktowej) oraz treść pracy dyplomowej w module APD systemu USOS są identyczne.

I certify that the content of the printed version of the diploma thesis, the content of the electronic version of the diploma thesis (on a CD) and the content of the diploma thesis in the Archive of Diploma Theses (APD module) of the USOS system are identical.

.....
czytelny podpis studenta
legible signature of the student

Contents

Chapter 1

Introduction

Following the invention of flaps and slats in aircraft, multi-element wings were brought to motorsport to increase effectiveness of front and rear wings. Usage of multi-element wings allows for much larger effective camber of the wing system as a whole compared to a single-element wing, reducing separation and increasing overall efficiency. The interaction between used profiles is complex, and requires a delicate balance of airfoil shapes and their relative positions to achieve desired effect. Research into physical behavior of these systems is conducted in aerospace industry since early 20th century and branched into separate motorsport use in 1970s [20]. With advent of computer-assisted design, several optimization techniques were devised, both in the airfoil shape generation as well as in broader field of function optimization algorithms. This work is inspired by unpublished research by Porsche AG into optimization of a two-element wing for Porsche 919 Evo car mentioned in press material [6].

1.1 Scope of the thesis

The following project involves design and implementation of a generator of airfoil shapes based on a encoder-decoder network trained on a widely used UIUC Airfoil Data Site airfoils coordinate database [19]. This algorithm generates entire airfoil shape from an arbitrarily set number of input parameters. Then, this proposed method was used in a numerical study to generate an efficient two-element wing through a genetic algorithm (GA) optimization. In the study, GA started with a random population of wings. Then, for each individual of the next generation, GA selected parameters for two airfoils generated with the new generator. The fitness function was calculated by scripting generation of mesh and solving a flow around the generated airfoils using OpenFOAM. This process was repeated by GA until a selected stop conditions was reached.

1.2 Airfoil optimisation

1.2.1 Airfoil shape parametrization

In order to optimize shape of the airfoil (or system of those), its shape needs to be parameterized. The first usable parametrization of airfoil shape was published in late 1920s and in 1930s by NACA in the form of their 4- and 5-digit airfoils [2]. These simple parabola- and hyperbola-based formulas were useful for some time, but provided a limited space of airfoil shapes. The other notable method was use of parametric curves, namely Bezier curves and B-Splines. This allowed representation of much more shapes at a cost of significant increase of number of parameters required, as well as lack of visible connection between parameter values and airfoil shape. A solution to this problem was proposed in 1990 in a form of PARSEC formula [21], generating smooth airfoil shape based on 11 design parameters, some of which are coordinates of highest and lowest points on the top and bottom surfaces as well as the position of the trailing edge. Since then, majority of papers used various parametric curves for airfoil optimization [22], [7]. Machine learning-based methods [4], [23], [26] are emerging, but it is difficult to define a trend as of now.

1.2.2 Genetic algorithms

Evolutionary, and especially genetic algorithms have been used in optimization of various nonlinear problems since their creation in 1960s and 1970s in works pioneered by John Holland [12]. Genetic algorithms work by applying a process of natural-selection-inspired actions (selection, crossover and mutation) on sets of parameter combinations (populations). As these actions are fundamentally driven by randomness and not any additional function knowledge (like derivative), they can be used to optimise virtually any function. However though, this advantage is balanced by the need to perform many more calculations of searched functions, slower convergence and need of correct choice of delicate configuration parameters (otherwise the algorithm might not converge at all or produce bad results).

1.2.3 Fluid flow solution

Historically, the first widely used solution was the potential flow panel code developed first in 1960s and 1970s [8], which discretizes only geometry of the object using potential sinks, sources and doublets allowing to quickly and reliably find flow velocities and pressures. While able to provide usable results for simple flows and geometries in time at least an order of magnitude lower than CFD, the rapid growth of available computing power and research on the latter, as well as ability to solve arbitrarily complex flow, CFD has become the leading

method of fluid flow simulation in aviation and the only method in fields requiring more complex simulations, like motorsport. Another approach is to use surrogate models, which approximate computationally intensive parts of the solution or replace the entire solution altogether. This is a rapidly growing field, with machine learning methods [3], [14] being often used to predict values of coefficients [5], predict entire flow solution or to increase fidelity of a simpler CFD solution.

1.3 Multi-element wings

Having finite usable wing chord length, race car designers have to increase camber in order to achieve higher downforce. However, excessive camber leads to adverse pressure gradient on the wing surface, leading to boundary layer separation and both increase in drag and loss of downforce, contrary to desired results [16]. In order to effectively increase camber into higher values an aviation solution was brought in – slats and flaps [13], [17]. While multi-element wings in motorsport have similar appearance than their aviation counterparts, the differences are significant, as motorsport multi-element wings rely on smooth flow and efficiency, not having to reduce drag for majority of the time to the same degree as airplane wings. Motorsport wings are essentially static "external airfoil" or "Junkers" flaps, used in Junkers aircraft until 1940s, and in few STOL (short takeoff and landing) airplanes afterwards. There has been research on multi-element wings with Junkers flap [18], or partially or strictly in motorsport sense [20], [27], but practical guidelines in textbooks [15] are still ambiguous and focus on experimental testing.

1.4 Optimisation of multi-element wings

Optimizing multi-element wing setup presents new problems not present in optimisation of a single airfoil, namely relative configuration between elements. This is an important distinction – as mentioned in the previous chapter the relative position and gap could noticeably influence the results. This leads to even faster growth of number of optimized variables: $2\mathrm{n}+(\mathrm{n}-1)\mathrm{m}$, where n is a number of elements and m is a number of variables needed to parametrize the relative positions and dimensions of neighboring elements. With the search space increasing exponentially with number of parameters, optimization of multi-element wing becomes a very computationally expensive problem. This does not include increased complexity of CFD solution caused by both more complex flow needing more iterations to converge as well as denser mesh necessary for such a flow to produce valid results. Knowing that the speed of

CFD solution cannot be easily and cheaply increased, it is necessary to decrease the number of optimized variables.

Chapter 2

Proposed parametrization method

Author's initial attempts at recovering entire airfoil data by the autoencoder resulted in shapes that resembled airfoils, but lacked smoothness and had problems with closure of trailing edge. This was due to the introduction of noise inherent to autoencoder behavior. To counter that, and take advantage of the airfoil format standardizing location of leading and trailing edges, the autoencoder was trained on a heavily reduced number of preselected nodes with the intent to later interpolate new nodes generated by a decoder for arbitrary combination of hidden layer parameters. This is a simpler solution than [23] which utilized complex autoencoder-GAN system to counter the same problems.

2.1 Generator

2.1.1 Autencoders

Autoencoders (Fig. 1) are artificial neural networks used for encoding input into a lower dimensional latent space representation (labeled "Code"), and later reconstruct the original from it. This unsupervised learning method has been initially introduced as a dimensionality reduction tool in the late 1980s [10] and were later used at denoising and generative modeling. In this case, ability to dimensionally reduce nonlinear data was used.

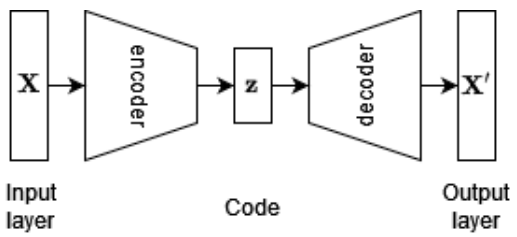


Figure 1. General autoencoder model

2.1.2 Data preparation and autoencoder training

UIUC Airfoil Database [19] contains ∼1600 airfoils in a form of X-Y coordinates. This format doesn't specify number of points, which requires additional standardization before any NN training. Instead of interpolating of the entire airfoil shape (to e.g. 200 points) and training on that, $2\mathrm{k}$ nodes were placed on each airfoil, k per side. These nodes, in combination with points $(1, 0)$ (twice) and $(0, 0)$ should be sufficient to represent sufficient space of airfoil shapes. Start of the training begins with selection of horizontal positions of nodes $\mathrm{x_i}$, where $\mathrm{i} = 1, 2 \ldots \mathrm{k}$ (Fig. 2). This distribution is suggested to be nonlinear, to both better capture the curvature near the leading edge, and also to help interpolation of new nodes generated by the decoder. Note that $\mathrm{x_i} \notin \{0, 1\}$ – since every airfoil contains these points, nodes in these positions would carry no information. The next step is to measure vertical position of both

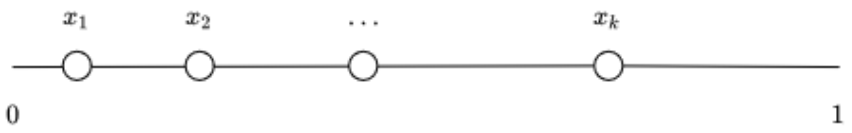
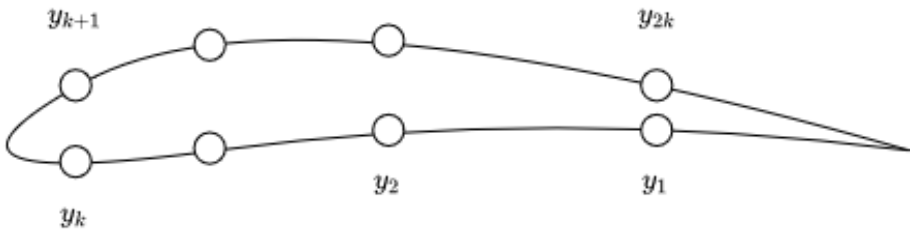
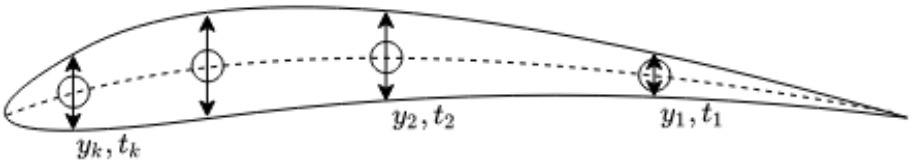


Figure 2. Horizontal positions $\mathrm{x_i}$ of nodes

airfoil surfaces at every $\mathrm{x_i}$ (Fig. 3a). An alternative method to surface nodes can be to store vertical position of camber line and airfoil thickness at $\mathrm{x_i}$ (Fig. 3b). Both ways have equal length and contain entire necessary information, but the latter only has positive values.



