Studium Magisterskie

Kierunek Analiza danych – Big Data
Specjalność

Imię i nazwisko autora
Mikołaj Zabiegliński
Nr albumu 116333

Usage of genetic algorithm search in aerodynamics as a means of decreasing aircraft development costs in the aviation industry

Praca magisterska
pod kierunkiem naukowym
dr Grzegorza Kolocha
Instytut
Ekonometrii

Warszawa 2024

Abstract

The purpose of this work was to explore the possibility of implementing genetic algorithms in aircraft design, as a means of reducing development costs and times. A brief summary of the aviation industry was given, which concluded with the competition in the aircraft manufacturer sector and the increasing aircraft development prices. Potential steady 2D flow was introduced as an aerodynamic model and the panel method for an aerofoil as an example of computational method in aerodynamics. The genetic algorithm (GA) was introduced and described. An application of GA tools towards aerofoils was proposed. In Python, a computing environment was created to obtain solutions for aerofoils in potential steady 2D flow, based off *AeroPython*. NACA0012 aerofoil was mutated to seed the first generation of solutions for GA. Multiple iterations of preliminary searches, including 2 parameter sweeps, were conducted with the proposed GA implementation over the aerodynamic search space. The obtained results showed the strength of GA principles and the inherent difficulties of computational fluid dynamics (CFD).

Table of contents

Introduction

The subject of this work is application of evolutionary algorithms to aerodynamic problems, as a means of showcasing ways of applying non-classic computation methods, specifically genetic algorithms, to save computation time and in turn save money.

In the first chapter, the general structure of the aviation industry is briefly discussed: its interconnected and highly dynamic nature. The industry's size is demonstrated through its contribution to GDP and raw revenues and profits. Additionally, the response to the COVID-19 pandemic is showcased as a testimony of aviation industry's short-term volatility and long-term stability. There is additional focus on the aircraft manufacturers market, including the Boeing-Airbus duopoly in the wide-body aircraft market and the resulting competition. Increasing development costs of such jets are highlighted to present the need of improving design time and costs – precisely where meta-heuristics would prove useful. Most of the numbers in this chapter, especially for the around-COVID period are sourced from state-of-the-art EASA and IATA reports while their interpretation comes from literature written by academic experts on the general topic of aviation industry.

The next chapter explains how CFD is used in modern aerodynamic surfaces design. Introduced and examined, based on classic academia literature for fluid mechanics, are: 2D potential flow aerodynamic model and the panel method which can be used to computationally approximate this flow for arbitrary surfaces. Barba and Mesnard's work in *AeroPython* which presents an engineering-student-level implementation of the panel method is recalled. An aerofoil is considered as a solution in a meta-heuristic and a general objective function for such aerofoil is introduced. Examples of studies examining other evolutionary methods in aerodynamics are given, which helped conceptualize the form of the following GA search.

The following chapter on application of the genetic algorithm focuses on what the genetic algorithm is (based Goldberg's renowned textbook), where it can be used (research in economics and genetics given as examples), and how the tools of genetic algorithm – reproduction, crossover, and mutation operators – were adapted to be used for aerofoils over bit-strings traditionally used in GA.

The last chapter is dedicated to the aerodynamic search using GA which fundamentals for had been described in previous chapters. The environment for the search was constructed in iPython and built upon the aforementioned *AeroPython* implementation. The first generation was artificially created by mutating a NACA0012 profile. After the initial search, which helped

visualize the limitations of the algorithm and direct the following iterations, 2 consecutive parameter sweeps were conducted to determine the search algorithm's parameters. Finally, a 200-generations-long search proper was conducted independently on 2 sets of parameters. Results were examined and compared.

It was ultimately concluded that while the genetic algorithm implementation performed properly as expected and showcased GA's universality and a way of applying it in the aviation industry, the results were likely not physically fit. Different solutions were put forward: improvement of the GA operators, improvement of the aerofoil modelling, and advancement of the aerodynamic model.

Optimization of costs in aviation industry

Aviation is a very dynamic industry where different types of operations interact. A breakdown of its structure, the net figures, and its competitive nature was made to showcase why optimization of aircraft development costs might be beneficiary.

Aviation industry structure

The aviation industry consists of all operations of commercial air transport and operations related to it. Chains of services naturally occur within it. The minimum required to transport a single passenger from point A to B is an aeroplane, which is operated by an airline, and 2 separate aerodromes, each of which hosts multiple independent enterprises. A logistic chain arises because an air traveller does not consider their choice of airline, or single connection, in a vacuum but also, and primarily, takes into consideration the final destination itself. Since in an airline's interest is for their flight to be chosen by a passenger, the airline has to work towards making the entire chain that includes this flight attractive enough for the passenger, who is their customer (Ison, Budd 2021, p. 1).

The intuition could be that this chain structure leads itself to unification of services within chains to larger market actors. At the very least, interconnected services, structure of which is universally similar, could dictate that there is a set direction to aviation industry's development. Neither of these is true. Changes in the aviation industry structure happen frequently and while they indeed follow some trends, they do not converge with time to a set framework. Instead, these changes usually correspond to new technological developments, such as debut of jet

engine in commercial aircraft, singular market decisions or new legislations; the latter 2 of which happened following the 9/11 attack in 2001 (Belobaba, Odoni, Barnhart 2009, p. 1-12). It is a market of great uncertainty (Ison, Budd 2021, p. 2).

However, the industry is not just passenger transport and the services related to that. The second facet of commercial aviation industry is freight transport, which in aviation carries also mail and passenger luggage, in addition to usual types of freight. The same logistic chains appear when considering cargo aeroplanes. Some airlines act as "combination", servicing both passengers and freight – it must be noted here that passenger carriers are of course equipped with cargo compartments and can take on board multiple types of freight in addition to human load (Belobaba, Odoni, Barnhart 2009, p. 47-48). In terms of numbers, 4543 thousand passengers and 61 thousand metric tonnes of freight were carried by commercial aircraft in 2019 (Ison, Budd 2021, p. 5). This is about 13 kg of freight for every passenger. Coincidentally, the mass of the average passenger with carry-on luggage in 2022 was 83.3 kilograms (Berg et al., 2022, p. 36).

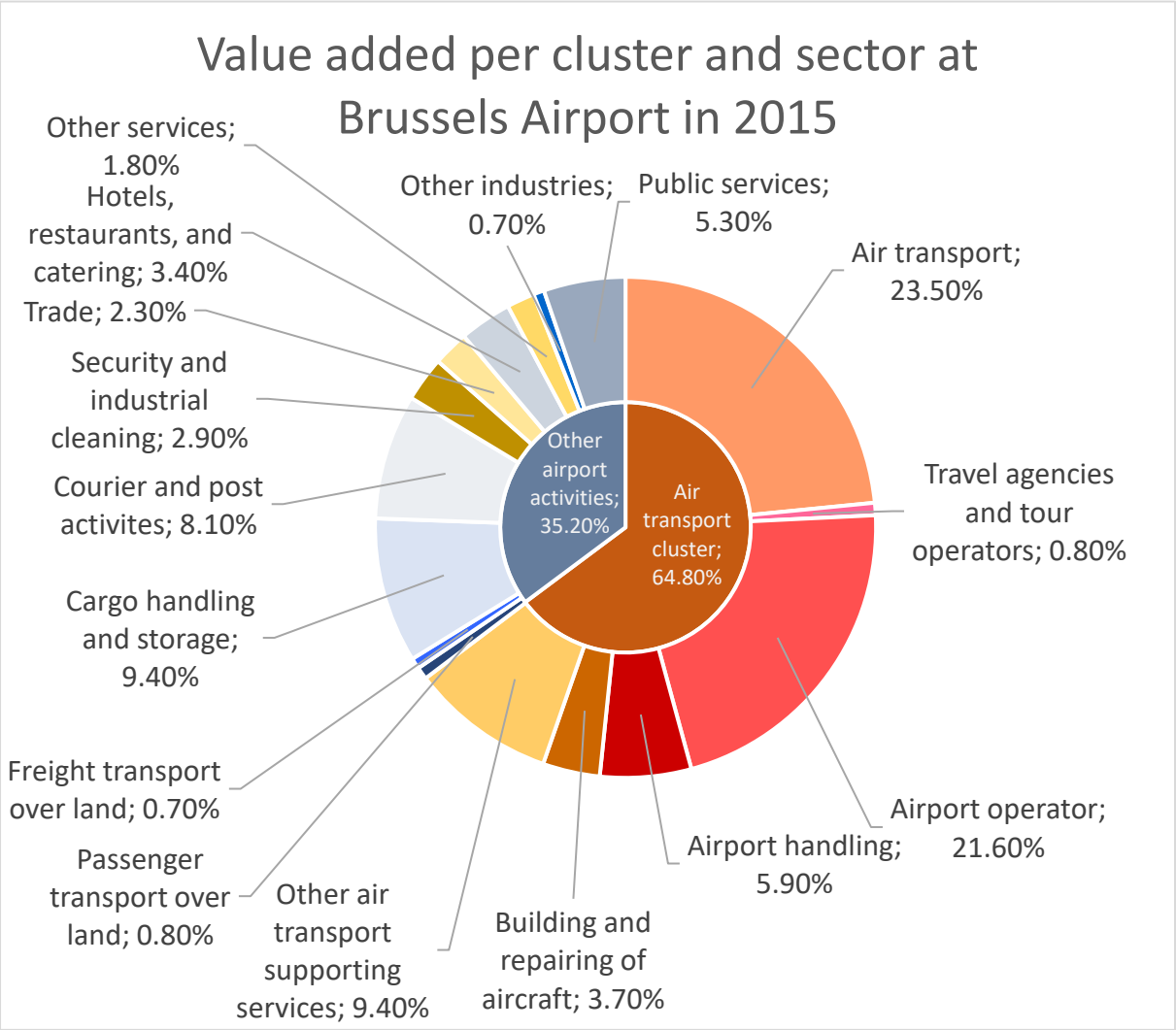

Figure 1: Value added of items in air transport *and* other *clusters at Brussels Airport in 2015, showcasing the industry structure. Based on Table 1.7 from (Ison, Budd, 2021).*

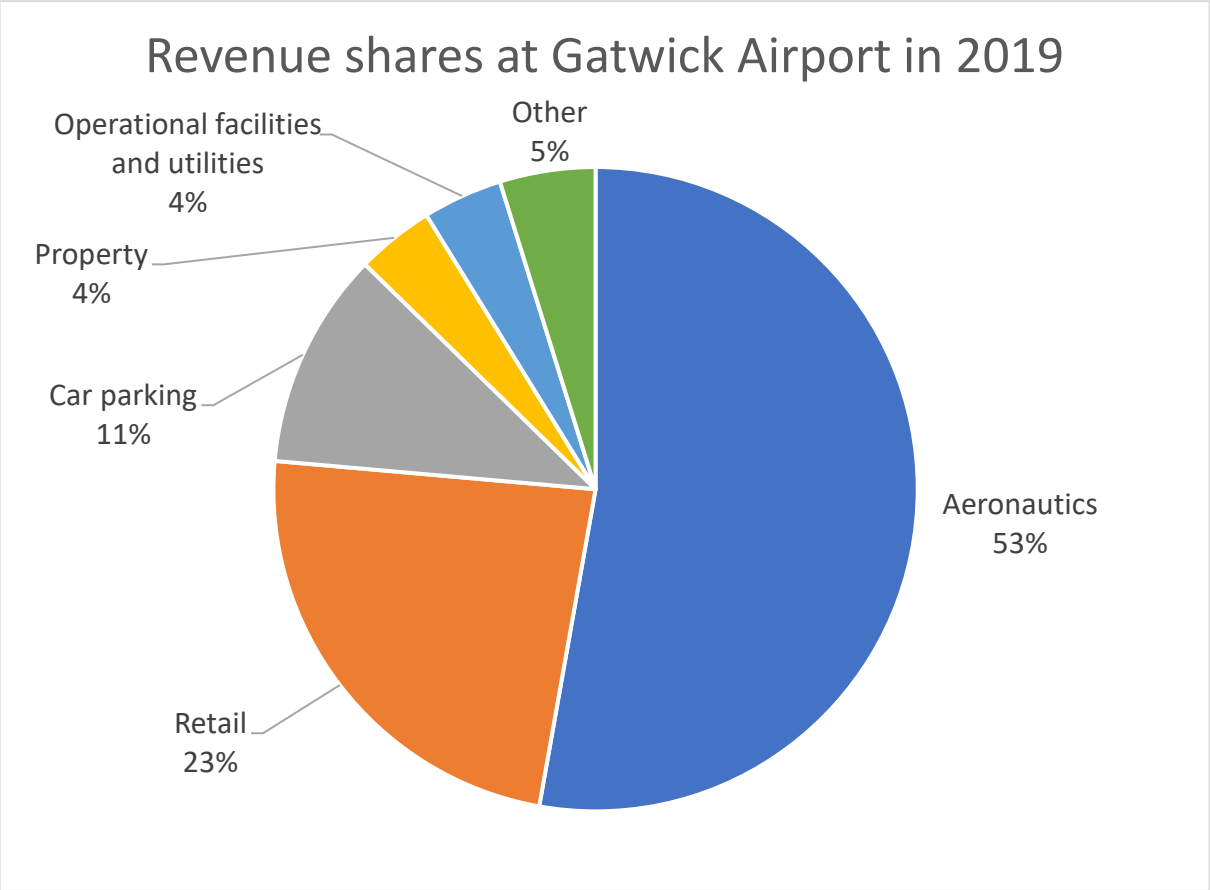

Figure 2: Revenue shares of items at Gatwick Airport in 2019, showcasing the industry structure. Based on Table 1.6 from (Ison, Budd, 2021).

Size and impact of the aviation industry

As of 2009, there had already been more than 2000 operating airlines, 23 000 commercial aircraft, and 3700 aerodromes in the world. In 2007 alone, more than 2 billion passengers in 29 million flights were transported. At that point, the annual growth of air travel had been doubling the annual growth of GDP. In 2008, aviation was estimated to be 7,5% of global GDP (Belobaba, Odoni, Barnhart 2009, p. 2-4). European Union Aviation Safety Agency (EASA), the agency governing civil aviation in the European Union, just on U27+EFTA airports registered 8 million flights in 2005 and 9 million flights in 2019, a 15% increase. The number of passengers on them increased by staggering 71%, from 478 million in 2005 to 818 million in 2019 (European Environment Agency, EASA 2022, p. 28).

As for most, if not all, industries and markets in the world, short-term dips do happen. The net profits for aviation actually decreased globally from 2014 to 2019, even as the passenger and cargo revenue increased. This of course corresponds to an increase of operating costs, which could be attributed to the ever so increasing oil price, with oil being required to manufacture

fuel for aircraft – both commercial airliners and cargo aircraft primarily use jet engines powered by fossil fuels, steady inflation, and increased interest rates. International Air Transport Agency (IATA), the prime international body for air transport, interprets this period of annual profit decrease as a showcase of the aviation industry's elasticity – its ability to fit itself with the changing markets – and a sign of strength and not weakness (IATA 2024, p. 14). In 2019, IATA reported 4543 million passengers – 500 times as much as what EASA reported at its aerodromes in that time period. This corresponded to a net profit of 25.9 billion USD (Ison, Budd 2021, p. 6).

Impact of COVID-19 on the aviation industry

The global aviation industry predictably and visibly suffered in the aftermath of the COVID-19 pandemic outbreak. Only 4 million flights took place in 2020. Similar dips occur for most, if not all, commercial air traffic metrics. Total annual flown distance, total passenger number, total passenger kilometres, and number of average passengers per flight all decreased between 2019 and 2021. This decrease was indeed the most dire for the total passengers. While from 2005 to 2019 the annual passenger number for commercial traffic skyrocketed from 478 to 818 million, it was only 229 million in 2019. Nevertheless, the recovery has already begun. 5 million commercial flights were estimated to take place in 2021, with 304 million passengers on board (European Environment Agency, EASA 2022, p. 28).

An argument for the size, condition, and the trajectory of the aviation industry even more compelling than the swiftness of its recovery from the deep COVID-19 recession are the metrics forecasted following the pandemic. EASA predicts a base 113% increase in commercial air traffic passengers between 2005 and 2050, with the highest estimate of 155% increase (European Environment Agency, EASA 2022, p. 28). IATA in their last estimates for 2024 does not reach as far as 2050 but the forecasts are similarly optimistic – they prognose a 32% increase in passenger global passenger numbers between 2023 and 2027 and a compound annual growth rate of 3.8% between 2023 and 2043 (IATA 2024, p. 9-10).

