Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Automated Fibre Coupling Module

Master's Semester Thesis

Isabelle Franzen
ifranzen@student.ethz.ch

Institute for Quantum Electronics
Departement of Physics, D-PHYS
ETH Zürich

Supervisors:
Prof. Dr. Jonathan Home
Edgar Brucke

September 18, 2024

Abstract

This Master's semester project aims to optimise the coupling of a laser beam into an optical fibre using an optimisation algorithm. To this end, a genetic algorithm (GA) is implemented on a micro-controller ESP32 that orchestrates the movement of 4 servo motors, adjusting two mirrors to achieve optimal fibre coupling. Parameter optimisation experiments with the GA identify that a mutation rate of 0.1, a population size of 70 with 5 elite individuals, and a maximum of 30 generations produce a robust and effective algorithm, resulting in a module capable of replacing manual fibre coupling in experimental procedures. Challenges encountered include defining optimal stopping criteria and addressing signal fluctuations. Finally, further enhancements of the GA involving the integration of additional optimisation algorithms are highlighted.

Contents

Chapter 1

Introduction

1.1. Motivation

Coupling a laser to an optical fibre is essential for the optical setup in many experiments. However, the manual process can be tedious and inefficient. Additionally, environmental factors such as mechanical stress on the setup or other influences can degrade the coupling over time, necessitating frequent re-coupling. This inefficiency in experimental setups highlights the need for process automation. The aim of this semester project is to automate the re-coupling of the laser into the optical fibre when the laser's alignment drifts and coupling efficiency decreases.

Before delving into the specifics of optical fibres and fibre coupling tools, an overview of the tools and methods used in this semester project is provided.

To automate the re-coupling process, embedded programming was employed to implement an optimisation algorithm on an ESP32 micro-controller. This micro-controller controls four servo motors, which adjust the positions of two mirrors. The alignment of these mirrors relative to the incoming laser beam affects the efficiency of the laser coupling into the fibre. The output from the fibre is measured using a photodiode, and the analogue signal is converted to digital via an Analog-to-Digital Converter (ADC). This digital feedback signal is then used by the algorithm to optimise the coupling efficiency.

1.2. Optical fibre and fibre Coupling

In this section, a brief overview of optical fibres and the fundamentals of fibre coupling is provided.

Optical Fibre Optical fibres confine light and transmit it efficiently over long distances. They are widely used in optical setups due to their excellent ability to carry stable signals with minimal loss and low interference from external electromagnetic sources.

The fibre mode describes specific solutions to Maxwell's equations, representing stable patterns of electromagnetic fields within the fibre. Single-mode fibres, with smaller diameters, exclusively support a single Gaussian mode, namely the fundamental Transverse Electromagnetic mode (TEM00). Multi-mode fibres support multiple modes or solutions of light waves.

The basic structure of an step-index optical fibre consists of a core surrounded by cladding. The core diameter varies in size, typically around 9 μm for single-mode fibres and 50 to 62 μm for multi-mode fibres. Thus, this difference in diameter leads to fundamentally different behaviour.

The Numerical Aperture (NA) measures the fibre's ability to collect light, defining the range of angles over which the fibre can accept incoming light. The NA is influenced by the fibre's core size: a smaller core results in a wider acceptance cone, leading to a larger NA.

Furthermore, the coupling efficiency describes how effectively incoming light is transmitted into the fibre. It is determined by the overlap integral of the Gaussian mode of the laser beam and the Gaussian (fundamental) mode of the fibre. Therefore, the efficiency is high when the fibre's spot size closely matches the mode size of the light source.

Laser-to-Fibre Coupling In this project, the focus lied on coupling a free-space laser beam to an optical fibre, specifically not fibre-to-fibre coupling. The laser beam must be incident on a fibre collimator connected to an optical fibre. The lens of the fibre collimator focuses the parallel beam into the fibre, ensuring proper mode matching. [1], [2]

When coupling a laser to a fibre, six degrees of freedom are considered: The first four include the coordinates $\mathbf{x}_i$ and $\mathbf{y}_i$ of the laser beam incident on the plane of the optical fibre, and the angles of the beam relative to the perpendicular of the fibre's plane, namely $\theta_{x,i}$ and $\theta_{y,i}$. The remaining two degrees of freedom are the the focal length of the collimator, as well as the distance between the fibre collimator lens and the fibre endface. However, these two parameters are stable and cannot be easily modified: The focal length is fixed by the geometry of the collimator's lens, and since the optical fibre is mounted into the fibre collimator, this degree of freedom also remains unchanged.