(a) Measurement of top and bottom surface vertical positions $\mathrm{y_j}$ of nodes at $\mathrm{x_i}$



(b) Measurement of camber line vertical positions $\mathrm{y_i}$ of nodes and airfoil thicknesses $\mathrm{t_i}$ at $\mathrm{x_i}$

Figure 3. Alternative methods of data measurement for generator training

Next, these values are stored in arbitrary order (e.g. in the same order as in the input database)

in an array Y of width equal to $2\mathrm{k}$ and height of the training set. The two alternative ways of storing data from Fig. 3 are presented below.

$$\mathrm{Y} = \begin{bmatrix} \mathrm{y}_{1,1} & \mathrm{y}_{1,2} & \cdots & \mathrm{y}_{1,2\mathrm{k}} \\ \mathrm{y}_{2,1} & \mathrm{y}_{2,2} & \cdots & \mathrm{y}_{2,2\mathrm{k}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{y}_{\mathrm{n},1} & \mathrm{y}_{\mathrm{n},2} & \cdots & \mathrm{y}_{\mathrm{n},2\mathrm{k}} \end{bmatrix}$$

$$\mathrm{Y} = \begin{bmatrix} \mathrm{y}_{1,1} & \mathrm{y}_{1,2} & \cdots & \mathrm{y}_{1,\mathrm{k}} & \mathrm{t}_{1,1} & \mathrm{t}_{1,2} & \cdots & \mathrm{t}_{1,\mathrm{k}} \\ \mathrm{y}_{2,1} & \mathrm{y}_{2,2} & \cdots & \mathrm{y}_{2,\mathrm{k}} & \mathrm{t}_{2,1} & \mathrm{t}_{2,2} & \cdots & \mathrm{t}_{2,\mathrm{k}} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathrm{y}_{\mathrm{n},1} & \mathrm{y}_{\mathrm{n},2} & \cdots & \mathrm{y}_{\mathrm{n},\mathrm{k}} & \mathrm{t}_{\mathrm{n},1} & \mathrm{t}_{\mathrm{n},2} & \cdots & \mathrm{t}_{\mathrm{n},\mathrm{k}} \end{bmatrix}$$

The autoencoder can now be trained, where hidden layer size l is the number of parameters of the (Fig 4).

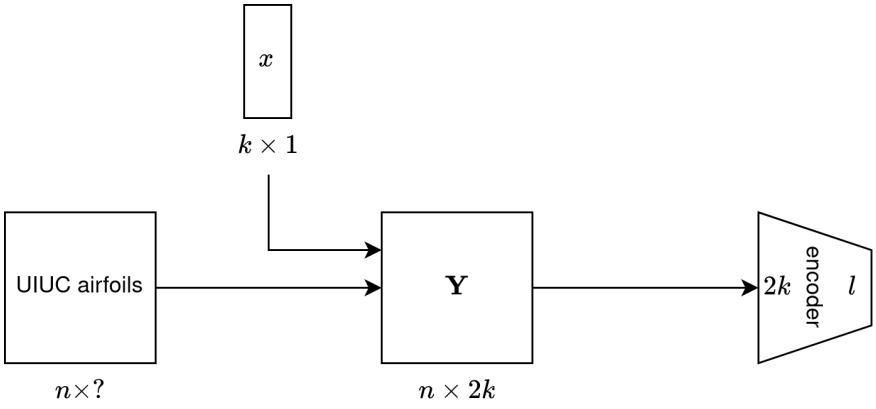


Figure 4. Training process. As airfoils in the UIUC database vary in number of data points, the width of this data on the diagram was described as unknown.

2.1.3 Airfoil generation

To generate an airfoil it is necessary to input l parameters into decoding function of trained autoencoder, and arrange $2\mathrm{k}$ of output parameters back into k nodes on top and bottom surfaces of the airfoil knowing their horizontal positions $\mathrm{x_i}$ selected at the beginning of the training process (if method 3b was used, nodes have to be regenerated from camber line and thickness). Airfoil shape can now be obtained by interpolating through (1,0), nodes on the bottom surface, (0,0), nodes on the top surface and (1,0) again (or in the opposite direction). Entire generation process is presented on Fig. 5.

2.2 Comments

The method seems to be able to successfully reproduce and generate variety of airfoil shapes with lower number of parameters than non-NN-based methods. However, there exist several drawbacks - the first being nondeterministic traning process leading to slightly different autoencoder coefficient at every training. This forces to save the trained autoencoder if one wants to store airfoil shapes in parametrized form. The other is that the parameter values are not bound in any way except being in $(0, 1)$ range. In many cases the autoencoder produces a valid airfoil shape, but in a significant number of parameter combinations the resulting shape is either useless or even self-intersecting (Fig. 6). As shown in the following chapter, this forces certain approach in its usage in evolutionary algorithms.

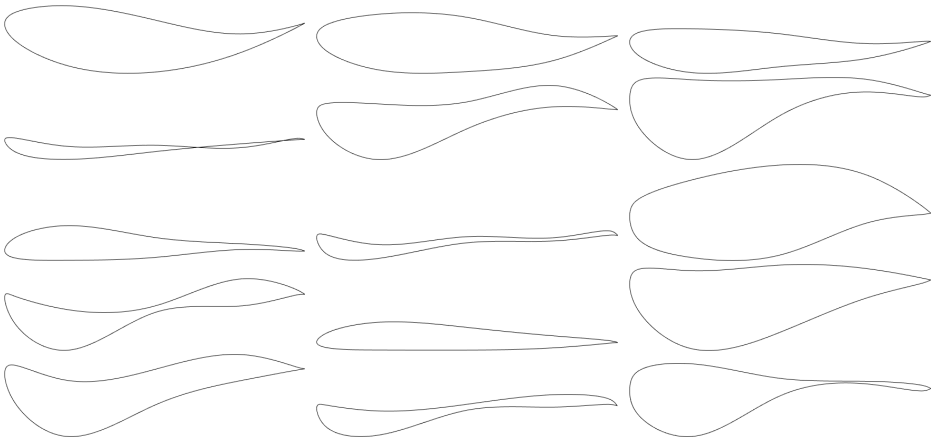


Figure 6. Examples of generator output for different manually created and random parameter combinations

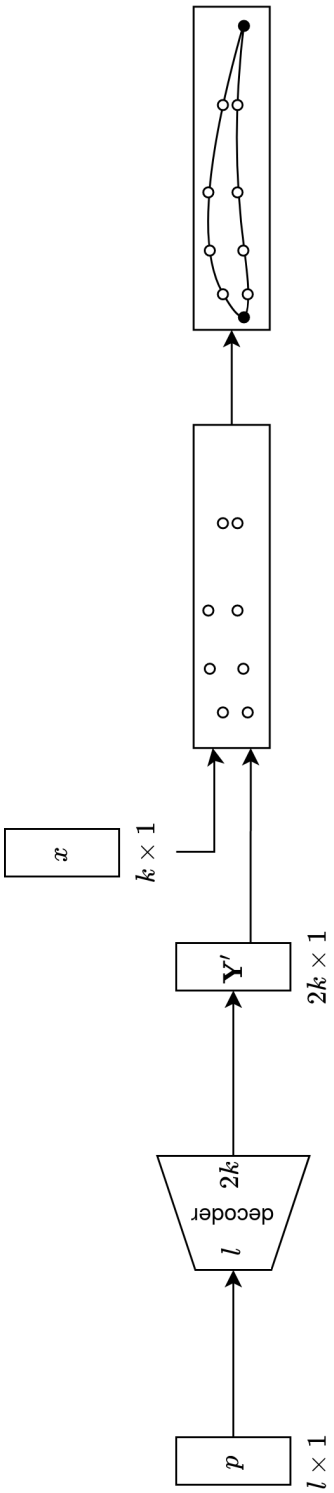


Figure 5. Airfoil generation

Chapter 3

Numerical study

The new parametrization method will now be applied to the practical problem of optimization of a two-element 2D wing.

3.1 Geometry processing for flow simulation

3.1.1 Airfoil generator

The airfoil shape generator was implemented in MATLAB using MATLAB's autoencoder implementation from Deep Learning Toolbox. The autoencoder was trained with data as described in subsection 2.1.2, particularly using the second presented method (Fig. 3b). After the training, the decoder part has been tested for various parameter combinations. However, due to time constraints, no thorough testing has been done. The default configuration parameters were used except of extending the training epoch limit.

Parameter	Value
Input layer length	10
Code length	5
Encoder/decoder transfer function	logsig
Epoch limit	10000
L2 weight regularizer coefficient	0.001
Loss function	msesparse
Sparsity proportion	0.05
Sparsity regularization coefficient	1
Training algorithm	trainscg

Table 1. Final autoencoder parameters

3.1.2 Multi-element wing configuration

An usual configuration of airfoils multi-element wings is via overlap and gap measured along X and Y axes respectively and arbitrary set angle and chord ratio. While possible to manually adjust these four values for each case with good results, four additional parameters in evolutionary algorithm search would significantly increase total time required to find an acceptable solution. Constant angle and relative position could also influence the direction of search into airfoils with trailing edge more suitable for them. As [27] notes, for the same pair of airfoils, variation of overlap and gap of 2-4% and 2.6-3.7% of main airfoil chord respectively could change the resulting $\mathrm{C_L}$ by over 5%. In order to minimize the influence of these factors, a different method of relative positioning was used (Fig. 7).