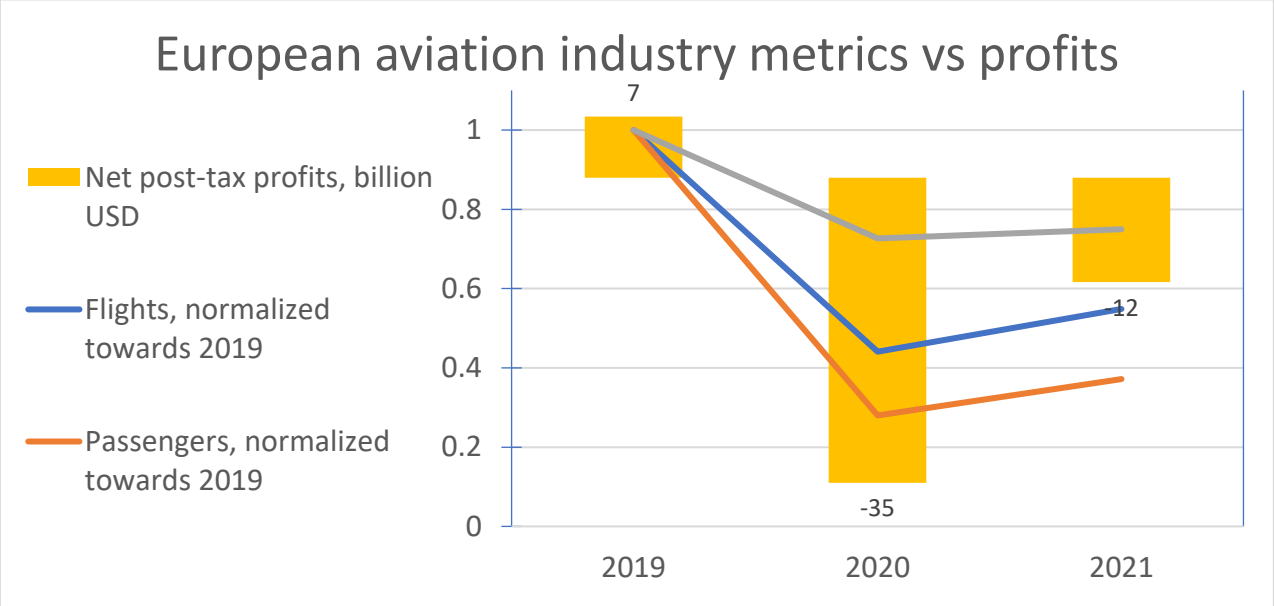

Figure 3: The change in European aviation industry metrics in the aftermath of the COVID-19 pandemic. Reproduced from Figure 1.7 (European Environment Agency, EASA 2022).

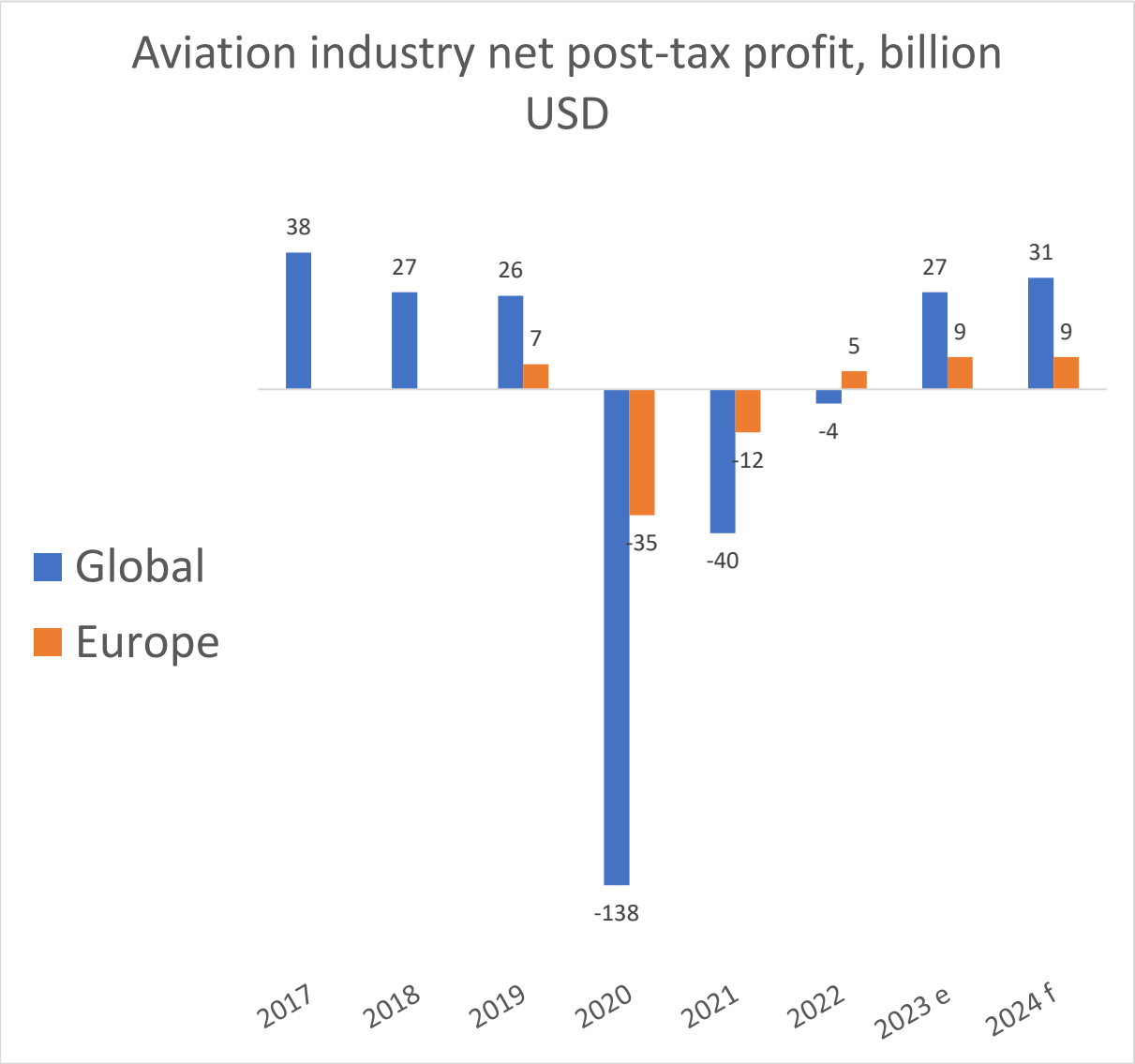

Figure 4: The change in profits in global and European aviation industry pre-, during, and post-COVID-19 pandemic. e - estimates, f - forecasts. Based on Table 1.2 from (Ison, Budd, 2021) and Table 7 from (IATA 2024).

Aircraft manufacturers market

Aircraft manufactured for large scale passenger aviation generally falls into one of two categories: narrow-body or wide-body. Narrow-body describes a single aisle airliner. These usually carry from 100 to 200 passengers and have a shorter range of 6000 km at most. Wide-body refers to a double aisle airliner which carries from 200 to 450 passengers. Their range goes up to 14 000 km, but it is more varied within this category and further segmentation exists there – different aeroplanes will service 10 000 km and 14 000 km flights (Irwin, Pavcnik 2004).

There are multiple competitive manufacturers for narrow-body aircraft. Most notable are Airbus, Boeing, Embraer, and Bombardier. You will see all of those makes on any international

airport that also handles local connections. However, the wide-body market is an example of a duopoly. Airbus and Boeing are the only providers of wide-body aircraft in the world. The scale effects are in full force here. Because of their size, each of the two is subsided by a different government – European Community for Airbus and the United Stated for Boeing. These subsides serve as a means of preventing a monopoly from emerging if 1 of the companies pulls ahead. Such a monopoly was actually enjoyed by Boeing 747 as the only-range wide-body aircraft, before Airbus debuted A-380. Furthermore, evidence has been found by researchers that the 1992 agreement between Europe and USA to limit subsidies lead to aircraft price increase (Irwin, Pavcnik 2004).

The aviation industry downsizing, which manifests itself not only in lower annual revenues, but also in aircraft delivery delays and lower aircraft demand, has been speculated to further intensify the market conflict between Boeing and Airbus – the lowered demand for wide-body airliners might even force them to compete with narrow-body aircraft manufacturers, such as the aforementioned Bombardier (Ison, Budd 2021, p. 20-21). Nevertheless, this sector of the industry is also rebounding and 26% more aircraft deliveries are scheduled globally for 2025 than there were in 2018 (IATA 2024, p. 18).

Aircraft prices

With numbers like this, it is apparent that aviation is a booming industry where a lot of money is to be made and to be saved. A large part of expenses are the aircraft themselves. The 2015 order backlog for Airbus and Boeing, combined totalled to 1.5 trillion USD. In the former's case, this was worth of just a decade of orders (Clark 2017). In 2015, a brand new Boeing 777-200ER was priced at 277.3 million USD. Such a high price for a single aircraft is dictated, amongst others, by the increasing development costs. Boeing 777, first serviced in 1995, cost the company 7.0 billion USD to develop. For the Boeing 787, first serviced in 2012, the development cost was almost double, at 13.4 billion USD (Rodrigue 2017).

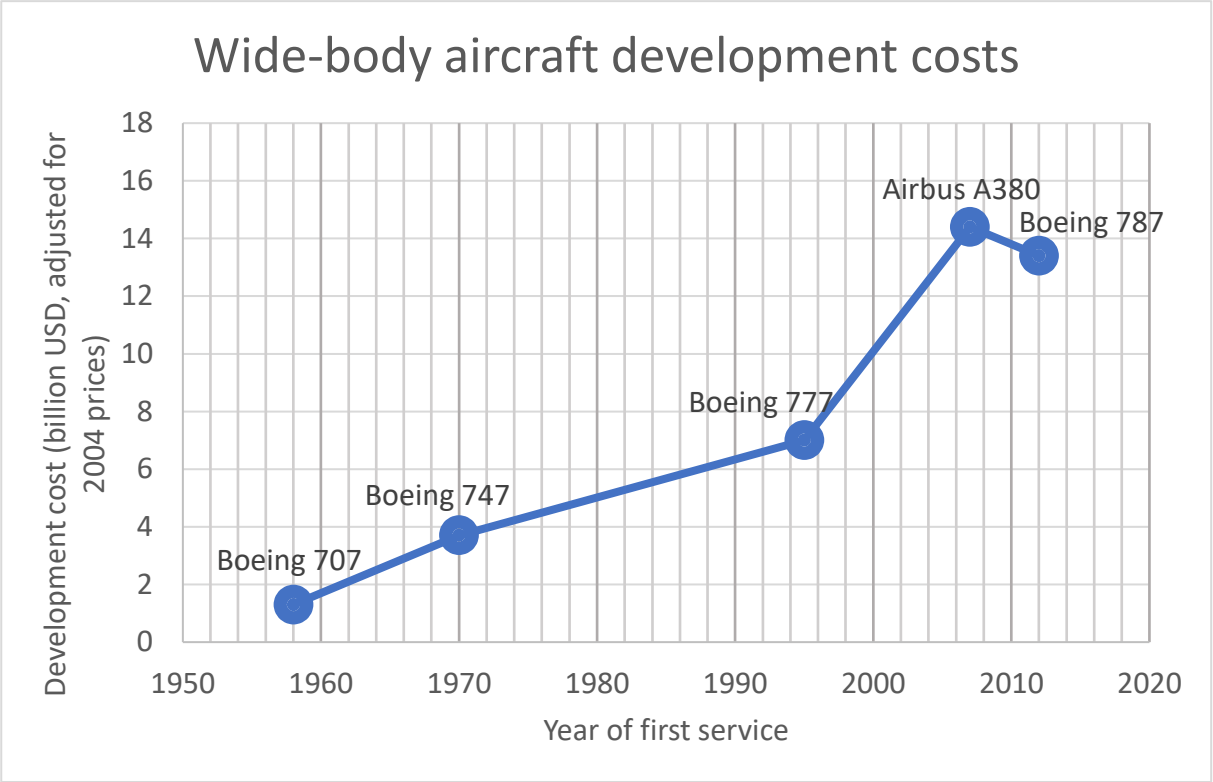

Figure 5: : Increasing costs of developing wide-body airliners. Based on the table in (Rodrigue 2017).

Decreasing development cost of aerodynamic surfaces through search optimization

A commercial aeroplane is a complicated machine, with many different systems and solutions that are worked on and implemented individually, before the whole vehicle is evaluated. One of these individual aspects are the aerodynamic surfaces. An aeroplane has to have a proper shape to generate lift and take-off, which makes air travel possible in the first place. Knowledge of aerospace engineering is required to design, simulate, evaluate, and test an aerodynamic surface. However, once an objective function for an aerodynamic surface is defined, the problem of searching for surfaces becomes purely mathematical and can be solved without intrinsic knowledge of aerospace models. This gives an aircraft manufacturer ability to reduce their research costs, by employing search algorithms to optimize the engineering portion of the development. Such optimization methods could be used in the aviation industry to increase net income of the manufacturers.

Aerodynamic search space and objective function

Aerodynamics are a complex field of study where there is no closed-form solution of a problem more often than not. This chapter explains how aerodynamic models are considered in practice, introduces 1 simple analytical model and its computational implementation, and proposes a method of evaluating a fitness of an aerodynamic solution. Research works in similar topic are referenced.

Usage of optimization methods for aerofoils

Before the development of computer methods and sufficient advancement of hardware, all aerofoil performance tests had to be conducted in aerodynamic laboratories. The laboratory workers had to put hundreds of physically crafted aerofoil models into wind tunnels, to measure their parameters (Anderson 2016, p. 406).

In modern times, such tests are only conducted in the very final stages of aerofoil development. Before that, aerodynamic properties of all candidate designs are extensively studied on the engineers computers', using computational fluid dynamics (CFD). CFD allows to compute flows that were impossible to evaluate prior, using discretization to separate complex non-linear problems into small, manageable linear chunks, albeit at high computational cost (Anderson 2016, p. 189-191).

As the domain of all possible aerodynamical shapes considered for each application is infinite and the time for aeroplane design is finite, it is impossible to study all cases. Hence, the designers must carefully choose the virtual profiles they will create and test, as the computing time is valuable. To limit the computational costs, numerical optimization methods are used. Such methods have been presented both for aerofoils used in aviation, in the form of both genetic algorithms (GA) and adjoint methods (Cayiroglua, Kilic 2016), and in wind turbines, where GA (Jureczko, Pawlak, Mężyk 2005) and response surface methodology (RSM) combined with bi-objective mesh adaptive direct search (BiMADS) (Benim, Diederich, Pfeiffelmann 2018) have been used.

The aim of was to explore to which extent genetic algorithm (GA) be used in search of aerofoils with preferable aerodynamic characteristics. As models commonly used in aviation industry are extremely complex and cannot be easily built and computed, a simpler model was used as the basis for the flow calculation around a singular aerofoil. Potential, steady two-dimensional (2D)

flow around an aerofoil in a freestream was examined. Panel method was used to obtain solutions for aerofoils of arbitrary shape. This way, lift coefficient could be obtained for any closed curve in 2 dimensions. Lift coefficient, coupled with the aerofoil's geometrical properties, was then used to determine the curve's viability as an aerofoil.

Potential steady 2D flow theory and application

A steady flow is a flow that does not change with time. In such a flow, all the parameters are set once and do not change. By definition, there is no turbulence and the pathlines, streaklines, and streamlines coincide (White 2011, p. 41).

A 2D flow is a flow where only 2 dimensions are considered. The fluid can only move on a plane. This is often applicable in theoretical aerodynamics as the interest lies in properties of particular cross-sections instead of full bodies. In particular, Kutta-Joukowski theorem is applicable in two dimensions, allowing one to calculate lift per unit span of a body (aerofoil) in potential air flow:

$$L' = \rho U_\infty \Gamma \quad (0)$$

, where L' denotes lift per unit span, ρ denotes air density, U_∞ denotes the freestream velocity and Γ denotes circulation around the aerofoil, defined as follows (Anderson 2016, p. 282):

$$\Gamma = \oint_A \boldsymbol{V} \cdot ds \quad (1)$$

Potential flow is a term used for fluid flows which can be described by a solution to Laplace's equation:

$$\nabla^2 \Phi = 0 \quad (2)$$

, where Φ denotes the velocity field V potential of the flow:

$$V = \nabla \Phi \quad (3)$$

This greatly simplifies the calculations, as Laplace's equation is linear. However, the model can only be obtained from the more general flow equations (infamous Navier-Stokes equations in particular) if the viscous effects are neglected (White 2011, p. 530-531).

For 2D flows, viscosity of air is negligible everywhere, but very close to solid walls (such as an aerofoil's surface), hence this simplification is acceptable for educational purposes and preliminary calculations.

The linearity of Laplace's equation is important because it allows one to superpose different elementary solutions on each other, obtaining solutions that correspond to wind tunnel experiment results.