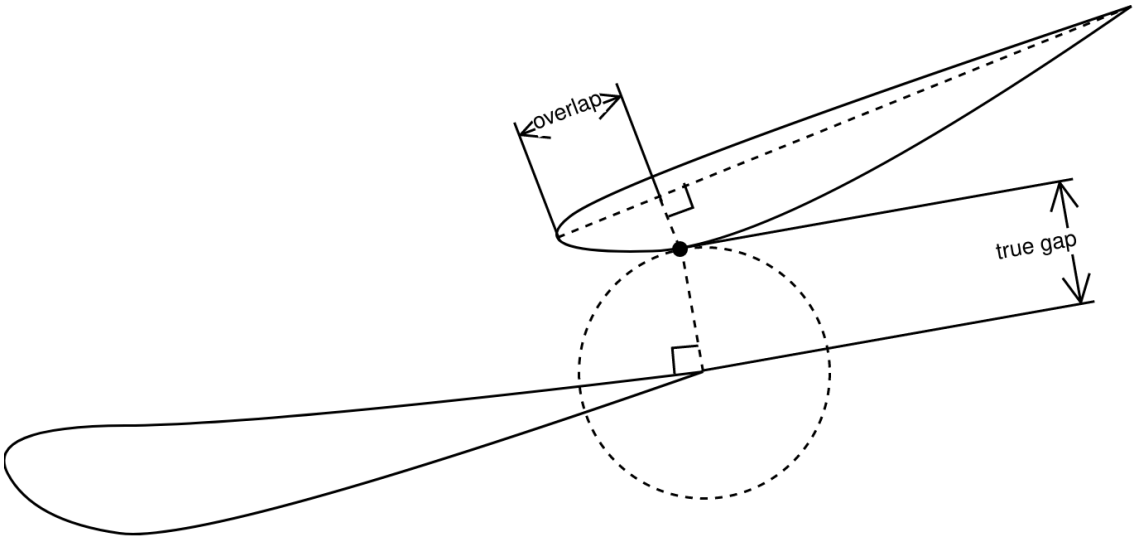


Figure 7. Relative positioning method used

This way the true gap is constant and angle of the flap is such that the bottom surface of the flap is tangent to the top surface of the main airfoil at the ends of the true gap. Even though this method does not create perfect relative configuration, it attempts to standardize interaction between various pairs of airfoils. Further research of physical interaction between close pairs of smooth airfoils could provide data to create more optimal general solution. For further reduction of number of parameters, the angle of attack of main airfoil was set to zero. Figure 8 presents behavior of this method in practice. Finally, the gap and overlap were set to 2% and 5% of the main element chord respectively.

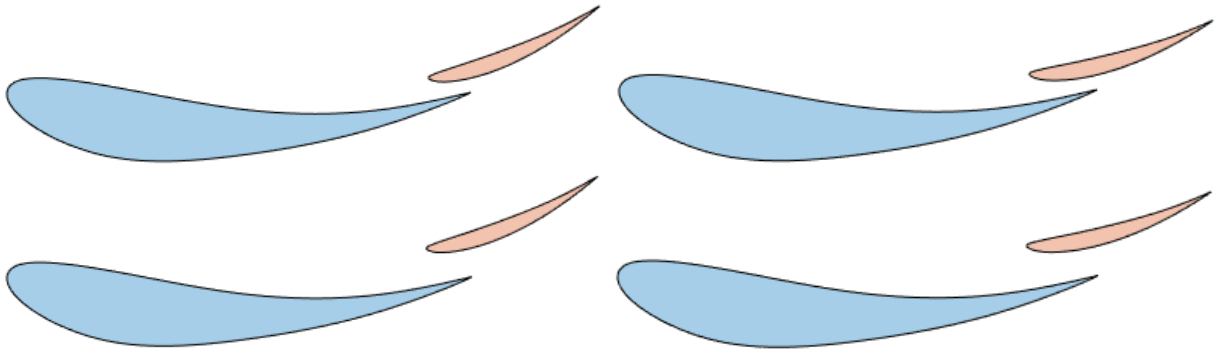


Figure 8. Examples of used relative positioning method. Different combinations of two gaps and two overlaps shown.

3.2 Genetic algorithm configuration

An implementation of MATLAB's genetic algorithm from Global Optimization Toolbox was used. Due to the nature of selected airfoil parametrization algorithm, many parameter combinations result in invalid airfoils. While unable to fully prevent this from happening, a few steps were taken to minimize the influence of the generated invalid airfoils on the optimization process. First way of dealing with the problem was algorithmically computed relative configuration of two airfoils (instead of controlled by optimized variables). The second way: before starting the optimizer, a random generator is used to fully populate the initial sample only with valid parameter combinations. Thirdly, if an invalid specimen is generated through a crossover or mutation, its fitness function is automatically set to zero to minimize the probability of this specimen further reproduction. Finally, all invalid specimens were replaced with new, randomly generated valid specimens every fifth generation and/or in case of over half of population being invalid. As preliminary test runs showed, optimising for the lift-to-drag ratio often promotes wings with extremely low drag. In consequence, a different fitness function was selected to promote both lift coefficient, as well as airfoil efficiency, equal to lift coefficient multiplied by the lift-to-drag ratio. All parameters are presented in Table 2.

3.3 Meshing

In numerical studies performed in literature on motorsport-oriented multi-element wings, meshes were often generated for $30 < \mathrm{y}^+ < 100$ [11]. While this is easily explainable through the "log law" of the wall functions, as noted in [25], usage of mesh with $\mathrm{y}^+ < 5$ can actually produce a different flow, leading to results differing by up to 9%. While this study attempts to solve meshes with inflation layers resolving flow in the viscous sublayer, the

Parameter	Value
Fitness function	$-\frac{C_L^2}{C_D}$
Selection function	selectionstochunif
Crossover function	crossoverscattered
Crossover fraction	0.8
Mutation function	mutationadaptfeasible
Mutation rate	0.01
Population size	10

Table 2. Final genetic algorithm parameters

intent was to still balance validity of the results with the solution speed necessary for usage in evolutionary algorithms. Gmsh [9] meshing software was used thanks to its ability to fully script the geometry and meshing configuration.

3.3.1 Domain parametrization

Regardless of its origins in the structured meshes, an unstructured C-shaped mesh was used in order to reduce the number of cells compared to a rectangular domain. The shape and size of the domain was designed so that it could be completely described with a single variable (Fig. 9). The cells were sized according to several simple functions. In the region nearest

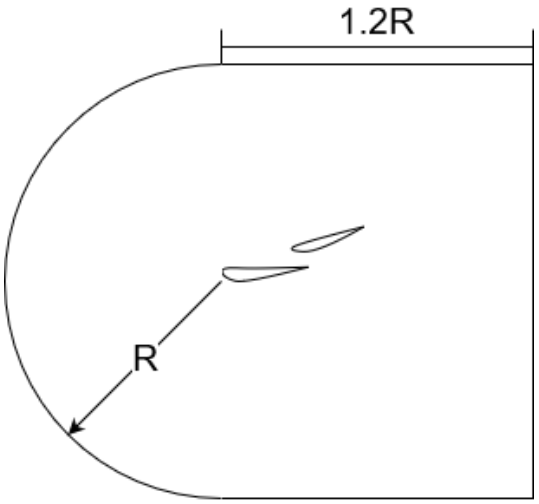
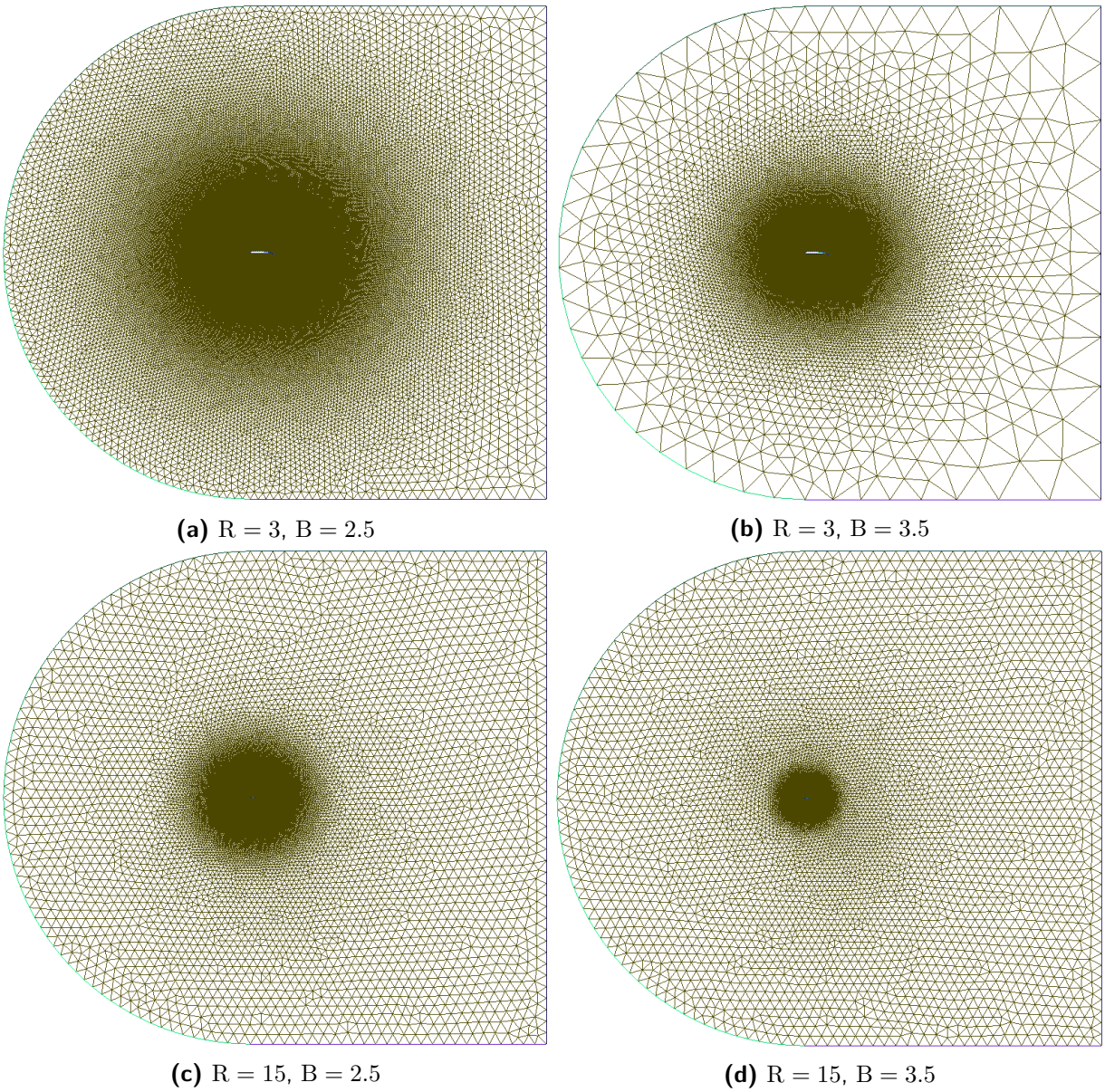


Figure 9. Shape and sizing of the computational domain

to the wall, quad boundary layer inflation cells were used with height specified by geometric sequence $h_{wall} \cdot G^{n-1}$, capped by total maximum height h_{total}, where h_{wall} is thickness of the first layer, G is boundary layer cell growth ratio and n is number of boundary layer cell. Boundary cell widths were sized by Gmsh using h_{far} variable. Next, the triangular cells were

scaled with a monomial $\mathrm{h_{far}} \cdot (\mathrm{dist(AF)} + 1 - \mathrm{h_{total}})^{\mathrm{B}}$ – a function of distance from the airfoils $\mathrm{dist(AF)}$ and exponent B, capped with an arbitrary chosen maximum cell size. This, when other parameters are all sensibly chosen constants, allows to completely define the shape and sizing of the mesh with only domain radius R and cell size function exponent B, as shown on the Fig. 10.



(a) $\mathrm{R} = 3$, $\mathrm{B} = 2.5$ **(b)** $\mathrm{R} = 3$, $\mathrm{B} = 3.5$

(c) $\mathrm{R} = 15$, $\mathrm{B} = 2.5$ **(d)** $\mathrm{R} = 15$, $\mathrm{B} = 3.5$

Figure 10. Examples of meshes generated with two-variable parametrization (not to scale). In both bottom cases, maximum cell size of 1 was reached.

3.3.2 Domain parameters selection and validation

The two variable domain parametrization as described in previous subsection was used to perform a parameter sweep over these two to analyse sensitivity of the results on the mesh size and cell growth rate, and in turn use the coarsest mesh that would provide satisfactory results. While analysis of sensitivity depending on mesh density is a standard procedure in such cases, domain far field effects can also have a significant influence on results as shown in [1]. As a reference, experimental results from NACA TR 573 report [18] were used, since they are already cited with the same purpose in papers, such as [11]. The geometry (10" NACA 23012 main element and 2" NACA 23012 flap angled at 20°, Fig. 11) was reproduced and placed in the several domains with R ranging from 3 to 20 m ($\sim$10 and $\sim$66 12" reference chord lengths respectively, top value should have been higher as suggested in [1], but number of cells would be too large for feasible computation time of hundreds of similar runs on the hardware used for the study) and cell size function exponents B ranging from 2 to 4. The maximum cell size was set to 1 m which is large, but for the selected cell sizing method and range of exponents used, this maximum cell size is never close to the analysed wing.

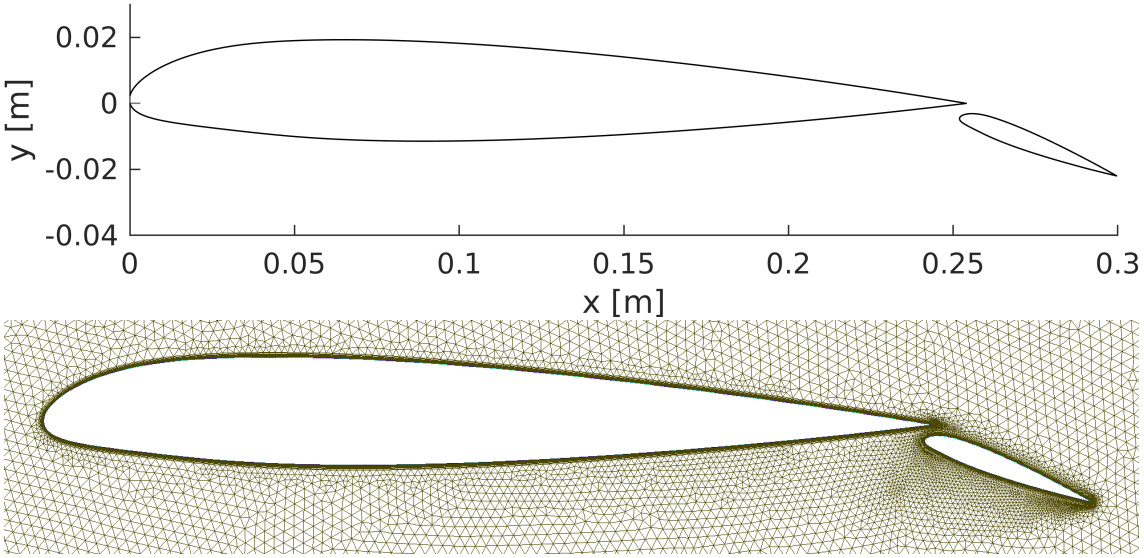
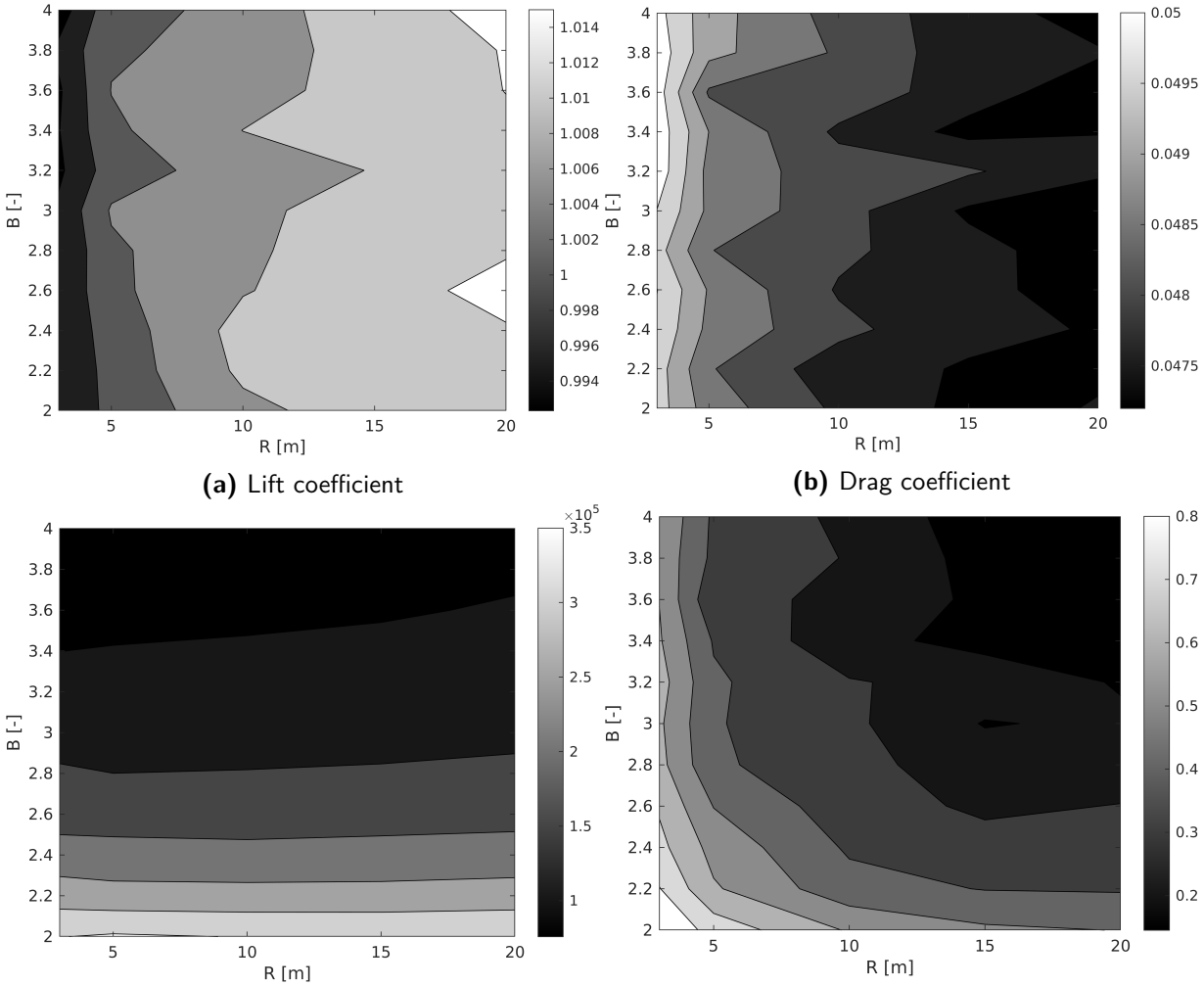


Figure 11. NACA TR 573 [18] validation geometry and mesh in close vicinity to the airfoils.