For $V = [u, v]$ following elementary solutions exist for (2):

Solution's descriptive name	Solution's constants	Solution's closed form
Source/sink Source for $\sigma \geq 0$ Sink for sigma $\sigma \leq 0$	Flow rate σ Position $(x_{\text{source}}, y_{\text{source}})$	$\begin{cases} u(x,y) = \frac{\sigma}{2\pi} \frac{x - x_{\text{source}}}{(x - x_{\text{source}})^2 + (y - y_{\text{source}})^2} \\ v(x,y) = \frac{\sigma}{2\pi} \frac{y - y_{\text{source}}}{(x - x_{\text{source}})^2 + (y - y_{\text{source}})^2} \end{cases}$
Vortex	Circulation Γ Position $(x_{\text{vortex}}, y_{\text{vortex}})$	$\begin{cases} u(x,y) = \frac{\Gamma}{2\pi} \frac{y}{(x - x_{vortex})^2 + (y - y_{vortex})^2} \\ v(x,y) = -\frac{\Gamma}{2\pi} \frac{x}{(x - x_{vortex})^2 + (y - y_{vortex})^2} \end{cases}$
Freestream	Freestream velocity U_∞ Angle of attack α	$\begin{cases} u(x,y) = U_\infty \cos(\alpha) \\ v(x,y) = U_\infty \sin(\alpha) \end{cases}$

Table 1: Elementary solutions to Laplace's equation

As a source is the only solution that creates a divisive streamline (a solid wall, like an aerofoil's boundary), a vortex is the only solution that creates circulation (necessary for lift to exist) and a freestream is the solution corresponding to a freestream with no disturbance, a complete model of a flow over a body is the superposition of the 3:

$$\begin{cases} u(x,y) = \frac{\sigma}{2\pi} \frac{x - x_{\text{source}}}{(x - x_{\text{source}})^2 + (y - y_{\text{source}})^2} + \frac{\Gamma}{2\pi} \frac{y}{(x - x_{vortex})^2 + (y - y_{vortex})^2} + U_\infty \cos(\alpha) \\ v(x,y) = \frac{\sigma}{2\pi} \frac{y - y_{\text{source}}}{(x - x_{\text{source}})^2 + (y - y_{\text{source}})^2} - \frac{\Gamma}{2\pi} \frac{x}{(x - x_{vortex})^2 + (y - y_{vortex})^2} + U_\infty \sin(\alpha) \end{cases} \quad (4)$$

However, solutions in this form correspond only to a few specific geometries, such as a cylinder or Rankine's oval. To obtain a solution for a steady potential 2D flow around an arbitrary geometry (such as an aerofoil's boundary curve), it is necessary to use discretization (Barba, Mesnard 2014). In this case, a panel method was chosen.

A crucial result in the theory of potential flow is the Kutta condition. The Kutta condition states that velocities at the trailing edge of the body – the point where the lower and upper halves of an aerofoil meet in a sharpened tip – must be equal in both magnitude and direction. This is a result of the velocity field being continuous. The same principle is responsible for the existence of the stagnation point – a point at the front of the aerofoil where the velocity is equal to 0 (Anderson 2016, p. 338-341). Mathematically, it means:

$$\forall s \in aerofoil\ curve: \lim_{h \to 0} V\ (s + h) = \lim_{h \to 0} V\ (s - h) \quad (5)$$

The final result that comes useful in these considerations is the lack of velocity normal to the aerofoil surface. Since the flow is inviscid, all velocity vectors on the aerofoil's curve are strictly tangent to it (Barba, Mesnard 2014):

$$\forall s \in aerofoil\ curve\ \wedge\ n \perp \text{curve(s)}:\ V(s) \cdot n = 0 \quad (6)$$

Panel method

On the boundary curve of any body (for example, an aerofoil) in a potential 2D flow, there is continuous function of volume flow rate σ, equivalent to a curve made of infinitely many sources placed next to each other, and a continuous function of circulation Γ, equivalent to a curve made of infinitely many vortices placed next to each other (Anderson 2016, p. 284-289, 369-370).

The panel method is a technique of discretizing the boundary curve into a set of connected straight line segments, each with being a single singularity sheet – a continuous line of sources or vortices. Then, by setting velocity at the centre of each panel tangent to the panel, circulation constant throughout the curve (equal on each panel), and the Kutta condition met, a system of equations is obtained which allows one evaluate the velocity and circulation at each panel, and hence the potential velocity field and the total circulation (Barba, Mesnard 2014).

The panel method is predominantly used in of 2 variants: either using only sources (source panel method) or only vortices (vortex panel method). However, the 2 can be combined into source-vortex panel method to increase the model's accuracy, at the price of increasing computing complexity. In this work, source-vortex panel method was used. Source-vortex panel method allows to apply equation (4) to arbitrary geometry, through discretization.

The following equation (Barba, Mesnard 2014) for potential flow at the i-th panel with start point (x_{a_i}, y_{a_i}), end point (x_{b_i}, y_{b_i}) and centre point (x_{c_i}, y_{c_i}) can be obtained:

$$\phi(x_{c_i}, y_{c_i}) = U_\infty x_{c_i}\cos(\alpha) + U_\infty y_{c_i}\sin(\alpha)$$
$$+\sum_{j=1}^{N}\frac{\sigma_j}{2\pi}\int_j \ln\left(\sqrt{\left(x_{c_i}-x_j(s_j)\right)^2+\left(y_{c_i}-y_j(s_j)\right)^2}\right)ds_j$$
$$-\sum_{j=1}^{N}\frac{\gamma}{2\pi}\int_j \arctan\left(\frac{y_{c_i}-y_j(s_j)}{x_{c_i}-x_j(s_j)}\right)ds_j \quad (7)$$

, where α is the angle of attack (angle between the freestream and the coordinate system) and N is the total number of panels and

$$\begin{cases} x_j(s_j) = x_{b_j} - s_j\sin\beta_j \\ y_j(s_j) = y_{b_j} - s_j\cos\beta_j \end{cases}$$

$$\wedge\begin{cases} \beta_j = \arcsin\left(\dfrac{y_{b_j}-y_{a_j}}{\sqrt{\left(x_{b_j}-x_{a_j}\right)^2+\left(y_{b_j}y_{a_j}\right)^2}}\right) \text{ if } x_{b_j} < x_{a_j} \\ \beta_j = \pi + \arcsin\left(\dfrac{-\left(y_{b_j}-y_{a_j}\right)}{\sqrt{\left(x_{b_j}-x_{a_j}\right)^2+\left(y_{b_j}-y_{a_j}\right)^2}}\right) \text{ if } x_{b_j} \geq x_{a_j} \end{cases} \quad (8)$$

Requirement of velocity tangency on the curve (and hence on each panel) provides (Barba, Mesnard 2014):

$$0 = V_\infty\cos(\alpha-\beta_i) + \frac{\sigma_i}{2} + \sum_{j=1,j\neq i}^{N}\frac{\sigma_j}{2\pi}\int_j \frac{\partial}{\partial n_i}\ln\left(\sqrt{\left(x_{c_i}-x_j(s_j)\right)^2+\left(y_{c_i}-y_j(s_j)\right)^2}\right)ds_j$$
$$-\sum_{j=1,j\neq i}^{N}\frac{\gamma}{2\pi}\int_j \frac{\partial}{\partial n_i}\arctan\left(\frac{y_{c_i}-y_j(s_j)}{x_{c_i}-x_j(s_j)}\right)ds_j \quad (9)$$

And by substituting (7) in the Kutta condition, tangential velocity at each panel can be obtained (Barba, Mesnard 2014):

$$V_{t_i} = \frac{\partial}{\partial t_i}\left(\phi(x_{c_i}, y_{c_i})\right) \rightarrow$$

$$V_{t_i} = V_\infty \sin(\alpha - \beta_i) + \sum_{j=1, j\neq i}^{N} \frac{\sigma_j}{2\pi} \int_j \frac{-(x_{c_i} - x_j)sin(\beta_i) + (y_{c_i} - y_j)cos(\beta_i)}{(x_{c_i} - x_j)^2 + (y_{c_i} - y_j)^2} \mathrm{d}s_j$$
$$- \sum_{j=1, j\neq i}^{N} \frac{\gamma}{2\pi} \int_j \frac{(x_{c_i} - x_j)cos(\beta_i) - (y_{c_i} - y_j)sin(\beta_i)}{(x_{c_i} - x_j)^2 + (y_{c_i} - y_j)^2} \mathrm{d}s_j - \frac{\gamma}{2} \quad (10)$$

Ultimately, this leads to a matrix equation:

$$Ax = b \quad (11)$$

, where:

$$A = \begin{bmatrix} A_{11} & \dots & A_{1N} & \sum_{i=1}^{N} B_{1i} \\ \vdots & \ddots & \vdots & \vdots \\ A_{N1} & \dots & A_{NN} & \sum_{i=1}^{N} B_{Ni} \\ K_1 & \dots & K_N & K_{N+1} \end{bmatrix} \wedge x = \begin{bmatrix} \sigma_1 \\ \vdots \\ \sigma_N \\ \gamma \end{bmatrix} \wedge b = \begin{bmatrix} -U_\infty \cos(\alpha - \beta_1) \\ \vdots \\ -U_\infty \cos(\alpha - \beta_N) \\ -U_\infty(\sin(\alpha - \beta_1) + \sin(\alpha - \beta_N)) \end{bmatrix}$$

In equation (11), A represents singularities and contains elements representing source (A_{ii}), vortex (B_{ii}), and Kutta condition (K_{ii}) influence. b represents freestream influence, and x represents source strength and vortex circulation for each panel.

The equation can be solved for x to obtain the circulation and lift coefficient for which the aerofoil can be optimized (Barba, Mesnard 2014):

$$C_l = \frac{\sum_{i=1}^{N} \gamma l_i}{\frac{1}{2} U_\infty c} \quad (12)$$

, where l_i is the length of i-th panel and c the chord of the aerofoil.

Fitness and objective function for an aerofoil in potential flow

The parameters that determine an aerofoil's quality are the lift and drag generated by it and its susceptibility to flow separation. Since drag and flow separation are strictly related to viscosity, they cannot be obtained from a potential flow model. Instead, an aerofoil's geometric properties were considered when specifying the objective function.

The starting point of the search was the well-known and widely used classic NACA profile NACA0012. NACA0012 is a symmetrical profile, hence it only generates non-zero lift when the angle of attack is non-zero.

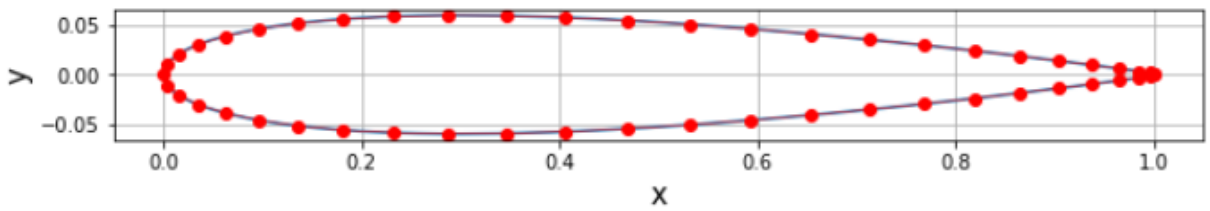

Figure 6: NACA0012 aerofoil profile. Red dots designate panel endpoints.

Fitness of a profile was determined by comparing it against NACA0012. Profiles that fared better than NACA0012 had a fitness greater than 1, the ones that fared worse had a fitness lower than 1 and the fitness of NACA0012 as specified by the objective function was 1.

The general objective function used in the initial search was given as:

$$F(profile) = \frac{A\left(\frac{C_l(profile)}{C_{l_{NACA0012}}}\right)}{B\left(\frac{shadow(profile)}{shadow_{NACA0012}}\right)} - r \cdot b(profile) \quad (13)$$

, where A, B are positive-definite functions, r is a positive constant, and b is a penalty function.

A profile's fitness was specified as the value of the objective function for the profile, and hence the goal was to maximize the objective function.

Geometric parameters as substitutes for drag and separation

For each aerofoil profile, parameters *shadow* and *roughness* were defined. *Shadow* is the vertical span of the profile and is supposed to emulate drag for preliminary calculations. *Roughness* is the total number of local extrema in the profile's upper curve and is supposed to emulate its susceptibility to flow separation.

For example, for NACA0012 *shadow* is *0.1200344* and *roughness* is *1*, as the upper curve has only 1 local extremum.

As aerofoils with more than 2 local extrema do not exist, *roughness* < *3* was chosen as a constraint for the problem.

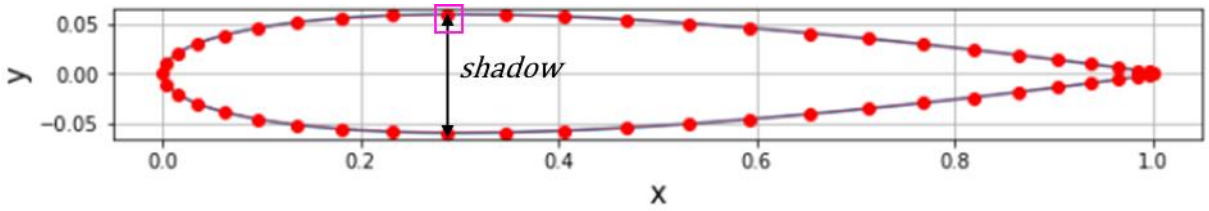

Figure 7: NACA0012's shadow and only local extremum on the upper curve marked.

Application of genetic algorithm towards the aerodynamic search space

There is no commonly used method of implementing meta-heuristics, let alone evolutionary algorithms, for aerodynamic solutions. An application of genetic algorithm to the earlier introduced panel method model is discussed and proposed here.

The principles of evolutionary algorithm

The idea behind evolutionary algorithm (EA) is emulating the evolutionary process that was famously observed and described by Charles Darwin. In EA, each singular solution is considered as a specimen of a generation.

Just as each specimen in evolutionary theory has some advantageous and some disadvantageous features, so does the specimen-solution. This advantage is measured by a fitness function that evaluates the solution, like for any other heuristic algorithm. Solutions in each generation breed with each other and produce offspring. To emulate natural selection, the specimen are more likely to be selected for breeding if they are better evaluated by the fitness function. Each act of this breeding spawns offspring solutions (or just a single one), that inherit partial features of each parent. Then the breeding process is repeated for a new generation created from the just-made offspring and so on, emulating the evolutionary process that nearly single-handedly uplifted the life on the earth from single-cell organisms to cetaceans, cephalopods, and apes.

The principles of genetic algorithm

The genetic algorithm is a specific form of the evolutionary algorithm that considers only bit-strings of finite length. A series of 1s and 0s is used to encode the features of a solution and these series undergo the evolution, exchanging substrings to create offspring and flipping 0s to 1s to represent mutation (Goldberg 1989, p. 7-9). A full bit-string is customarily referred to as a *chromosome* and a *gene* is some feature that might be encoded within it (Goldberg 1989, p. 22).

The basis of genetic algorithm are 3 operators: reproduction, crossover, and mutation. They are, in turn, responsible for selecting best solutions for reproduction, crossing the chosen solutions over, and applying a random change to the resulting solution with a specified, low probability. Reproduction and crossover help steer the solution towards the best traits while mutation protects against the crossover removing traits that are potentially advantageous (Goldberg

1989, p. 10-14). There are many ways to define each of these operators, depending on the problem being solved.

The simplicity and universality of GA has made it a popular choice amongst researchers in many different disciplines. Some naturally tend themselves to it – like molecular recognition and design (Willet 1995) – and some are less obvious – like frameworks for real options (Zhang, Babovic 2011).

Translation of GA principles to aerofoil search

The basis of this study was the observation that the GA principles of splicing strings could be similarly used to splice physical objects, especially those of elongated shape, such as aerofoils. Treating each aerofoil as a bit string encoding certain features, like stagnation point, or geometrical extrema, allows the methods of genetic algorithm to be generalized to these objects of large scale.

In the examined problem, the search space consisted of all 2D aerofoils discretized using panel method. Each aerofoil was a solution for which the fitness function was to be evaluated. It was decided to create an initial population of aerofoils of similar scale and shape, which would make it possible to easily mix and match them with each other while still generating somewhat physically viable solutions.

The main obstacle in the search of aerodynamic profiles was protection against propagation of disallowed solutions. The search space's constraints were against the number of local extrema (*roughness*). However, while crossing 2 solutions over was likely to increase *roughness*, it was unlikely to reduce it. For this reason, the genetic algorithm implementation focused on promoting fitter solutions and increased likelihood of mutation.

The other problem encountered was the complexity of the problem being solved itself. Despite the simplifications, evaluating the fitness of a singular solution was nevertheless computationally expensive. This also made the computational cost of every step in the algorithm where a profile's fitness was not evaluated negligible.

Reproduction

For the reproduction operator, the roulette wheel selection was chosen. Roulette wheel selection randomly selects 2 solutions from the pool of the entire generation. The probability of selecting a solution is linearly proportional to its fitness. The name comes from the fact that the selection process simulates spinning a roulette wheel, where the size of each field depends on the fitness.