Unfortunately, as presented on fig. 12, the actual results were never actually close to experimental or those obtained in [11] (i.e. $\mathrm{C_L} \approx 1$ instead of 0.74, further commentary in subsection 3.3.3). Apart from an apparent setup error, the simulations in the study are consistent with each other and probably predict the flow well enough. The other observation from Figures 12a, 12b and 12c is that due to selected cell sizing method, the domain radius R has the major influence on results and cell size function exponent B has the major influence on



(a) Lift coefficient

(b) Drag coefficient

(c) Number of cells

(d) Weighted sum of normalized 12a, 12b and 12c

Figure 12. Domain parameters sensitivity analysis results

the number of cells. To finally select the parameters, values of lift and drag coefficient errors relative to the best solution were normalized and overlayed with normalized differences of cell numbers relative to the lowest number used (weighted x2 to account for two other results) (Fig. 12d). This last plot shows how favorable are different combinations of R and B are for the study. Overall, higher values of R and B should balance good results with reasonable computation time. However though, during the tests performed for GA calibration, it was found that for upper values of B, the choice of the smallest values of R from analysed range resulted in more than 50% savings on computation time due to fast convergence instead of waiting for reaching a predefined iteration limit. Because of that, and the fact that the domain size errors were shown to be quite predictable, finally a domain diameter R of 3 m was used for the final results. Final values are presented in Table 3.

Parameter	Description	Value	Unit
R	Domain radius	3	m
B	Cell size function exponent	3.6	-
$\mathrm{h_{wall}}$	First cell thickness	2×10^{-5}	m
$\mathrm{h_{far}}$	Last cell size	3×10^{-3}	m
$\mathrm{h_{total}}$	Maximum allowed boundary layer thickness	2.5×10^{-3}	m
G	Boundary layer cell size growth ratio	1.2	-
max	Maximum allowed cell size	1	m

Table 3. Final mesh parameters

3.3.3 Comments

The possible reasons for such a significant result mismatch include different influence of boundary condition implementations used (velocity inlet/pressure outlet vs freestream) and implementation of force coefficients library used in OpenFOAM. The geometric and other physical parameters are understood to be replicated as well as possible, but an error here also cannot be ruled out. While fig. 11 appears to match perfectly drawings from NACA TR 573, the airfoils as shown in the Appendix 1 of [11] seem to have larger gap.
It should be noted that irregularities in sensitivity analysis plots (except numbers of cells) come from running all cases once. When repeated and averaged, the results would probably have been smoother.

3.4 CFD solver setup

For the CFD solution, finite volume method OpenFOAM [24] toolbox was selected (and its steady-state simpleFoam solver in particular) thanks to its wide use and ease of automation. Following [11], the widely used Spalart-Allmaras single equation RANS turbulence model was used. The study was run on a personal PC with AMD Ryzen 3900X 12-core processor, and as of such the domain was decomposed into 12 subdomains and the solver was run in 12 parallel processes. Physical parameters were copied from sensitivity analysis (which were copied from NACA TR 824). After plotting $\mathrm{C_L}$ and $\mathrm{C_D}$ as a function of solution iteration of several cases it became apparent that the force coefficients stabilize earlier than the whole solution, and solutions with complicated flow (usually leading to unfavorable coefficients) can converge very slow or even stagnate. This led to setting a hard iteration limit of 10000.

Solver	SIMPLE
Discretization	Spalart-Allmaras

Parameter	Value	Unit
p	0	Pa
U	35.763	$\frac{\mathrm{m}}{\mathrm{s}}$
ρ	1.18	$\frac{\mathrm{kg}}{\mathrm{m}^3}$
ν	1.5e-5	$\frac{\mathrm{m}^2}{\mathrm{s}}$

Residual	Value
p	1e-5
U	1e-5
ν_t	1e-5

Table 4. Final solver settings, flow parameters and residual control

3.5 Results of the optimization procedure

3.5.1 Summary of the GA run

Due to necessity of multiple test runs of the entire system in order to fine-tune the parameters and decrease generator fragility, a suboptimal parameters were chosen. The final results were chosen from the 20 hour long test run at R equal to 3 m and B equal to 3.6. In total, the optimization lasted 200 generations, during which the fitness function was evaluated 1811 times, out of which 635 did not pass geometry validation and instantly got zeroed. In total, there were 1294 calculated cases, out of which 892 were shapes generated by the GA, and 402 were replacements for excessive dead members generated by the "safety mechanism" described in section 3.2. While generation of new cases was probably necessary, the fragility of the generator and possibly improper choice of crossover implementation caused the algorithm to work in cycles of saturation of population with similar specimens, followed by generation of multiple incorrect specimens and reversal to diverse population. These cycles can be seen on Fig. 13 as cycles of high and low mean fitness of the entire population.

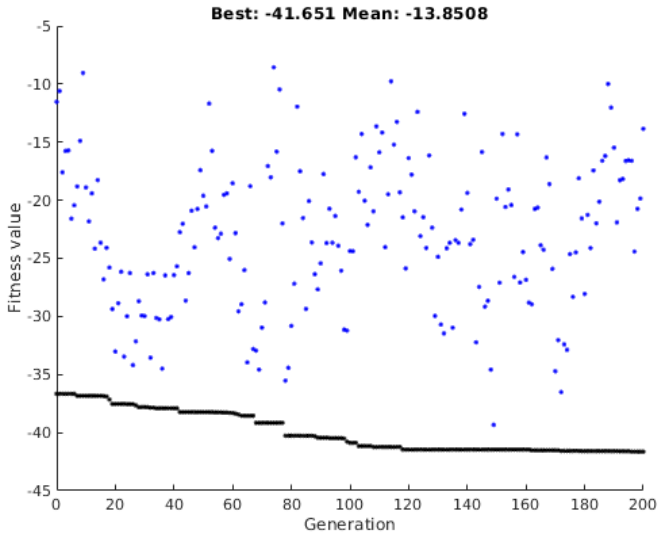
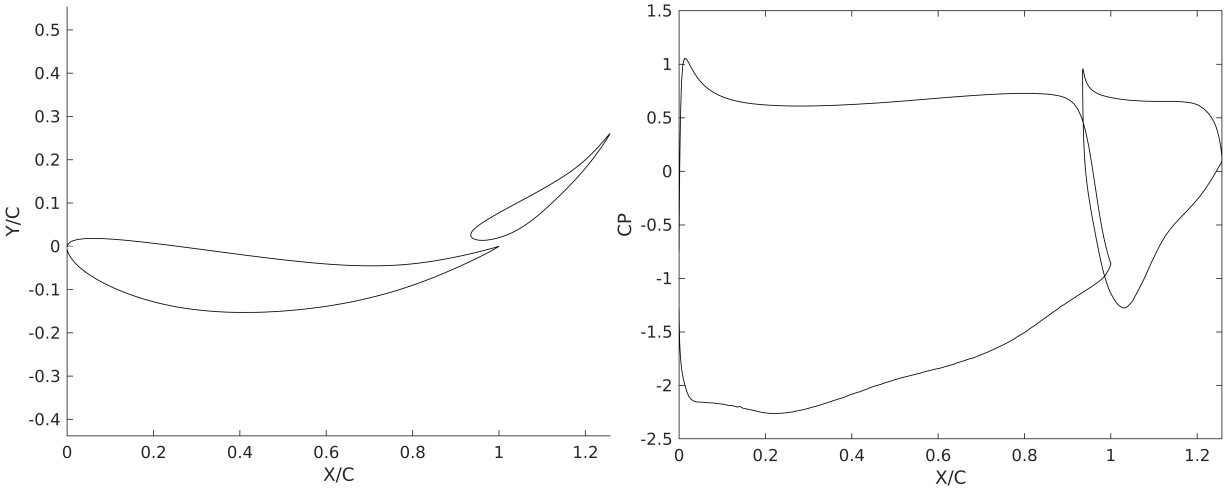


Figure 13. Mean and best values of fitness function

3.5.2 Final shape and result validation

The optimized shape appears to be similar to the wings used in practice (Fig. 14). Comparing to the fittest member of random initial population (Fig. 15), the lift coefficient increased by 10% at a cost of drag coefficient increase of 6.5% (Table 5).



(a) Final optimized shape

(b) C_P distribution over the final wing

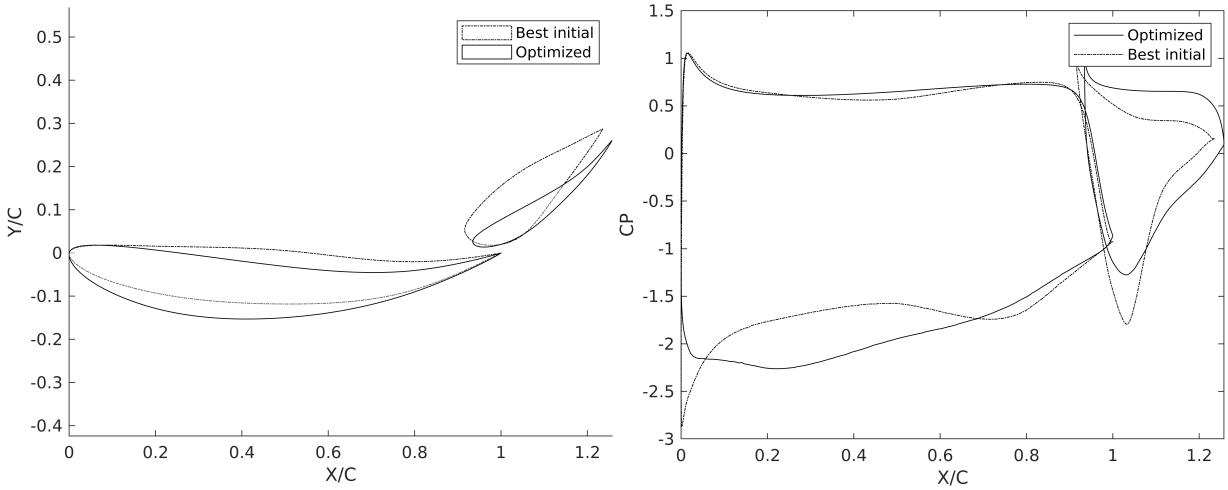
(c) Velocity magnitude distribution

(d) Pressure distribution

Figure 14. Final wing and its properties

-	GA optimized	Initial best	Absolute increase	Relative increase
C_L	-2.64897	-2.40788	0.24109	10.01%
C_D	0.168473	0.158121	0.010352	6.55%
$\frac{L}{D}$	-15.7235	-15.2281	0.4954	3.25%

Table 5. Results from the best wing obtained from GA and from its validation



(a) Final optimized shape

(b) Final shape vs best initial random shape

Figure 15. Comparison of the final wing to the fittest member of the randomly generated inital population

For final validation, the best wing was then recalculated on a much finer mesh for final validation. For the final run, first layer thickness $\mathrm{h_{wall}}$ was reduced in half, the boundary layer cell growth ratio G was reduced to 1.1, and R and B were set to 20 m and 2 respectively. As expected from subsection 3.3.2, lift coefficient increased and drag coefficient decreased. All resulting differences in force coefficients are presented in Table 6.

-	GA run	Validation run	Absolute error	Relative error
$\mathrm{C_L}$	-2.64897	-2.663653	0.014683	0.55%
$\mathrm{C_D}$	0.168473	0.147957	0.020516	13.87%
$\frac{\mathrm{L}}{\mathrm{D}}$	-15.7235	-18.002903	2.279403	12.66%

Table 6. Results from the best wing obtained from GA and from its validation

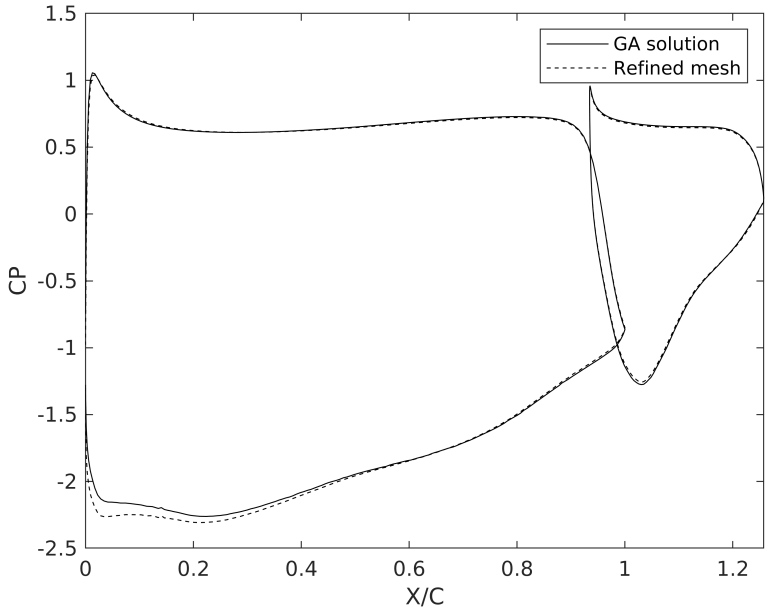


Figure 16. C_P of final shape with GA and validation meshes

Chapter 4

Conclusions

A new method of airfoil parametrization was proposed. The presented method is fragile, but capable of generating wide variety of airfoil shapes. The numerical study shows that while it is possible to use it with genetic algorithms in its current state, the fragility requires a special corrective approach for even the most basic work. Still, the method shows potential of high usability if the robustness of the entire system can be increased. A further investigation into autoencoder training is necessary, possibly together with refining other parts of the generator algorithm. It is also necessary to better prepare the optimization algorithm (or choose a completely different one) to better suit the generator behavior and its fragility, even if it is reduced. Finally, a more efficient mesh generation process, possibly with further tweaking of the FVM solver would allow for more detailed, and even more importantly broader search of optimal cases.

Bibliography

[1] *2D NACA 0012 Airfoil Validation - Effect of Farfield Boundary*. [Online]. Available: https://turbmodels.larc.nasa.gov/naca0012_val_ffeffect.html.

[2] Abbott, I. H., Von Doenhoff, A. E., and Stivers, L. J., „Summary of Airfoil Data," Tech. Rep., 1945. [Online]. Available: https://ntrs.nasa.gov/citations/19930090976.

[3] Brunton, S. L., Noack, B. R., and Koumoutsakos, P., „Machine learning for fluid mechanics," *Annual Review of Fluid Mechanics*, vol. 52, no. 1, pp. 477–508, 2020. DOI: 10.1146/annurev-fluid-010719-060214. [Online]. Available: https://doi.org/10.1146/annurev-fluid-010719-060214.

[4] Chen, W., Chiu, K., and Fuge, M., „Aerodynamic design optimization and shape exploration using generative adversarial networks," 2019. DOI: 10.2514/6.2019-2351.

[5] Cinnella, P. and Congedo, P., „Optimal airfoil shapes for viscous transonic flows of bethe–zel'dovich–thompson fluids," *Computers Fluids*, vol. 37, no. 3, pp. 250–264, 2008, ISSN: 0045-7930. DOI: https://doi.org/10.1016/j.compfluid.2007.04.007. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0045793007001065.

[6] Dubsky, J. and Pashias, C., „How machine learning transformed the porsche 919 hybrid evo," <https://medium.com/next-level-german-engineering/how-machine-learning-transformed-the-porsche-919-hybrid-evo-33d9881cb0e5>, dostęp uzyskano 2021-10-26.

[7] Ebrahimi, M. and Jahangirian, A., „Aerodynamic optimization of airfoils using adaptive parameterization and genetic algorithm," *Journal of Optimization Theory and Applications*, vol. 162, no. 1, pp. 257–271, 2014, ISSN: 1573-2878. DOI: 10.1007/s10957-013-0442-1. [Online]. Available: https://doi.org/10.1007/s10957-013-0442-1.

[8] Erickson, L. L., „Panel methods: An introduction," Tech. Rep., 1999. [Online]. Available: https://ntrs.nasa.gov/citations/19910009745.

[9] Geuzaine, C. and Remacle, J.-F., „Gmsh: A 3-d finite element mesh generator with built-in pre- and post-processing facilities," *International Journal for Numerical Methods in Engineering*, vol. 79, no. 11, pp. 1309–1331, 2009. DOI: https://doi.org/10.1002/

nme.2579. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.2579.

[10] Goodfellow, I., Bengio, Y., and Courville, A., *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[11] Goto, K. and Sakurai, H., „Numerical study for the optimal flap chord length of a two-element airfoil," in *Motorsports Engineering Conference & Exposition*, SAE International, 2006. DOI: https://doi.org/10.4271/2006-01-3643. [Online]. Available: https://doi.org/10.4271/2006-01-3643.

[12] Holland, J. H., „Adaptation in natural and artificial systems," 1975.

[13] Katz, J., *Race Car Aerodynamics: Designing for Speed*. Robert Bentley, Incorporated, 1996, ISBN: 9780837601427. [Online]. Available: https://books.google.pl/books?id=8PUNQwAACAAJ.

[14] Le, Q. T. and Ooi, C., „Surrogate modeling of fluid dynamics with a multigrid inspired neural network architecture," *Machine Learning with Applications*, vol. 6, p. 100 176, 2021, ISSN: 2666-8270. DOI: https://doi.org/10.1016/j.mlwa.2021.100176. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2666827021000888.

[15] McBeath, S., *Competition Car Aerodynamics 3rd Edition*. Veloce Publishing Ltd, 2017, ISBN: 9781787110861. [Online]. Available: https://books.google.pl/books?id=RIP4DQAAQBAJ.

[16] McLean, D., *Understanding Aerodynamics: Arguing from the Real Physics*. Wiley, 2012, ISBN: 9781119967514. [Online]. Available: https://books.google.pl/books?id=UE3sxu28R0wC.

[17] Piechna, J., *Podstawy aerodynamiki pojazdów*, pol. Warszawa: Wydawnictwa Komunikacji i Łączności, 2000, ISBN: 8320613663.

[18] Platt, R. C. and Abbott, I. H., „Aerodynamic Characteristics of NACA 23012 and 23021 Airfoils with 20-Percent-chord External-Airfoil Flaps of NACA 23012 Section," Tech. Rep., 1937. [Online]. Available: https://ntrs.nasa.gov/citations/19930091648.

[19] Selig, M., *UIUC airfoil data site*, English (US). Department of Aeronautical and Astronautical Engineering University of Illinois at Urbana-Champaign, 1996.

[20] Smith, A. M. O., „High-lift aerodynamics," *Journal of Aircraft*, vol. 12, no. 6, pp. 501–530, 1975. DOI: 10.2514/3.59830. [Online]. Available: https://doi.org/10.2514/3.59830.

[21] Sobieczky, H., „Parametric airfoils and wings," in *Recent Development of Aerodynamic Design Methodologies: Inverse Design and Optimization*. Vieweg+Teubner Verlag, 1999, ISBN: 978-3-322-89952-1. DOI: 10.1007/978-3-322-89952-1_4. [Online]. Available: https://doi.org/10.1007/978-3-322-89952-1_4.

[22] Vatandas, E. and Özkol, İ., „Coupling dynamic mesh technique and heuristic algorithms in 3-d-tapered wing design," *International Journal for Numerical Methods in Engineering*, vol. 74, no. 12, pp. 1771–1794, 2008. DOI: https://doi.org/10.1002/nme.2231. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.2231.

[23] Wang, Y., Shimada, K., and Farimani, A. B., *Airfoil gan: Encoding and synthesizing airfoils foraerodynamic-aware shape optimization*, 2021. arXiv: 2101.04757 [cs.CE].

[24] Weller, H. G., Tabor, G., Jasak, H., and Fureby, C., „A tensorial approach to computational continuum mechanics using object-oriented techniques," *Computers in Physics*, vol. 12, no. 6, pp. 620–631, 1998. DOI: 10.1063/1.168744. [Online]. Available: https://aip.scitation.org/doi/abs/10.1063/1.168744.