Roulette wheel was chosen as it promotes fitter solutions more than other reproduction algorithms, namely tournament and rank selection, do.

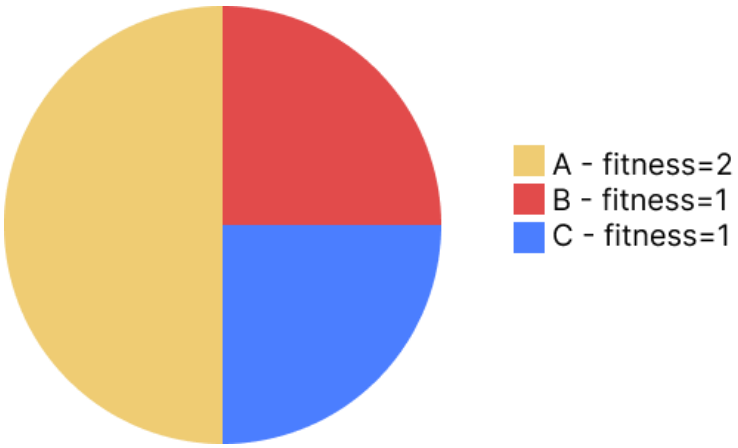

Figure 8: Roulette wheel selection visualized - the probability of selecting solution A is twice the probability of selecting B and twice the probability of selecting C since it is twice as fit as each of them.

Crossover

A crossover operator is responsible for selecting traits from multiple parent solutions and generating a single offspring solution that combines them.

Because of the solutions being aerodynamic profiles, which are inherently prone to sudden geometry changes (perform the best when smooth), it was decided to perform the crossover geometrically. An offspring solution was created by slicing 2 parent solutions and combining the slices. The point of slicing was chosen randomly along the x axis, using uniform distribution. Only 1 offspring was created this way. The unused slices were ignored. This method mirrors a single-point crossover of 2 chromosomes.

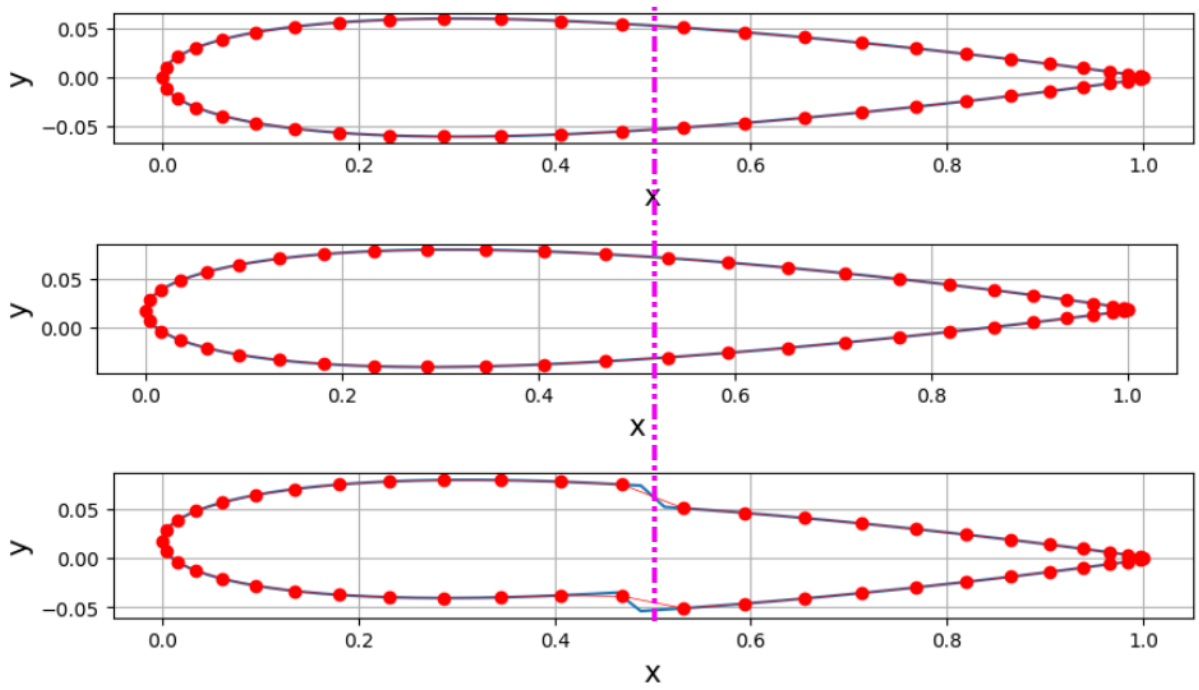

Figure 9: The crossover operator applied to 2 different solutions at 50% of the chord. The slicing is applied to the geometry itself and not the panel endpoints. A new set of panels is generated for each curve. In order: aft parent, front parent, offspring.

Mutation

Mutation was realized by warping the shape of an aerofoil. The mutation operator selected a random point μ (using uniform distribution) alongside the x axis and applied the following transformation to all y coordinates of the aerofoil's curve:

$$\begin{cases} y'(x) = y(x) + y_+(x) \\ y_+(x) = \dfrac{1}{\sqrt{2\pi}} \exp\left(\dfrac{-(x-\mu)^2}{2}\right) \cdot d \\ d = shadow_{NACA0012} \cdot \theta \end{cases}$$

, where $y(x)$, $y'(x)$ are the aerofoil's curve y coordinates pre and post-mutation and θ is a real number chosen randomly from a specified interval.

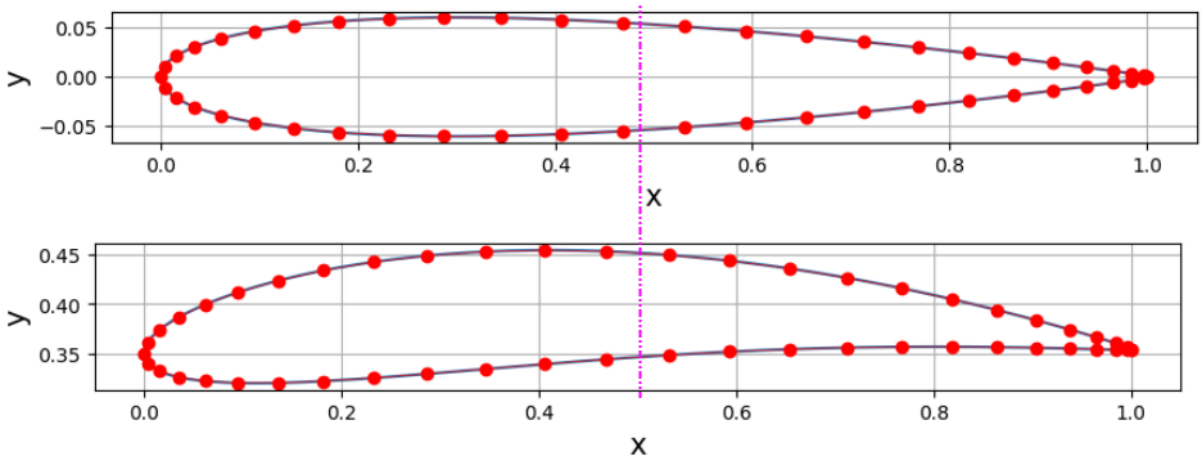

Figure 10: NACA0012 and NACA0012 post-mutation. d = 1, μ = 0.5. Note the translation of the profile along y axis.

Customarily, the mutation rate (probability of a new solution mutating upon its creation through reproduction) is specified as a very low value and impacts only a very small portion of the solution – typically 1 bit per 1000 (Goldberg 1989, p. 12). However, for the conducted aerodynamic search, the mutation operator defined affects the entire aerofoil – albeit with the impact diminishing from the centre μ. Additionally, the mutation rate was specified at a very high rate – initially 0.5. These 2 deviations from the classic approach were responsible for high variety of new solutions in each generation which secures a wide range of aerofoil shapes (solution types) even when the population size is small, which was desirable because of the computation time constraints posed by the problem's numerical complexity.

Computing

This chapter describes the implementation of the so far introduced theoretical search of solutions (panel-method-described-aerofoils) in an aerodynamic search space of potential steady 2D flow using GA. The preliminary and proper GA searches in this implemented computational environment and their results are examined.

Environment

All the computations were performed in Python 3.10.5 extended with packages SciPy and NumPy. Package matplotlib was used for visualizations. The code implementing the aerodynamic model was imported from a *.py* file while the genetic algorithm search was conducted in Jupyter environment, using *.ipynb* files to track progress and easily extend executable code.

The aerodynamic model implementation was the one used by Barba and Mesnard in their publicly available lecture series called *AeroPython*. It is an introductory CFD course which focuses on panel method in potential flow (Barba, Mesnard 2014). The *AeroPython* code was augmented. The augmentation consisted of creating a new Python class *Profile* which stored information about each aerodynamic profile (each being 1 solution in the search space). The class was equipped with methods and attributes necessary to conduct the search through genetic algorithms. Namely, they allowed to compute the aerodynamic and geometric parameters for each *Profile* class instance, apply the mutation operator to the profile, and display the shape of both the profile curve and panels, as well as the coefficient of pressure chart. The class was also fitted with an ID attribute that eased identifying each solution.

All aerodynamic parameters were calculated for a preset angle of attack $AoA = 4°$.

As GA is an inherently stochastic algorithm, a fixed random seed of 666 was chosen for all computations, to keep the results verifiable with the code provided.

Search parameters

The parameters that had to be defined for each iteration of the search through the space were the aerodynamic conditions (AoA in the freestream), the exact form of the objective function, the size of each generation, the total number of generations, the mutation scale factor θ*'s* range, and the probability of mutation in each generation.

The values for some of these were changed and adjusted through parameter sweeps and other were set *a priori*.

Search iteration	Freestream AoA	Objective function	Generation size	Number of generations	Mutation probability	Mutation scale factor range
First generation creation	4°	None	100	1	1	$\langle -0.01, 0.01 \rangle$
Initial search		1 set closed form	100	61	0.5	
1st parameter sweep		Varied	1000	21	Varied	
2nd parameter sweep		Varied	1000	21	0.1	
Final search		2 separate closed forms	1000	221	0.1	

Table 2: Summary of all phases of the conducted GA search. Specifics elaborated upon in respective sections.

First generation

The first generation for the genetic algorithm was created by replicating the NACA0012 profile 100 times, resulting in 100 solutions, and applying mutation once on each of the copies. The interval for θ was specified for these mutations as $\langle -1, 1 \rangle$. This meant that the mutation could dent a profile up or down up to a maximum equal to its total thickness.

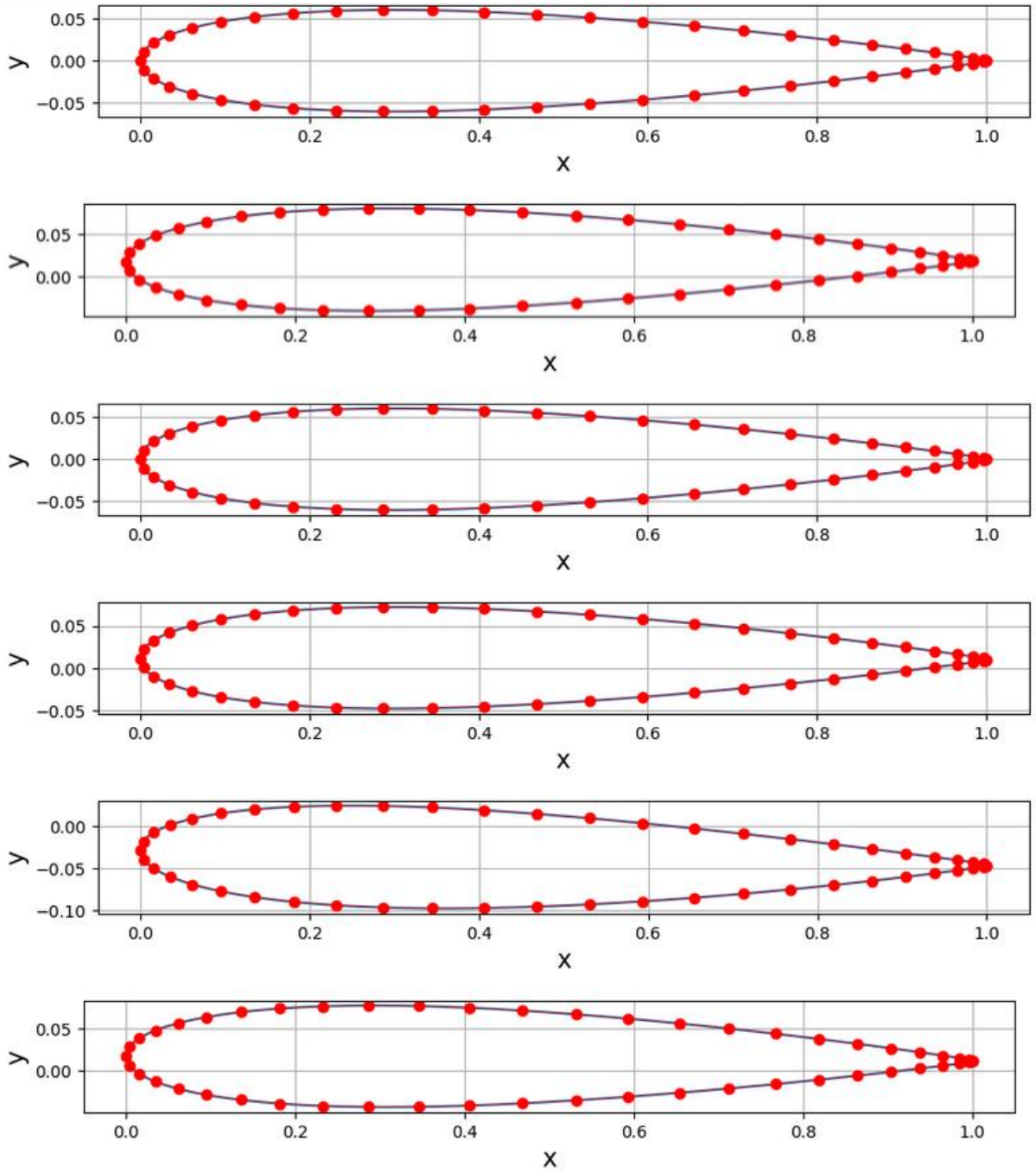

Figure 11: 6 examples of profiles mutated from NACA0012 for the solutions in the first generation of the GA search.

Initial search

In the initial search, 61 generations, excluding the first preset generation, were created. Each generation consisted of 100 specimen (profiles, solutions). The 2nd generation was created as a means of testing the algorithm and the following 60 were used as a means of exploring the search space further. The number was chosen based on the computing limitations of the hardware.

In the initial search, the following objective function was used:

$$F(profile) = max\left(\frac{\left(\frac{C_L(profile)}{C_{L_{NACA0012}}}\right)}{\left(\frac{shadow(profile)}{shadow_{NACA0012}}\right)} - r \cdot b(profile), \quad 0\right)$$

, with the penalty component defined as:

$$r = 0.1 \ \wedge \ b(profile) = \begin{cases} (roughness(profile) - 2)^2 \text{ if } roughness(profile) \geq 2 \\ roughness(profile) \text{ if } roughness(profile) < 2 \end{cases}$$

The probability of mutation was 0.5, and the interval for θ mutation scale factor $\langle -0.01, 0.01 \rangle$.

After the search was completed, the solution with the highest and lowest fitness were examined.

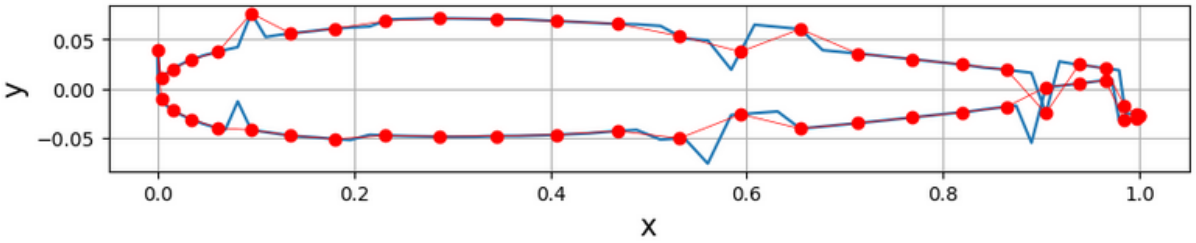

Figure 12: The solution with the best fitness obtained in the initial search. Fitness≈56134301.45, C_L≈30246044.37

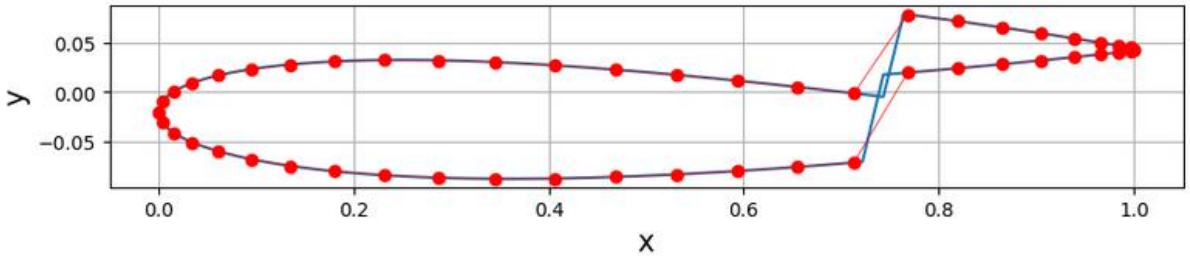

Figure 13: The solution with the worst fitness obtained in the initial search. Fitness=0, C_L≈0.02

It was observed that "the best" solution was from one of the later generations as it featured multiple abrupt changes in curvature which corresponded to instances of inherited crossover. The worst solution had only 1 such crossover artifact. It was also additionally noted that "the best" solution was not actually good. Not only was it physically impossible with multiple instances of the upper and lower curve of the aerofoil crossing over each other, it also had multiple local extrema that would disrupt the flow and cause flow separation. The parameter *roughness* was introduced and included in the penalty function to protect against it. The penalty was seemingly not harsh enough, especially compared to extreme and physically impossible lift coefficient of over 30 million.

When the predetermined constraint of *roughness* < 3 was applied to the solution of the initial search, a different best solution was obtained.

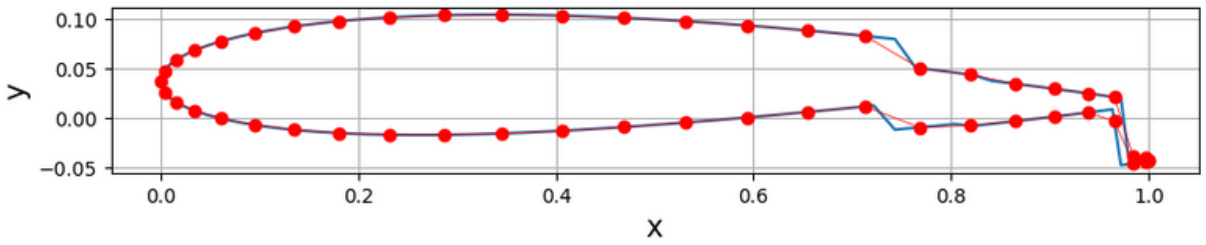

Figure 14: The solution with the best fitness obtained in the initial search that met the problem constraints. Fitness≈4.19, C_L≈2.70

The best solution that met the problem constraints had *roughness* equal only to 1, despite its shape indicating that it must have been crossed over at least twice (and was therefore of generation 3 or later). The solution had a realistic lift coefficient of 2.70.

Finally, the fitness development through generations was examined. It was observed that if the constraint of *roughness* < 3 is disregarded, then the both the maximum and average fitness radically increased with each generation.. However, if the constraint was included, then the solutions had similar maximum fitness in each generation following the initial one, but the average fitness decreased each generation. Additionally, solutions with *roughness* < 3 no longer appeared past generation 13.

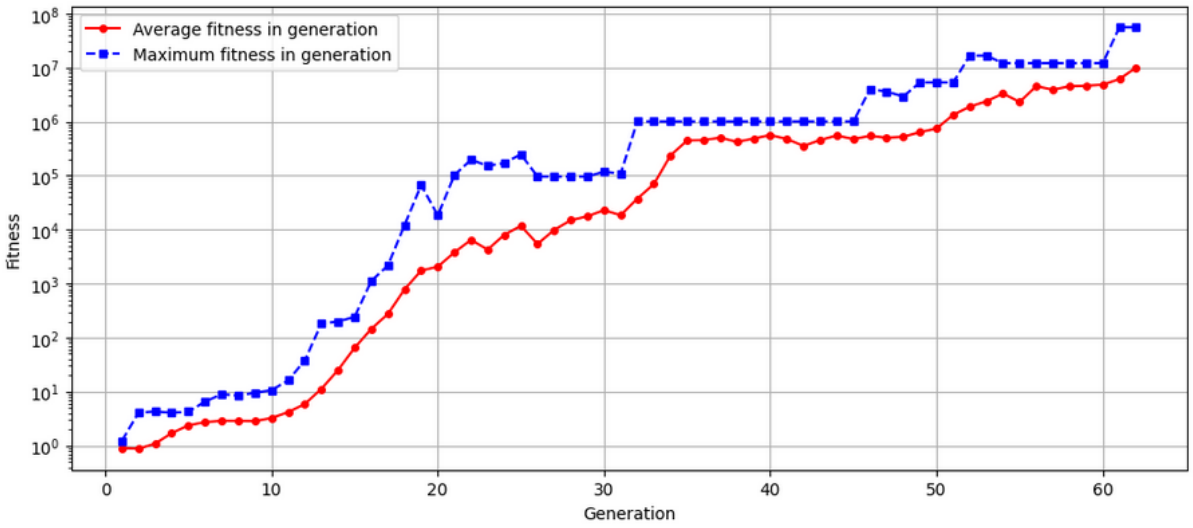

Figure 15: Plot of solution fitness across generations in the initial search. Constraints ignored. The Fitness scale is logarithmic.

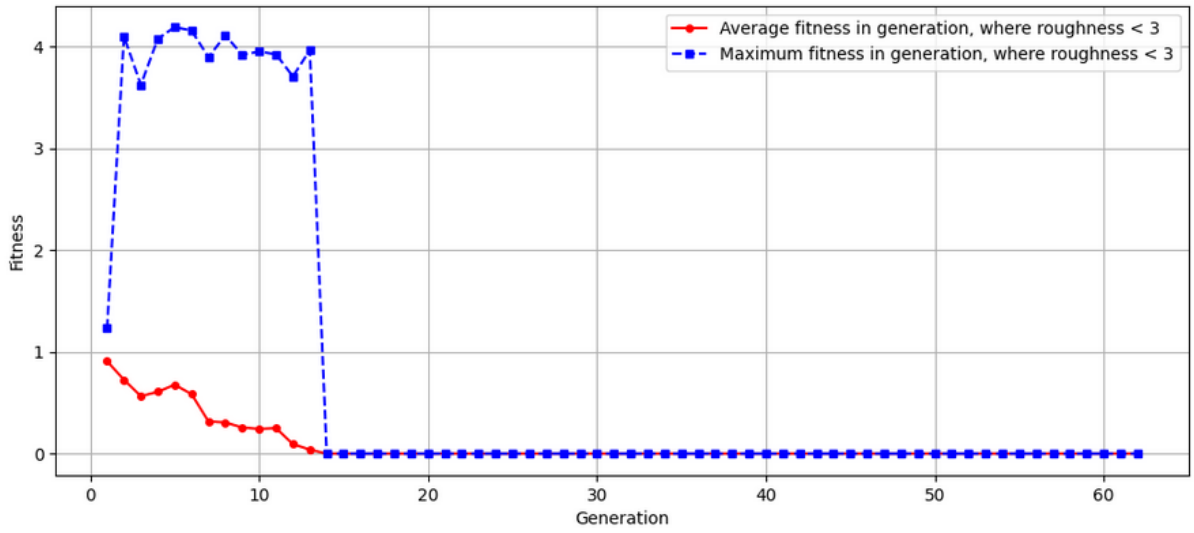

Figure 16: Plot of solution fitness across generations in the initial search. Constraints in place. Fitness=0 indicates that no solutions that met the constraints appeared in the generation.

Parameter sweeps across mutation probability and objective function

In GA searches, parameter sweep experiments are commonly used as a way of finding the best set of configurations (parameters) for the search (Samples et al. 2005). Parameter sweep is a common name for conducting multiple runs of the same experiment or algorithm with different parameter values. This allows the researcher to systematically choose parameters for their given problem without any *a priori* information.

The same approach was used in this aerodynamic search. Because of the numerical complexity of the aerodynamic problem and the resulting computing times, the parameter sweep was conducted in multiple batches over 2D grids of parameters. The first parameter sweep was processed for varying mutation probability and positive component of the objective function – namely, the denominator function B which regulates the impact of the ratio of *shadow* between NACA0012 and the solution.

Parameter	Values in 1st parameter sweep
Mutation probability p	$\{0.01, 0.05, 0.1, 0.5\}$
Denominator function $B(x)$	$\{x^2, x^3, x^4\}$
Numerator function $A(x)$	x
Mutation scale factor θ range	$\langle -0.01, 0.01 \rangle$
Generation size	1000
Number of generations	21 (including 1st gen)

Table 3: Parameter values in the first parameter sweep.

The second parameter sweep was conducted over the penalty function. In the previous searches, the penalty was shown not to be harsh enough and the unconstrained solutions eventually disappeared from the reproduction pool. The penalty function was hence then modified from a static to the dynamic function proposed by Joines and Houck (Yu, Gen 2010, p.145):

$$(c \times gen)^{\alpha} \times b(profile)^{\beta}$$

, where gen is the generation number, $b(profile)$ is the previously used static function, and c, α, and β are parameters. The sweep was over the latter 3.

Parameter	Values in 2nd parameter sweep
Mutation probability p	0.1
Denominator function $B(x)$	x^2
Numerator function $A(x)$	x
Mutation scale factor θ range	$\langle -0.01, 0.01 \rangle$
Generation size	1000
Number of generations	21 (including 1st gen)
Inner penalty function $b(profile)$	$\begin{cases} (roughness(profile) - 2)^2 \text{ if } roughness(profile) \geq 2 \\ roughness(profile) \text{ if } roughness(profile) < 2 \end{cases}$
Parameter c	$\{0.25, 0.5, 0.75\}$
Parameter α	$\{1, 2\}$
Parameter β	$\{1, 2\}$

Table 4: Parameter values in the second parameter sweep.

1st parameter sweep results

It is clearly visible that for all runs, solutions not meeting the constraints vastly overperform the constrained ones. However, simultaneous and steady increase of fitness through generations can be observed for both constrained and unconstrained solutions. For the constrained solutions, finding of which is the purpose of the search, 2 critical observations can be made:

1. Regardless of the shape of the maximum fitness per generation curve, the average fitness steadily increased for 11 out of 12 cases – this showcased the algorithm's tendency to improve solution quality per generations, which was the indented behaviour
2. The unconstrained solutions were eventually completely eliminated from the solution pool before the computation threshold of 20 generations was met. The fitness disparity between constrained and unconstrained solutions increased, decreasing the probability of selecting the latter for reproduction. This was often caused by the appearance of an extremely fit individual solution in the unconstrained pool which singlehandedly dominated the roulette wheel and caused extinction of the constrained solutions.

Unique behaviour in the sweep was noted for runs where $B(x) = x^3$. For both $p = 0.05$ and $p = 0.5$, there occurred high spikes in maximum fitness of the constrained solutions. It was discovered that for $p = 0.5$, the spike corresponded to emergence of non-physical solutions – a profile crossed over itself at its trailing edge. The solution related to the spike for $p = 0.05$ was physically possible – it was a single spike in the profile's shape at the stagnation point.

Following the immediate post-spike dip, the profile's curve was standard and hence the solution had roughness of just 2 while obtaining lift coefficient of 11.53.

For $B(x) = x^3$ and $p = 0.01$, in turn, the shape of the fitness curves was similar to most runs but the fitness values themselves were orders of magnitudes higher than expected – in thousands instead of ones. This was caused by non-physical solutions. In this case, the trailing edge crossover propagated from generation 4 onwards. Additionally, from generation 10 onwards, dimple-shaped anomalies in the lower curve of the profiles started occurring – these should increase the solution's *roughness* and make it not meet the constraints but the *roughness* definition controls only for the upper curve's extrema, assuming their similarity to the lower curve. That was evidently not the case in the run. It is possible that other non-physical solutions occurred in all 12 cases, but it was impossible to check for them manually. Only outliers visible on the graphs were examined.

Lastly, when discussing the domination of unconstrained solutions and the dying off of the constrained ones, it must be made clear that the definition of the crossover operator will cause increase in the solutions' *roughness* through the act of combining different profiles that do not necessarily align cleanly.

Denominator function $B(x) = x^3$ was rejected because of its tendency to generate the aforementioned anomalies. For both $B(x) = x^2$ and $B(x) = x^4$, the 2 mutation probability values that showcased the best fitness development with generations (maximum values and monotonicity for both maximum and average fitness) were $p = 0.1$ and $p = 0.5$.

Mutation probability $p = 0.1$ was chosen for further parameter sweeps, as for this value the constrained solutions went extinct later than for $p = 0.5$

Denominator function $B(x) = x^2$ was chosen for further parameter sweeps, as the relative increases in maximum fitness for constrained solutions in were lower in each generation. This indicates more predictable behaviour and potentially provides further proofing against non-physical solutions.

It was checked that the run combining the 2 chosen values generated physical solutions that actually looked similar to real engineering designs.

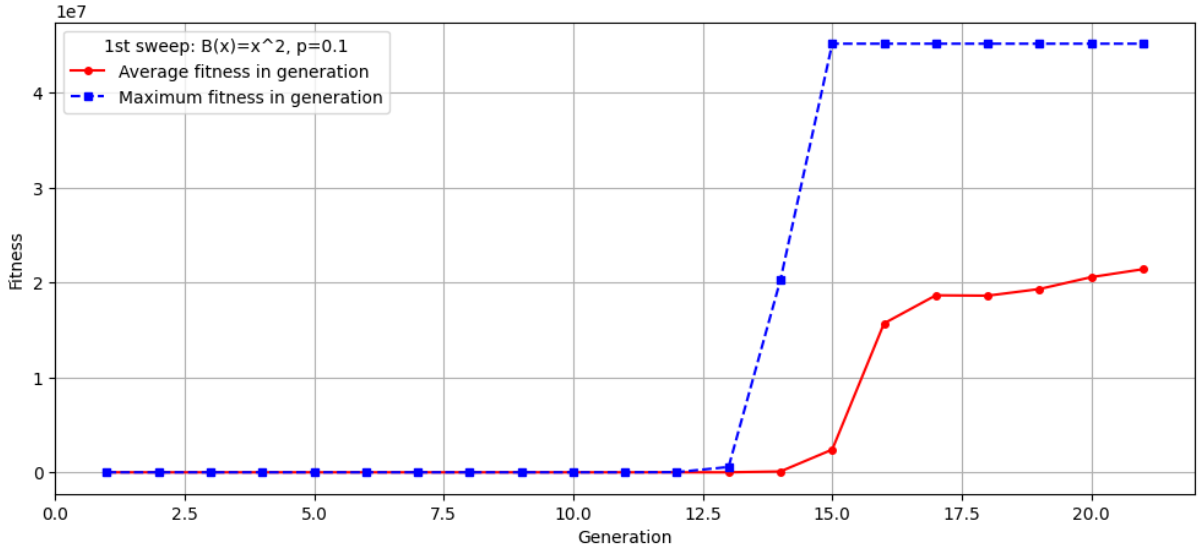

Figure 17: Plot of solutions' fitness for the 1st sweep parameter values chosen for further iterations. The fitness blew up around 13th generation.

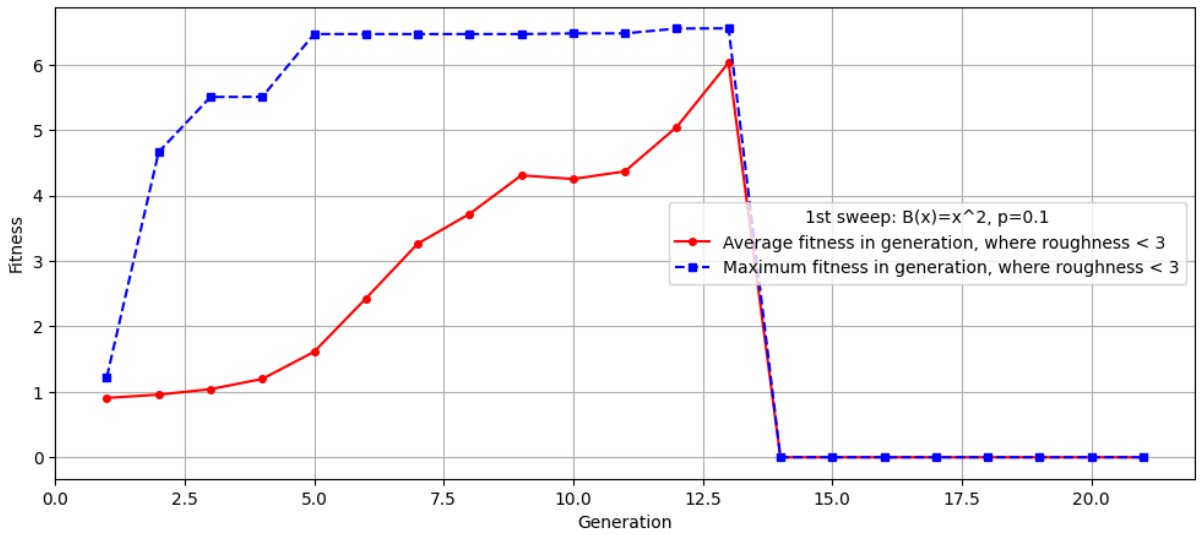

Figure 18: Plot of constrained solutions' fitness for the 1st sweep parameter values chosen for further iterations. Constrained solutions' extinction coincides with the blow up of unconstrained solutions' fitness.