[25] *What y+ should I use? Part 3 – Understanding impact of Y+ and number of prism layers on flow resolution | Computational Fluid Dynamics (CFD) Blog – LEAP Australia & New Zealand*, [Online; accessed 27. Nov. 2021], 2021. [Online]. Available: https://www.computationalfluiddynamics.com.au/y-plus_part3_understanding-impact-of-y-and-number-of-prism-layers-on-flow-resolution.

[26] Yonekura, K., Wada, K., and Suzuki, K., *Generating various airfoil shapes with required lift coefficient using conditional variational autoencoders*, Jun. 2021.

[27] Zerihan, J., „An investigation into the aerodynamics of wings in ground effect," Ph.D. dissertation, University of Southampton, 2001. [Online]. Available: https://eprints.soton.ac.uk/426058/.

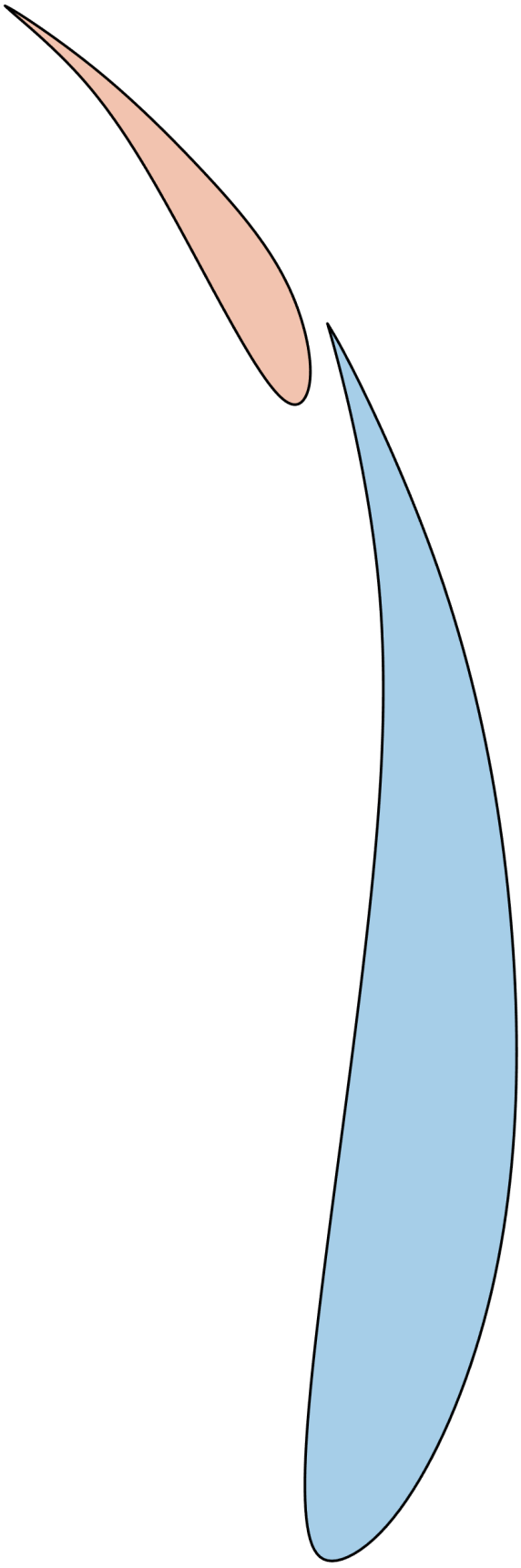
List of Figures

List of Appendices

Appendix I

Final optimized shape

In order to show the wing in the best possible detail, it is presented on the next page.



Appendix II

Final shape coordinates

Element 1

1.000000000000000	0
0.997323792817482	-0.001634317183465
0.994544569851707	-0.003314199865336
0.991559315319421	-0.005085213544021
0.988265013437367	-0.006992923717926
0.984558648422289	-0.009082895885458
0.980337204490931	-0.011400695545023
0.975497665864801	-0.013991888192579
0.969960566491008	-0.016889932535590
0.963738985266294	-0.020080710415853
0.956868731071295	-0.023538418311519
0.949385612786642	-0.027237252700738
0.941325439292969	-0.031151410061662
0.932724019470909	-0.035255086872441
0.923617162201096	-0.039522479611226
0.914040676364163	-0.043927784756168
0.904030370840743	-0.048445198785417
0.893622054511469	-0.053048918177124
0.882851536256974	-0.057713139409441
0.871754624957893	-0.062412058960517
0.860367129494857	-0.067119873308504
0.848724858748500	-0.071810778931552
0.836863621599456	-0.076458972307812
0.824819226928358	-0.081038649915435
0.812627483615839	-0.085524008232572
0.800324200542532	-0.089889243737374
0.787944557392934	-0.094109054194610
0.775507796424360	-0.098170834870746
0.763016362848223	-0.102075363400445
0.750471900607309	-0.105824055796806
0.737876053644405	-0.109418328072924
0.725230465902296	-0.112859596241895
0.712536781323768	-0.116149276316818
0.699796643851608	-0.119288784310787
0.687011697428601	-0.122279536236899
0.674183585997533	-0.125122948108252
0.661313953501190	-0.127820435937941
0.648404443882358	-0.130373415739063
0.635456701083824	-0.132783303524714
0.622472369048373	-0.135051515307992
0.609453091718791	-0.137179467101992
0.596400513037865	-0.139168574919811
0.583316276948380	-0.141020254774545
0.570202027393122	-0.142735922679292
0.557059408314877	-0.144316994647147
0.543890063656431	-0.145764886691207
0.530695637360570	-0.147081014824568
0.517477773370081	-0.148266795060328
0.504238115627749	-0.149323643411582
0.490978517876790	-0.150252661968061
0.477703156625113	-0.151051477277931
0.464417653561532	-0.151715553472171
0.451127652172798	-0.152240322065615

0.437838795945658	-0.152621214573096
0.424556728366863	-0.152853662509448
0.411287092923162	-0.152933097389505
0.398035533101303	-0.152854950728102
0.384807692388036	-0.152614654040072
0.371609214270110	-0.152207638840249
0.358445742234274	-0.151629336643467
0.345322919767277	-0.150875178964560
0.332246390355869	-0.149940597318361
0.319221797486798	-0.148821023219706
0.306254784646814	-0.147511888183427
0.293350995322666	-0.146008623724359
0.280516073001103	-0.144306661357336
0.267755661168875	-0.142401432597192
0.255075403312730	-0.140288368958759
0.242480942919417	-0.137962901956874
0.229977923475686	-0.135420463106369
0.217571988468286	-0.132656483922078
0.205268782477969	-0.129666396784704
0.193078108735907	-0.126448925513830
0.181023330698336	-0.123013526413742
0.169130571138471	-0.119371839699917
0.157425952829530	-0.115535505587834
0.145935598544728	-0.111516164292969
0.134685631057282	-0.107325456030800
0.123702173140407	-0.102975021016804
0.113011347567319	-0.098476499466460
0.102639277111236	-0.093841531595245
0.092612084545373	-0.089081757618636
0.082955892642945	-0.084208817752111
0.073696824177171	-0.079234352211147
0.064861001921264	-0.074170001211223
0.056474548648443	-0.069027404967814
0.048563587131922	-0.063818203696400
0.041154240144919	-0.058554037612457
0.034272630460648	-0.053246546931464
0.027944880852327	-0.047907371868897
0.022197042476693	-0.042548533814174
0.017052513004802	-0.037196177157564
0.012531316845326	-0.031894400253761
0.008653260230687	-0.026688462686032
0.005438149393306	-0.021623624037640
0.002905790565603	-0.016745143891849
0.001075989980000	-0.012098281831926
-0.000031446131083	-0.007728297441133
-0.000396711535223	-0.003680450302736
0	0
0.001105834841097	0.003144566153463
0.002915229562028	0.005930713488450
0.005416526836310	0.008377543427088
0.008598069337459	0.010504157391507
0.012448199738991	0.012329656803834
0.016955260714424	0.013873143086198
0.022107594937272	0.015153717660727
0.027892912886215	0.016190217152575
0.034288596259105	0.016997151941635
0.041263606191113	0.017585505426259
0.048786658264524	0.017966158154124
0.056826468061622	0.018149990672912
0.065351751164690	0.018147883530302
0.074331223156013	0.017970717273974
0.083733599617875	0.017629372451609
0.093527596132559	0.017134729610886
0.103681928282350	0.016497669299486
0.114165311649532	0.015729072065088
0.124946461816388	0.014839818455372
0.135994094365203	0.013840789018018
0.147276924878260	0.012742864300707
0.158763668937843	0.011556924851118
0.170423042126237	0.010293851216931
0.182223760025726	0.008964523945827
0.194134538218592	0.007579823585485
0.206124092287134	0.006150630683581
0.218166041514465	0.004686490910588
0.230253538301591	0.003191632664407
0.242384597992620	0.001668960560735
0.254557235931661	0.000121379215265
0.266769467462822	-0.001448206756307
0.279019307930213	-0.003036892738287

```
0.291304772677942  -0.004641774114982
0.303623877050118  -0.006259946270695
0.315974636390848  -0.007888504589732
0.328355066044242  -0.009524544456399
0.340763181354409  -0.011165161255001
0.353196997665457  -0.012807450369843
0.365654530321494  -0.014448507185232
0.378133794666630  -0.016085427085471
0.390632806044973  -0.017715305454867
0.403149579800631  -0.019335237677725
0.415682131277713  -0.020942319138350
0.428228475820328  -0.022533645221048
0.440786628772584  -0.024106311310125
0.453354605478591  -0.025657412789884
0.465930421282456  -0.027184045044633
0.478512091528288  -0.028683303458676
0.491097631560196  -0.030152283416318
0.503685058222964  -0.031588065798364
0.516272511580850  -0.032986540612729
0.528858344693791  -0.034341539338977
0.541440931781449  -0.035646688955351
0.554018647063490  -0.036895616440092
0.566589864759577  -0.038081948771442
0.579152959089373  -0.039199312927643
0.591706304272543  -0.040241335886937
0.604248274528752  -0.041201644627565
0.616777244077660  -0.042073866127769
0.629291587138935  -0.042851627365790
0.641789677932238  -0.043528555319871
0.654269890677235  -0.044098276968253
0.666730599593588  -0.044554419289177
0.679170178900962  -0.044890609260886
0.691587002819020  -0.045100473861622
0.703979445567426  -0.045177640069625
0.716345881365845  -0.045115734863138
0.728684684433940  -0.044908385220402
0.740994228991374  -0.044549218119660
0.753272889257813  -0.044031860539152
0.765519039452919  -0.043349939457120
0.777731053796356  -0.042497081851807
0.789907306507789  -0.041466914701454
0.802044444278187  -0.040254335141252
0.814119417555985  -0.038868721871543
0.826096650583635  -0.037328663423820
0.837940366818841  -0.035652895955570
0.849614789719305  -0.033860155624283
0.861084142742729  -0.031969178587449
0.872312649346815  -0.029998701002556
0.883264532989266  -0.027967459027094
0.893904017127782  -0.025894188818551
0.904195325220067  -0.023797626534417
0.914102680723822  -0.021696508332182
0.923590307096750  -0.019609570369334
0.932622427796553  -0.017555548803363
0.941163266280932  -0.015553179791757
0.949177046007591  -0.013621199492007
0.956627990434231  -0.011778344061600
0.963480323018554  -0.010043349658027
0.969698267218262  -0.008434952438777
0.975246046491058  -0.006971888561338
0.980107284449636  -0.005666905579708
0.984357005163441  -0.004504536783890
0.988097125083385  -0.003461014096726
0.991429564368241  -0.002512568296487
0.994456243176781  -0.001635430161440
0.997279081667777  -0.000805830469854
1.000000000000000  -0.000000000000000
```