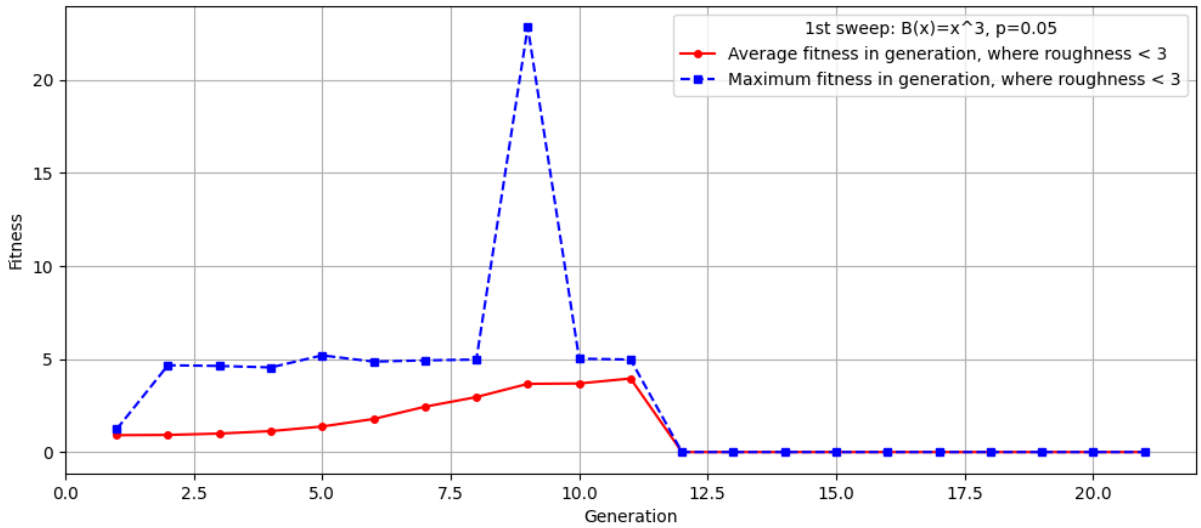

Figure 19: Plot of constrained solutions' fitness for 1 of the 1st sweep parameter values that caused physically unfeasable singularities. The singularity is visible as a spike in the plot.

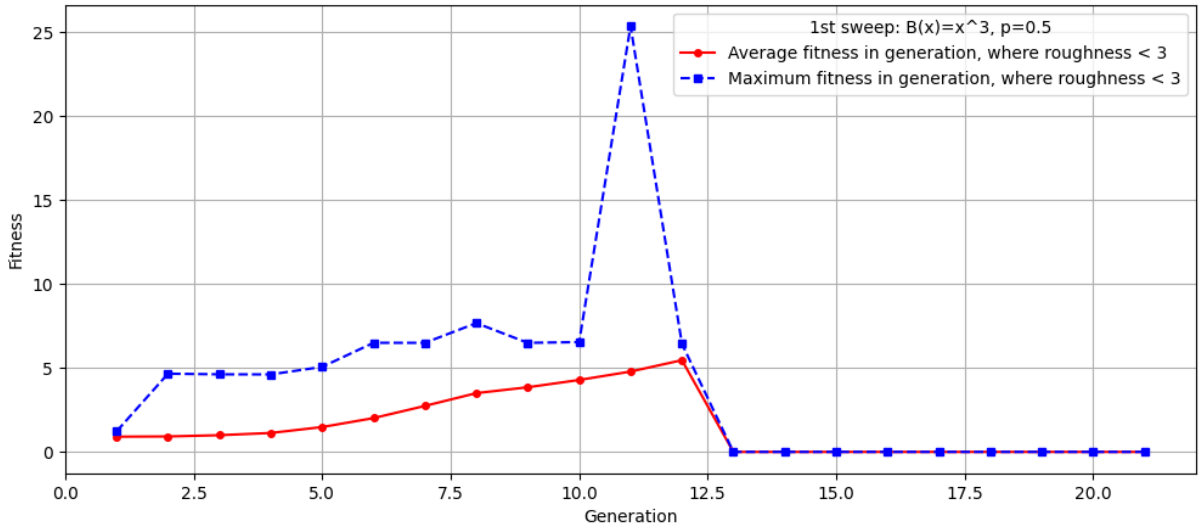

Figure 20: Plot of constrained solutions' fitness for 1 of the 1st sweep parameter values that caused physically unfeasable singularities. The singularity is visible as a spike in the plot.

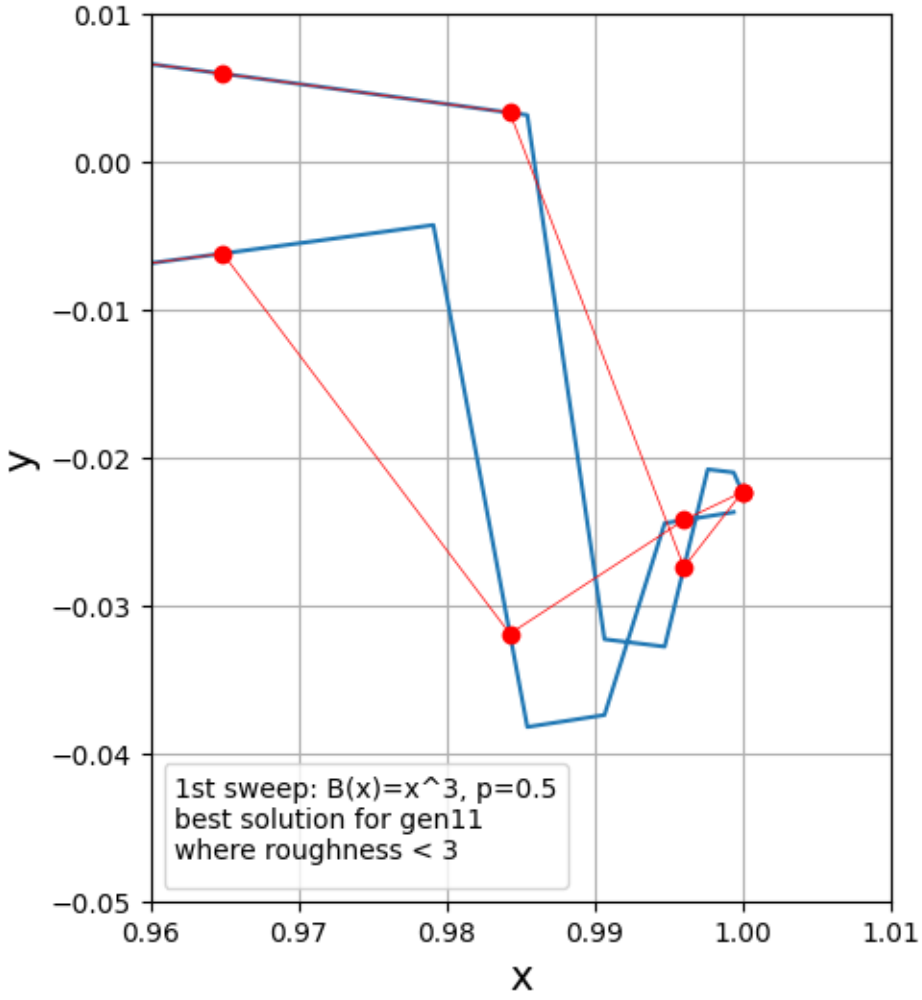

Figure 21: Zoom-in on a singularity in an aerofoil: physically impossible trailing edge crossover.

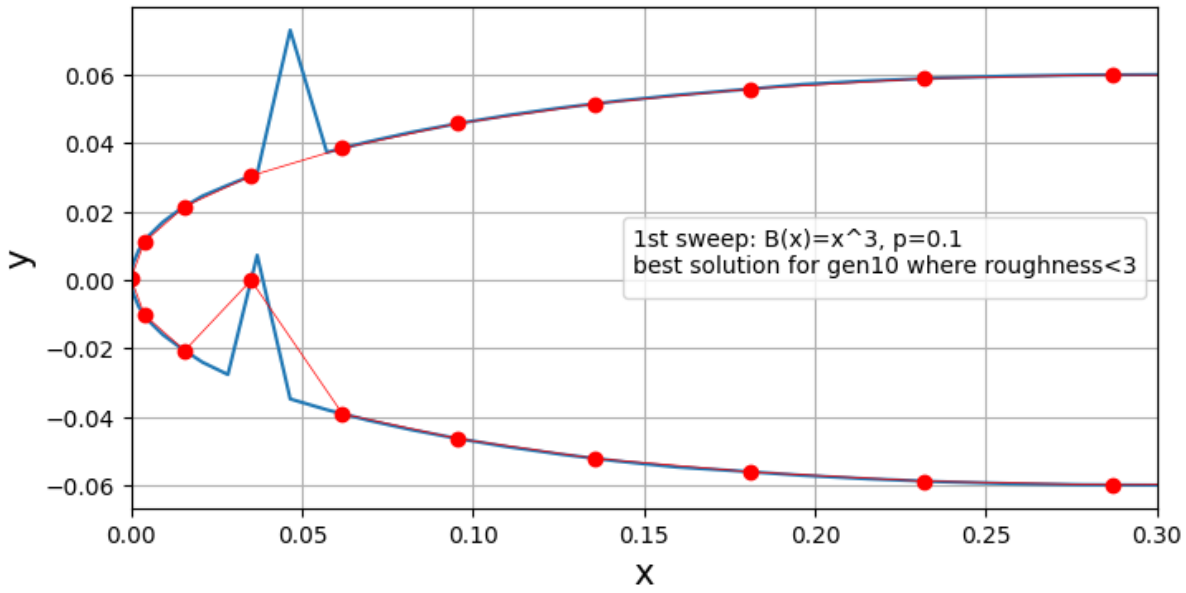

Figure 22: Zoom-in on a singularity in an aerofoil: physically unfeasable nook in the lower boundary. Notice the discrepancy between the plot of the aerofoil curve (blue) and the panels that approximated it (red).

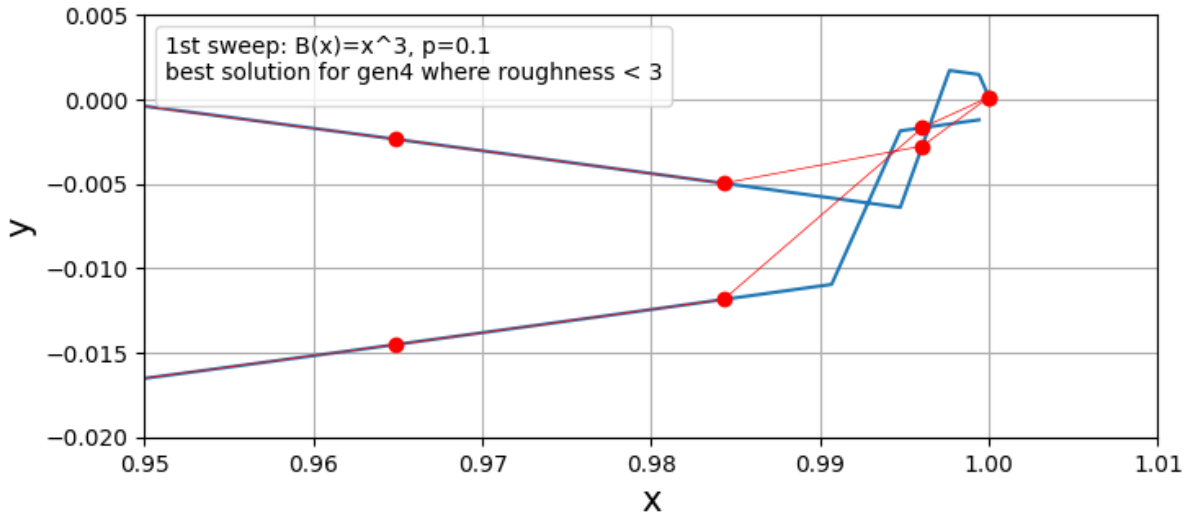

Figure 23: Zoom-in on a singularity in an aerofoil: physically impossible trailing edge crossover.

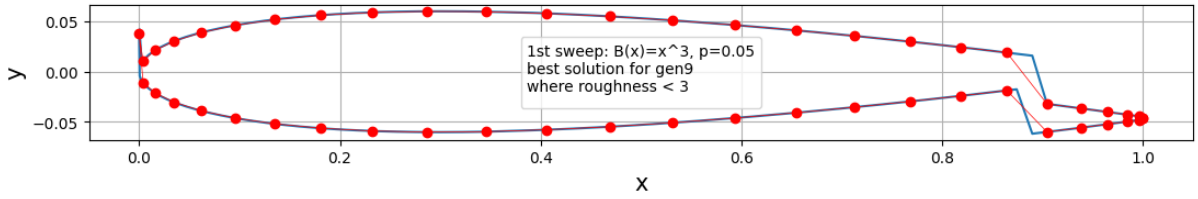

Figure 24: Best unconstrained solution for the run with 1st sweep parameter values chosen for further iterations. Notice the spike at the leading edge: it is physically unfeasible.

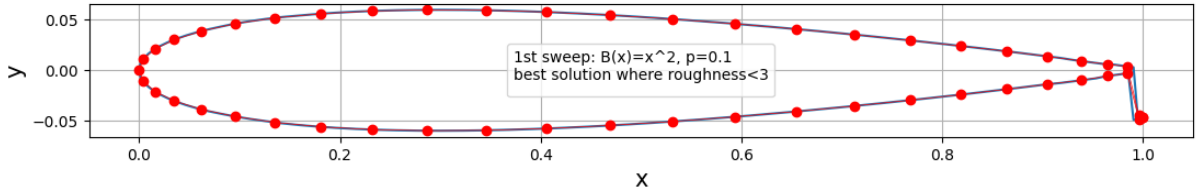

Figure 25: Best solution for the run with 1st sweep parameter values chosen for further iterations that met the constraints. Physically feasible.

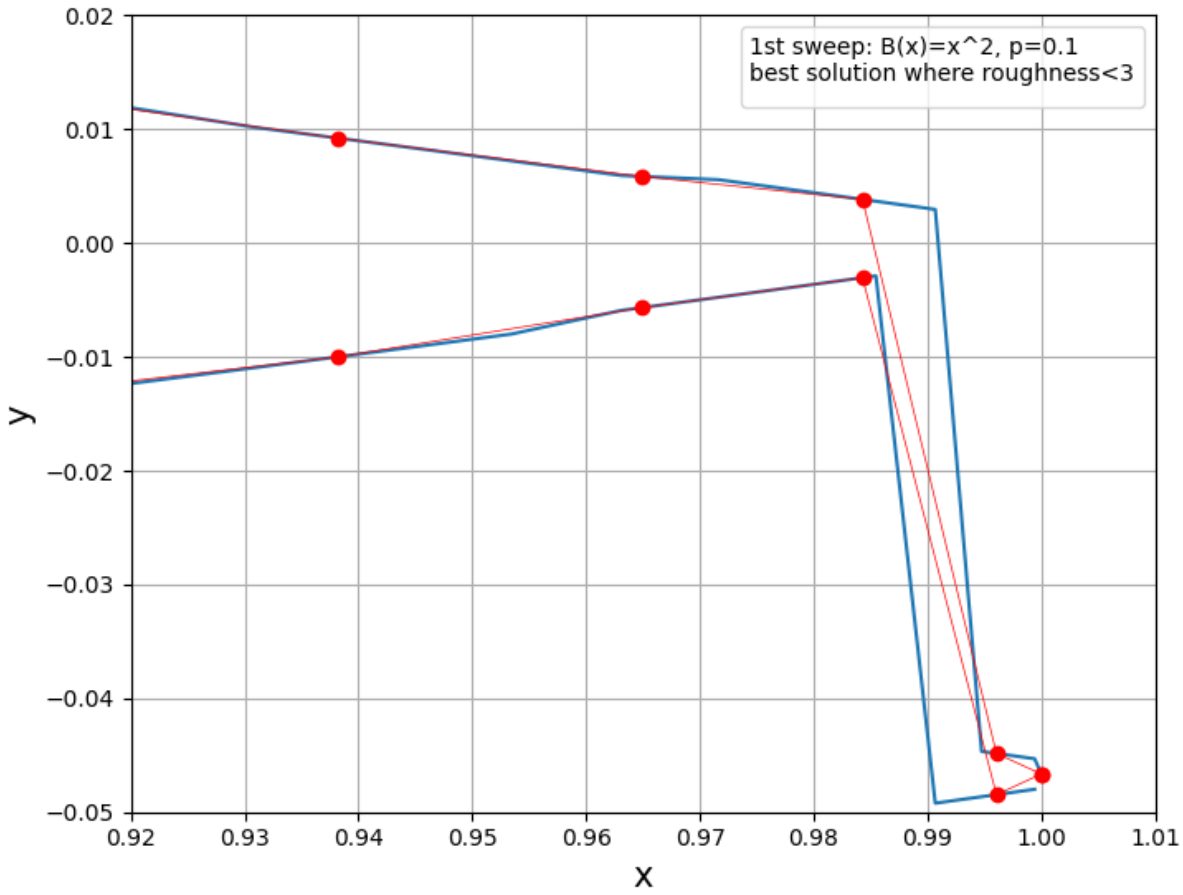

Figure 26: Zoom-in on the trailing edge of the aerofoil in Figure 25. The geometry resembles a so-called Gurney flap which has practical uses in aviation.

2nd parameter sweep results

Out of the 12 computational runs conducted in the 2nd parameter sweeps, 5 had the desired property of the fitness for unconstrained solutions diverging from the constrained solutions' fitness. However, out of them, none was actually physically proper. All had benefited from features that would discredit them in the real world, where the edge cases of the simplified steady potential flow model are unfit.

Nevertheless, tendencies were observed. Generally, solutions for $c = 0.50$ and $c = 0.75$ behaved better than for $c = 0.25$. Additionally, out of the overperforming parameter combinations, only 1 was of $\alpha = 1$. Coincidentally, it was the single run with $c = 0.25$. This is likely because both c and α are part of the dynamic penalty function's scale factor and increasing one requires decreasing the other, as to keep that factor within the range optimal for the algorithm.

It was also noted that for 2 of the runs, the results were the same. This is not an error but a result of the penalty function's properties. Coincidentally, these 2 runs were amongst the 5 best performers.

None of the runs provided perfect results. Those where the penalty function properly constrained the solution trajectory all resulted in solutions with non-physical features that the algorithm had not been built to handle.

Ultimately, 2 sets of parameter values were chosen to be used in the search proper: $\{ c = 0.25, \alpha = 1, \beta = 2\}$ – a representant of trailing edge crossover solutions, and $\{ c = 0.50, \alpha = 2, \beta = 2\}$ – a representant of stagnation point spike solutions that also happened to have the same parameter values as the one recommended by Joines and Houck.

No.	c	α	β	Anomalies in the best solution	Additional notes
1	0.25	1	2	Trailing edge crossover	
2	0.50	2	1	Stagnation point spike	Identical to #3
3	0.50	2	2	Stagnation point spike	Identical to #2
4	0.75	2	1	Stagnation point spike	
5	0.75	2	2	Stagnation point spike	

Table 5: 2nd parameter sweep runs where the constrained solutions were not the overall best.

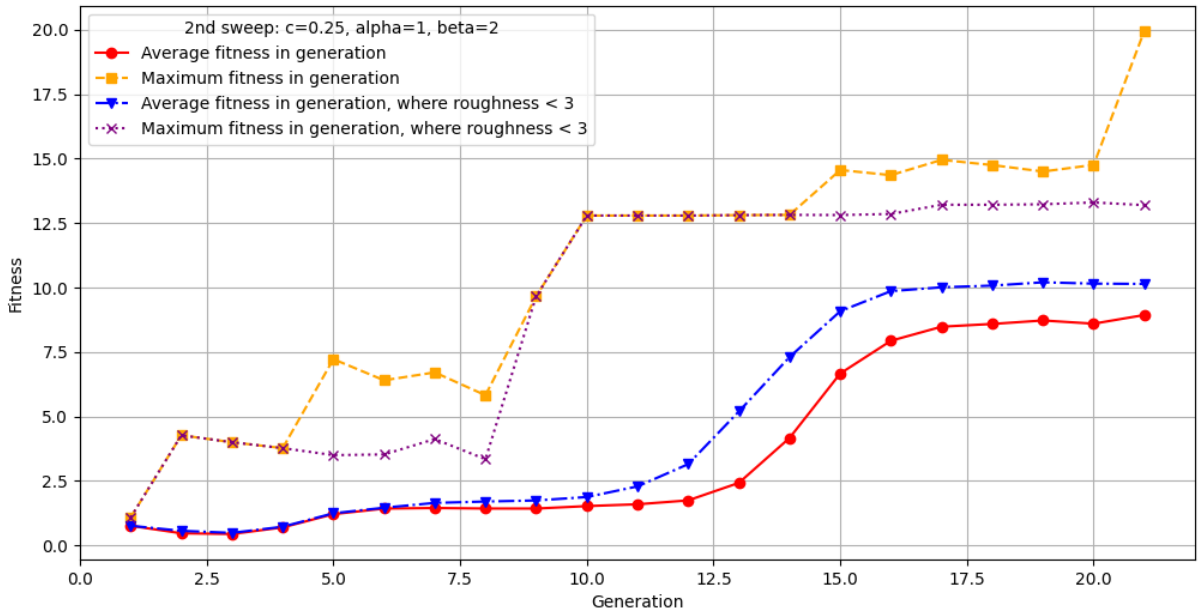

Figure 27: Plot of solutions' fitness for 1 of the parameter values chosen for the final search. Unconstrained solutions outperform constrained ones.

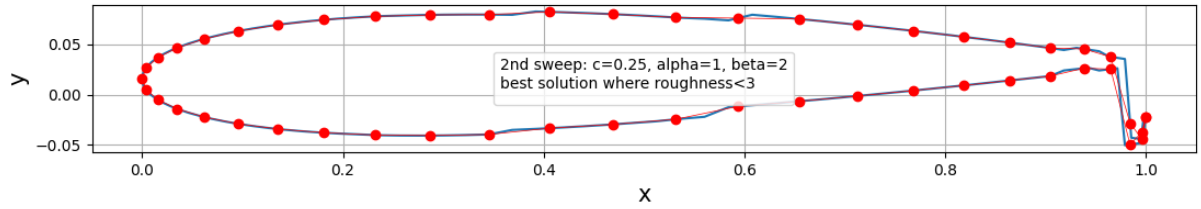

Figure 28: Best constrained solution for 1 of the parameter values chosen for the final search.

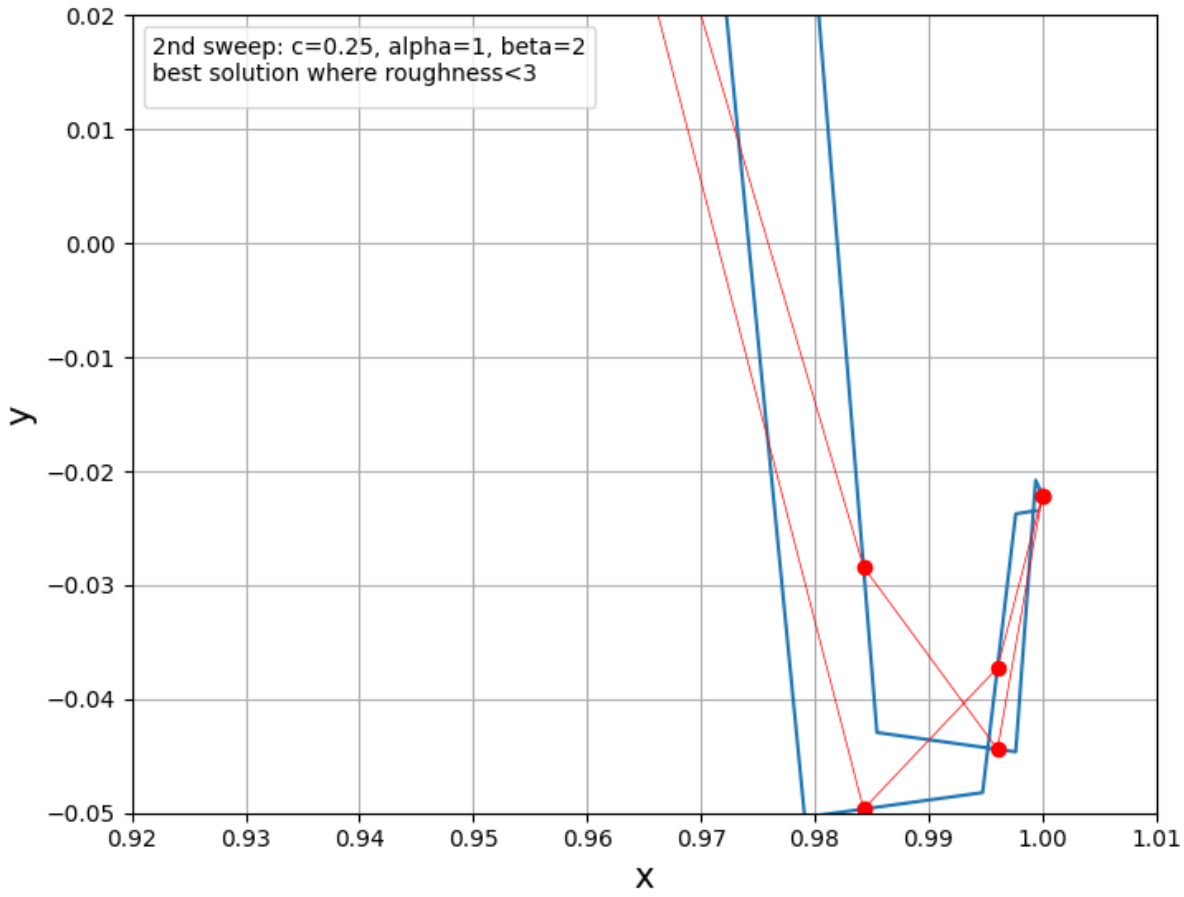

Figure 29: Zoom-in on the trailing edge of the aerofoil in Figure 27. Physically impossible crossover appears on the Gurney flap-resembling geometry.

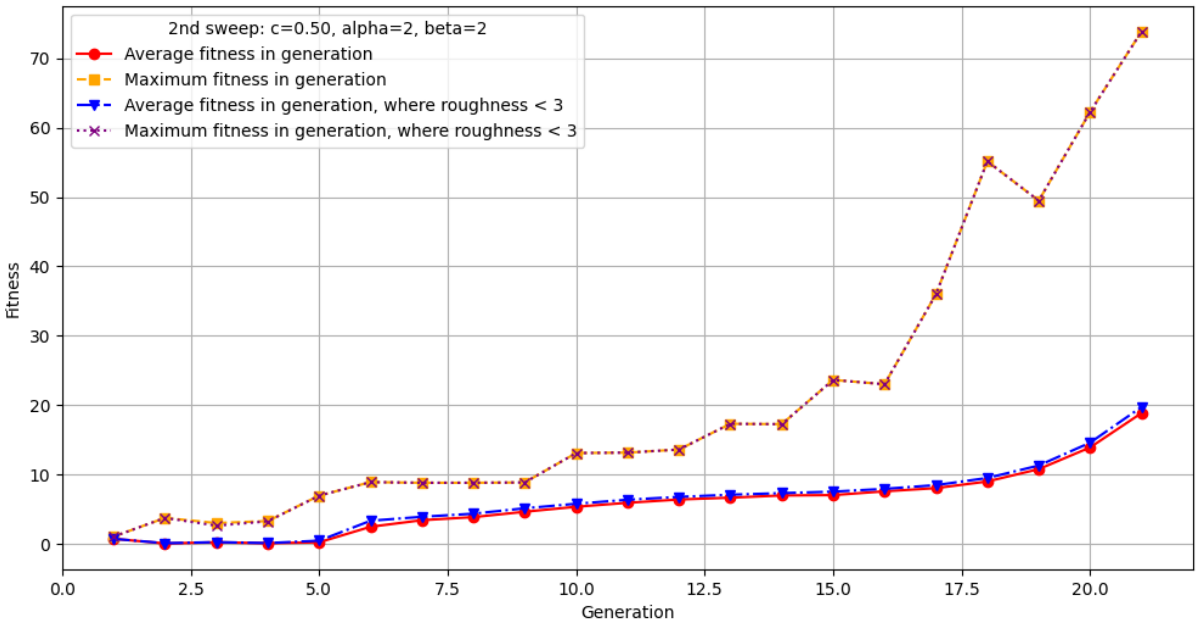

Figure 30: Plot of solutions' fitness for 1 of the parameter values chosen for the final search. Unconstrained solutions perform equally to constrained ones.

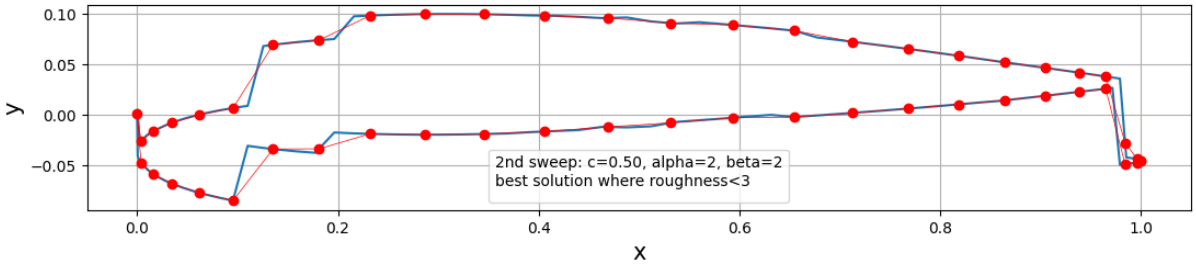

Figure 31: Best constrained solution for 1 of the parameter values chosen for the final search. The profile is not smooth. Physically unfeasible spike at the leading edge.

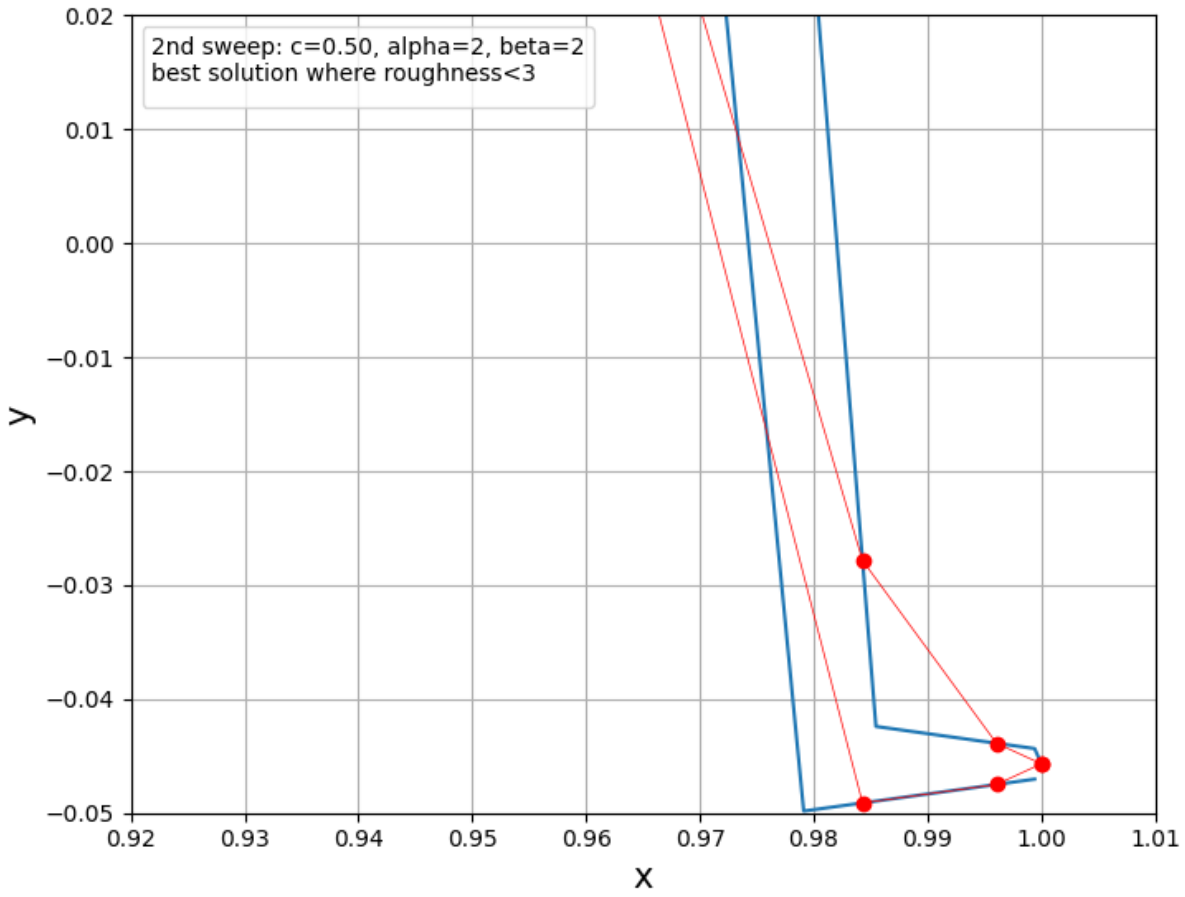

Figure 32: Zoom-in on the trailing edge of the aerofoil in Figure 30. The Gurney flap-resembling geometry has not crossed over.