Element 2

1.256844309773956	0.260535034607157
1.256285519618823	0.259453216910259
1.255699889420043	0.258335100104758
1.255060579133966	0.257144385082051
1.254340748716944	0.255844772733534
1.253513558125329	0.254399963950605
1.252552167315474	0.252773659624659
1.251429749043363	0.250929579458670
1.250127805314908	0.248843684624139
1.248650509459731	0.246525256255132
1.247005855924001	0.243989191366901
1.245201839153885	0.241250386974699
1.243246453595553	0.238323740093775
1.241147693695171	0.235224147739383
1.238913553898909	0.231966506926774
1.236552028652934	0.228565714671198
1.234071112403415	0.225036667987909
1.231478799596520	0.221394263892158
1.228783084678418	0.217653399399195
1.225991962095275	0.213828971524274
1.223113426293261	0.209935877282645
1.220155471718545	0.205989013689560
1.217126092817292	0.202003277760271
1.214033284035674	0.197993566510029
1.210885039819857	0.193974776954086
1.207689354616009	0.189961806107694
1.204453600519352	0.185968189074217
1.201180657548106	0.181997630791118
1.197871433661800	0.178049520672302
1.194526828369428	0.174123229639077
1.191147741179982	0.170218128612751
1.187735071602454	0.166333588514632
1.184289719145837	0.162468980266027
1.180812583319124	0.158623674788244
1.177304563631307	0.154797043002590
1.173766559591380	0.150988455830374
1.170199470708335	0.147197284192904
1.166604196491164	0.143422899011485
1.162981636448860	0.139664671207428
1.159332690090417	0.135921971702038
1.155658256924826	0.132194171416625
1.151959236461080	0.128480641272494
1.148236528208173	0.124780752190955
1.144491031675095	0.121093875093315
1.140723646370841	0.117419380900882
1.136935271804403	0.113756640534962
1.133126807484774	0.110105024916865
1.129299152920946	0.106463904967897
1.125453207621912	0.102832651609367
1.121589653392268	0.099211279260328
1.117707951924351	0.095603408787393
1.113807140115907	0.092013916678861
1.109886254432861	0.088447680699423
1.105944331341139	0.084909578613768
1.101980407306665	0.081404488186586
1.097993518795365	0.077937287182567
1.093982702273164	0.074512853366401
1.089946994205987	0.071136064502778
1.085885431059759	0.067811798356388
1.081797049300407	0.064544932691921
1.077680885393854	0.061340345274067
1.073535975806025	0.058202913867515
1.069361357002848	0.055137516236957
1.065156065450245	0.052149030147081
1.060919137614144	0.049242333362578
1.056649609960468	0.046422303648137
1.052346518955143	0.043693818768449
1.048008901064095	0.041061756488204
1.043635792753247	0.038530994572091
1.039226230488527	0.036106410784800
1.034779250735858	0.033792882891022
1.030293897263991	0.031595286763435
1.025772598583702	0.029517621357695
1.021226537026612	0.027561617694579
1.016668291302087	0.025728645022119
1.012110440119490	0.024020072588347
1.007565562188184	0.022437269641294

```
1.003046236217536   0.020981605428994
0.998565040916909   0.019654449199478
0.994134554995667   0.018457170200779
0.989767357163174   0.017391137680928
0.985476026128795   0.016457720887957
0.981273140601895   0.015658289069899
0.977171279291836   0.014994211474786
0.973183020907984   0.014466857350649
0.969320944159703   0.014077595945522
0.965597627756358   0.013827796507435
0.962025650407311   0.013718828284422
0.958617590821929   0.013752060524514
0.955386027709574   0.013928862475743
0.952343500641065   0.014250444372788
0.949501565953092   0.014714021734914
0.946870748270283   0.015312618408244
0.944461523376090   0.016039059805396
0.942284367053966   0.016886171338988
0.940349755087362   0.017846778421637
0.938668163259731   0.018913706465962
0.937250067354527   0.020079780884581
0.936105943155200   0.021337827090110
0.935246266445203   0.022680670495169
0.934694223730362   0.024052448013774
0.934417522231572   0.025495299689977
0.934416312195797   0.027007870531725
0.934690743869997   0.028588805546967
0.935240967501137   0.030236749743648
0.936067133336177   0.031950348129717
0.937169391622080   0.033728245713121
0.938547864753935   0.035569082953451
0.940198528530221   0.037470823149676
0.942108849782415   0.039430040042541
0.944265250969857   0.041443136821430
0.946654154551884   0.043506516675729
0.949261982987833   0.045616582794824
0.952075158737045   0.047769738368098
0.955080104258855   0.049962386584938
0.958263242012603   0.052190930634728
0.961610994457625   0.054451773706854
0.965109784053262   0.056741318990701
0.968746033258850   0.059055969675653
0.972506164533727   0.061392128951097
0.976376600337232   0.063746200006417
0.980343763128702   0.066114586030998
0.984394075367476   0.068493690214225
0.988513959512892   0.070879915745484
0.992689838024288   0.073269665814159
0.996908133361001   0.075659343609637
1.001156013800766   0.078045623109026
1.005427894919194   0.080427809599308
1.009722270518686   0.082806689068076
1.014037679950739   0.085183064040642
1.018372662566848   0.087557737042320
1.022725757718510   0.089931510598422
1.027095504757222   0.092305187234262
1.031480443034480   0.094679569475151
1.035879111901779   0.097055459846404
1.040290050710617   0.099433660873334
1.044711798812490   0.101814975081252
1.049142895558895   0.104200204995473
1.053581880301326   0.106590153141309
1.058027292391282   0.108985622044072
1.062477671180258   0.111387414229077
1.066931556019751   0.113796332221636
1.071387486261256   0.116213178547062
1.075844001256271   0.118638755730668
1.080299640356292   0.121073866297767
1.084752942912814   0.123519312773672
1.089202448277335   0.125975897683695
1.093646695801352   0.128444423553151
1.098084224836359   0.130925692907351
1.102513574708265   0.133420508307346
1.106933176114790   0.135929824028875
1.111341102772357   0.138455092922727
1.115735355175872   0.140997870103940
1.120113933820239   0.143559710687550
1.124474839200365   0.146142169788595
1.128816071811154   0.148746802522111
```

```
1.133135632147512   0.151375164003134
1.137431520704343   0.154028809346703
1.141701737976554   0.156709293667853
1.145944284459048   0.159418172081621
1.150157160646733   0.162156999703044
1.154338367034512   0.164927331647160
1.158485904117291   0.167730723029004
1.162597772389975   0.170568728963613
1.166671972347469   0.173442904566025
1.170706504484680   0.176354804951276
1.174699369296511   0.179305985234403
1.178648567277868   0.182298000530443
1.182552098923656   0.185332405954433
1.186407964728781   0.188410756621408
1.190214165188148   0.191534607646407
1.193968700796662   0.194705514144466
1.197669572049228   0.197925031230622
1.201314668201806   0.201194335625422
1.204899498005152   0.204506506435351
1.208417284580618   0.207846847885855
1.211861158795618   0.211200350387970
1.215224251517562   0.214552004352732
1.218499693613860   0.217886800191177
1.221680615951923   0.221189728314341
1.224760149399162   0.224445779133261
1.227731424822988   0.227639943058972
1.230587573090812   0.230757210502511
1.233321725070044   0.233782571874914
1.235927011628095   0.236701017587216
1.238396563632376   0.239497538050455
1.240723511950297   0.242157123675665
1.242900987449270   0.244664764873883
1.244922120996706   0.247005452056146
1.246780043460014   0.249164175633489
1.248467885706606   0.251125926016948
1.249978778603893   0.252875693617560
1.251309481954610   0.254403665094341
1.252478695518362   0.255731442782358
1.253513322394553   0.256892375323531
1.254440279920696   0.257919831747229
1.255286485434303   0.258847181082819
1.256078856272886   0.259707792359673
```