Final search results

The final search for a solution was conducted over 221 generations, independently for the 2 sets of parameters chosen from the 2nd parameter sweep, labelled as case A and B. The best feasible solutions obtained for cases A and B were examined and compared.

As a reference for the parameters summary for A, B, the general form of the objective function in the final search was:

$$\begin{cases} F(profile) = max\left(\dfrac{\left(\dfrac{C_l(profile)}{C_{l_{NACA0012}}}\right)}{\left(\dfrac{shadow(profile)}{shadow_{NACA0012}}\right)^2} - (c \times gen)^{\alpha} \times b(profile)^{\beta}, \quad 0\right) \\ b(profile) = \begin{cases} (roughness(profile) - 2)^2 \text{ if } roughness(profile) \geq 2 \\ roughness(profile) \text{ if } roughness(profile) < 2 \end{cases} \end{cases}$$

Final search case	c	α	β	Constrained solutions	Fitness values	Best constrained solution's lift coefficient (rounded to whole numbers)	Flaws of the best constrained solution
A	0.25	1	2	Became extinct in the first "fitness blow-up"	Blew up around 40th and 120th generation	8	Non-physical crossover at trailing edge.
B	0.50	2	2	Identical to unconstrained	Blew up multiple times before 40th generation	1 759 821 884	Spike at leading edge. Multiple rough bends that would inhibit real flow.

Table 6: Final search runs with results.

Search case A was found to develop solutions that to some degree resemble real, physical aerofoils. While the non-physical crossover at the trailing edge from the preliminary searches persisted, and even developed further, the best solution's lift coefficient remained more grounded in reality. While 8 is very high – about 16 times higher than for NACA0012 used for reference – the profile has been unphysically crossed over itself and this likely generated extreme amounts of lift.

For search case B, there was no non-physical crossover in the best solution but it had other flaws: rough edges, and a spike at leading edge. Visually, the aerofoil that was found to be the

best solution was identical to the one found in the preliminary search. The lift coefficient for this solution was a 10 digit number – a value so high that it cannot be interpreted in any way beyond being an anomaly, a blind spot of the algorithm.

While case B was picked for the search as the solutions in it did not blow up into 6 digit fitness in the first 20 generations despite constrained and unconstrained solutions converging, the runs for A and B both included multiple sudden increases in average and maximum fitness. For case A, the first of these "blow-ups" resulted in extinction of constrained solutions, as was observed for some cases in the preliminary searches. For case B, the constrained-but-physically-unfit offspring of the best solution from the preliminary search already dominated the gene pool of the algorithm and no visual changes were observed amongst the best solutions of each generations.

Ultimately, final search case A was considered successful. In this run, solutions resembled to some extent physical aerofoils. Their unphysical features, such as not completely smooth curvature, an effect of splicing, and non-physical upper and lower curve crossovers, could be mitigated if not outright eliminated by either improving the GA operators used (introducing profile smoothing and physical bounds control during crossover) or by extending the penalty function to punish these features.

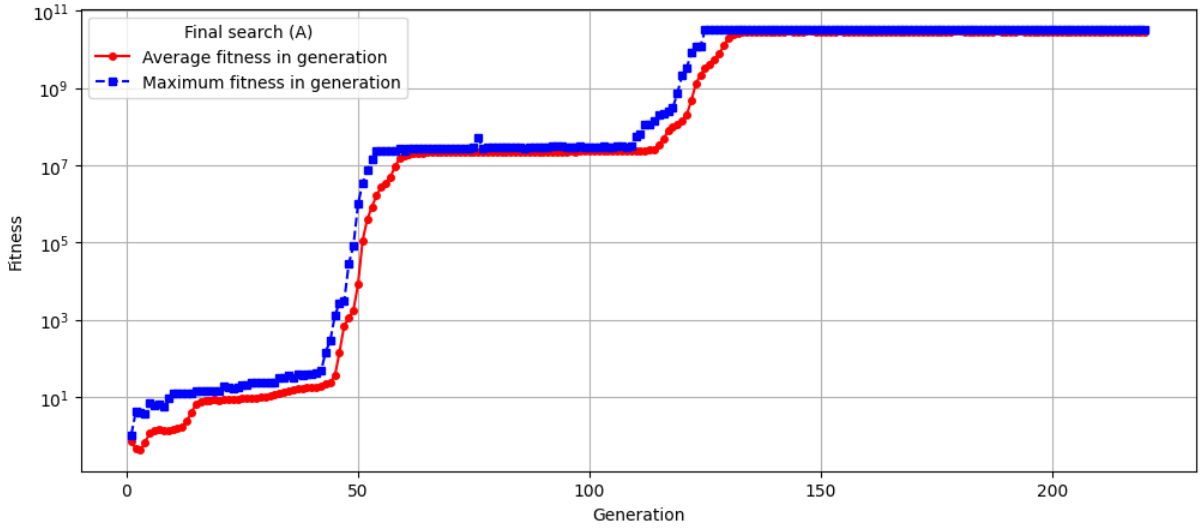

Figure 33: Plot of unconstrained solutions' fitness for case A of final search. The Fitness scale is logarithmic.

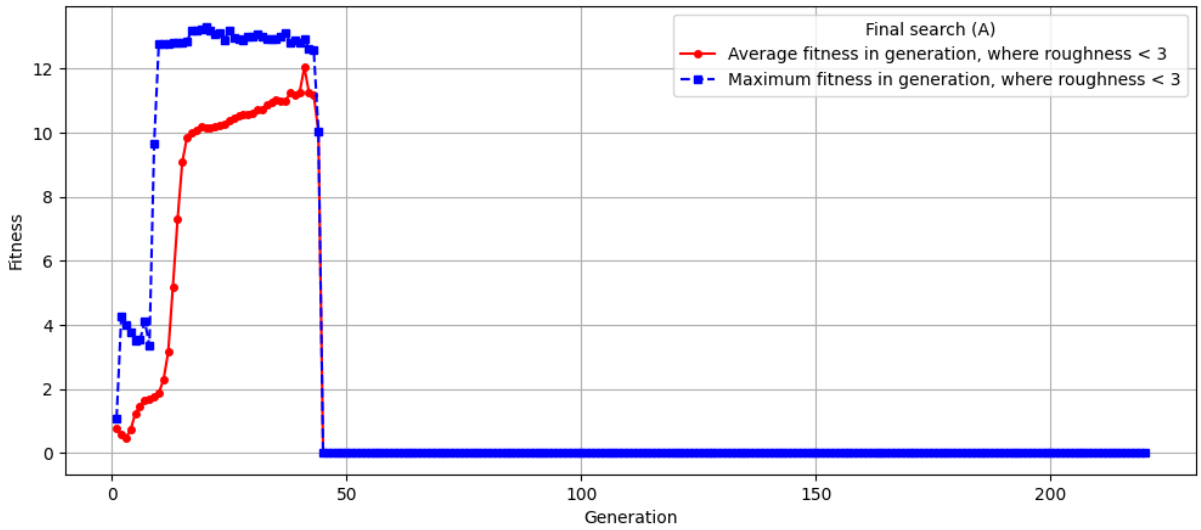

Figure 34: Plot of constrained solutions' fitness for case A of final search.

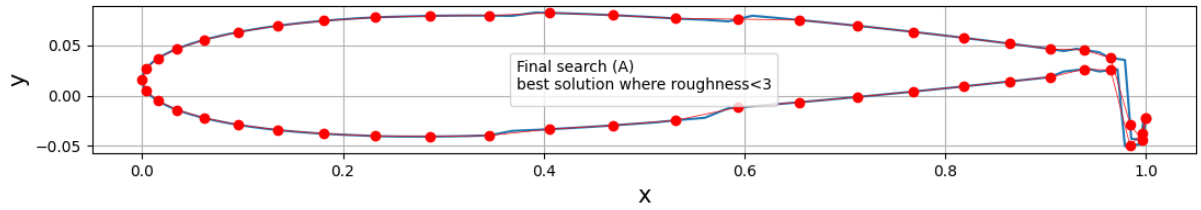

Figure 35: Best constrained solution found in case A of final search. The trailing edge crossover has progressed since the preliminary search in the 2nd parameter sweep.

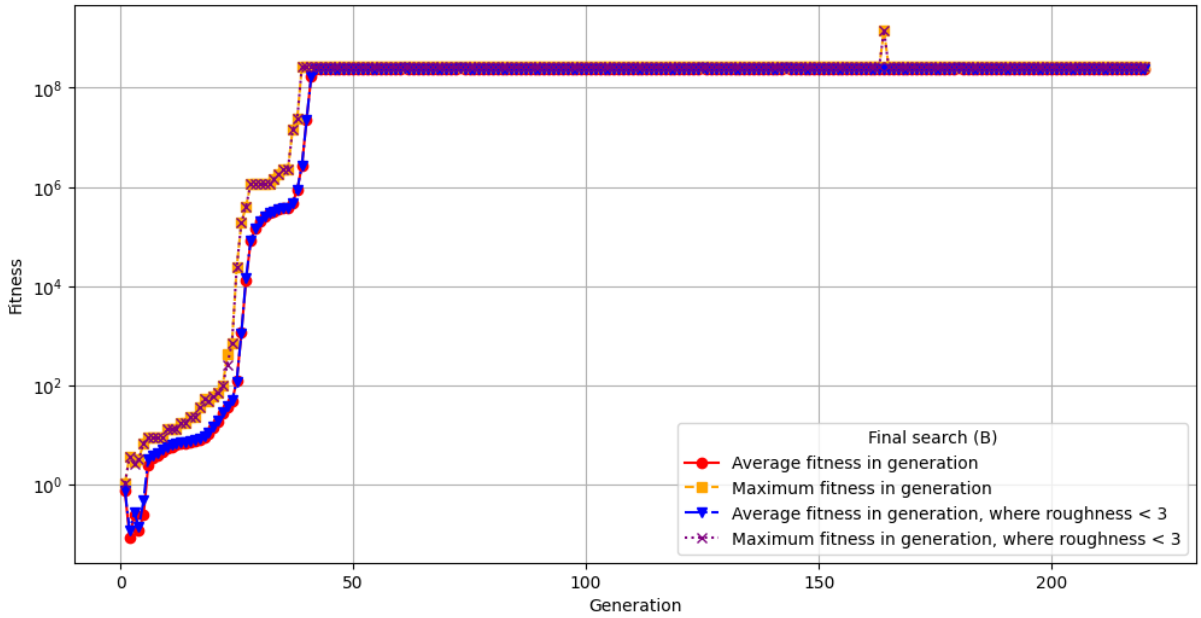

Figure 36: Plot of both constrained and unconstrained solutions' fitness for case B of final search. The Fitness scale is logarithmic.

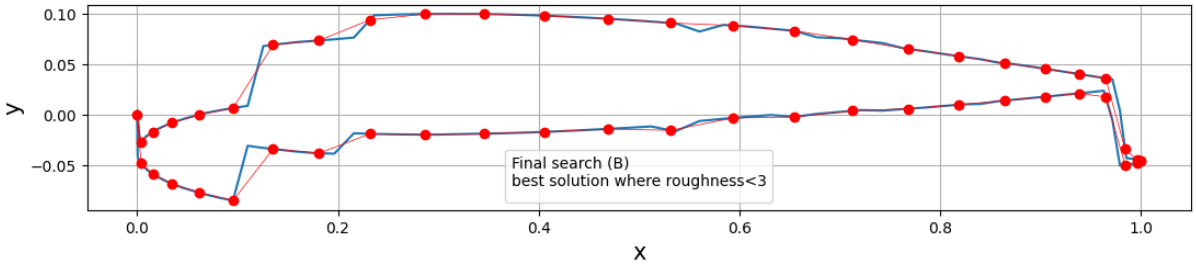

Figure 37: Best constrained solution found in case B of final search. Visually identical to the one found in preliminary search in 2nd parameter sweep.

Conclusions

It was showcased how volatile in the short term and how stable in the long term the aviation industry is. A point was made of the competitive nature of the commercial aircraft manufacturers industry and the constantly and exponentially increasing costs of airliner development.

With the assumption that some stages of aircraft development are universal and inevitable, namely the aerodynamic analysis, genetic algorithms were proposed as a tool to decrease the development costs.

A simplified computational aerodynamic model was briefly introduced and outfitted with custom-made algorithm that adapted the tools of GA into splicing aerodynamic surfaces.

The obtained results showed that the genetic algorithm's principles hold for broad aerodynamic usage – the solution population increases its overall fitness and produces more exceptional offspring with each generation.

However, even with a very simple aerodynamic model, there were difficulties obtaining physically proper solutions. Some of the issues stemmed from the simple aerodynamic model itself (favouring rough edges and singular spikes across the aerofoil's curve) and some were a result of lack of foresight (upper and lower curves of an aerofoil crossing over each other).

It is possible that physically fitter results could be obtained by improving the mutation and crossover operators, increasing the panel number to improve smoothness, or further fine-tuning the objective function to control for anomalies. Another possible improvement is moving from aerofoils represented by multitude of fixed points to curve parametrization which is often done

in design (Benim, Diederich, Pfeiffelmann 2018), and which should ensure smooth surface. Using a more advanced aerodynamic model should also improve the results.

The results obtained with the rudimentary model together with the cited works serve as examples of how meta-heuristics, and genetic algorithms in particular, can be used to decrease development time and increase quality of aerodynamic surfaces. This in turn leads to direct saving in aircraft manufacturing, which is a process where the development time and price are crucial for commercial aviation.

References

Anderson, J., 2016. *Fundamentals of Aerodynamics.* 6th red. New York: McGraw-Hill Education.

Barba, L. & Mesnard, O., 2019. Aero Python: classical aerodynamics of potential flow using Python. *Journal of Open Source Education*, 2(15)(45).

Belobaba, P., Odoni, A. & Barnhart, C., 2009. *The Global Airline Industry.* Chichester: John Wiley & Sons, Inc.

Benim, A., Diederich, M. & Pfeiffelmann, B., 2018. Aerodynamic Optimization of Airfoil Profiles for Small Horizontal Axis Wind Turbines. *Computation*, 25 April, 6(2)(34).

Berg, H. et al., 2022. *Review of Standard Passenger Weights,* Frankfurt: Lufthansa Consulting GmbH.

Cayiroglu, I. i Kilic, R., 2017. Wing Aerodynamic Optimization by Using Genetic Algoritm and Ansys. *Acta Physica Polonica A*, 132(3-II), pp. 981-985.

Clark, P., 2017. *Buying the Big Jets.* 3rd ed. s.l.:Taylor and Francis.

European Environment Agency, 2022. *European aviation environmental report 2022,* s.l.: Publications Office of the European Union.

Goldberg, D., 1989. *Genetic Algorithms in Search, Optimization, and Machine Learning.* Reading: Addison-Wesley Publishing Company, Inc.

IATA, 2024. *Global Outlook for Air Transport: Deep Change,* IATA: Montreal.

Irwin, D. & Pavcnik, N., 2004. Airbus versus Boeing revisited: international competition in the aircraft market. *Journal of International Economics*, 64(2), pp. 223-245.

Ison, S. & Budd, L., 2021. *Contemporary Issues in Air Transport.* Amsterdam: Elsevier.

Jureczko, M., Pawlak, M. & Mężyk, A., 2005. Optimisation of wind turbine blades. *Journal of Materials Processing Technology*, 167(2-3), pp. 463-471.

M.E., S. et al., 2005. *Parameter Sweeps for Exploring GP Parametrers.* s.l., s.n., pp. 1791-1792.

Rodrigue, J., 2017. *The Geography of Transport Systems.* [Online] Available at: https://transportgeography.org/contents/chapter5/air-transport/aircraft-

development-costs/

[Accessed 2016 September 2024].

White, F., 2009. *Fluid Mechanics.* 7th ed. New York: McGraw-Hill Education.

Willet, P., 1995. Genetic algorithms in molecular recognition and design. *Trends in Biotechnology*, 13(12), pp. 516-521.

Yu, X. & Gen, M., 2010. *Introduction to Evolutionary Algorithms.* London: Springer-Verlag (London) Limited.

Zhang, S. & Babovic, V., 2011. An evolutionary real options framework for the design and management of projects and systems with complex real options and exercising conditions. *Decision Support Systems*, 51(1), pp. 119-129.

List of tables

List of figures

List of attachments

Additional_figures.pdf – contains figures from the computing stage which visualize results for all analyzed runs

Codes.zip – contains all codes used in the computing stage and a requirements.txt file with packages installed in the environment used for computing