Manual Fibre Coupling Manual fibre coupling is a tiresome tasks that begins with attaching a light pen to the output end of the fibre to send laser light through it. This allows both the laser and light pen to transmit through the fibre and collimator in opposite directions. In a first step, one aims for maximal overlap between the laser beam and the light pen's beam by adjusting the aforementioned four degrees of freedom. This step ensures some degree of fibre coupling.

In order to enhance the coupling, the light pen is subsequently replaced with a power meter to precisely measure the coupling efficiency. The alignment is then refined through a technique called 'Beam Walking', wherein two interrelated degrees of freedom are iteratively adjusted until an optimal signal is attained. These degrees of freedom can either be the vertical displacement of the laser beam relative to the perpendicular of the optical fibre and the laser's angle relative to it, or the horizontal displacement and angle.

1.3. Electronics

This thesis project is multifaceted as it integrates optical, electrical, and software components into a singular system. The primary motivation behind this module is its flexible integration into various experimental setups. Therefore, the software runs on embedded systems, resulting in an independent, modular system.

The electronic components of the module are responsible for managing optical signals (received by a photodiode) and controlling servo motors. Detailed descriptions of these components and their integration within the system are provided in the following sections.

1.4. Optimisation Algorithm

The objective of this semester project was to employ an optimisation algorithm to achieve optimal fibre coupling. An overview of typical and suitable optimisation algorithms for this specific problem is presented, followed by a detailed description of the genetic algorithm used in this project. In order to perform fibre coupling, the algorithm aims to find the optimal 4 mirrors' positions to maximise the photodiode signal F, which measures the optical power transmitted through the optical fibre. In this sense, one has 4 variables $\mathbf{x}$, θ_x, $\mathbf{y}$, θ_y and a fitness function F.

Grid Search Grid search is an optimisation technique that methodically evaluates the fitness of a set of coordinates corresponding to specific combinations of parameters within a predefined space. Prior to applying the algorithm, this space must be explicitly defined by specifying the lower and upper bounds of each parameter, as well as the intervals between discrete steps. As implied by its name, grid search systematically traverses this grid, computing the objective function for *each* coordinate. Ultimately, it identifies the parameter set that yields the optimal performance. The algorithm inherently converges upon evaluating all possible combinations within the grid. However, it is also possible to terminate the search early upon achieving a predefined performance target. [3] [4] [5]
This exhaustive search method can be further refined by applying more focused search techniques, such as random search or Bayesian optimisation, around the region of interest identified in the initial grid search iteration. Random search involves the random selection of coordinates and is often effective in high-dimensional spaces. [6] The more sophisticated Bayesian optimisation technique employs a probabilistic model to selectively and purposefully explore the most promising regions of the grid. [4] [5] [7]
One advantage of grid search is its straightforward implementation and result traceability. However, the algorithm is computationally expensive, as the grid size grows exponentially with each additional parameter. Consequently, this method is only appropriate for low-dimensional spaces and objective functions that require minimal computational resources.

Gradient Descent Gradient Descent is an advanced mathematical technique that iteratively finds a local minimum by calculating the gradient of the objective function. The core idea is to compute the gradient of a convex, differentiable, multi-variable function f at a point a and move in the direction opposite to the gradient in small, adjustable steps α. The vector x is then iteratively optimised according to Equation 1.1. This process produces a monotonically decreasing sequence of coordinates $f(x_n) \geq f(x_{n+1})$, converging to the optimum a.

$$x_{n+1} = x_n - \alpha_n \nabla f(x_n) \tag{1.1}$$

For gradient descent to be applicable, the objective function must be differentiable, to allow computation of gradients, and convex, to ensure that the local minimum is also the global minimum. Further, the step size, or learning rate α, is crucial. If α is too large, the algorithm may overshoot the optimum and fail to converge as seen in figure 1.1. Conversely, if α is too small, the algorithm will require many iterations and may not converge within a reasonable number of steps.

This method is extensively used in complex optimisation tasks, such as parameter optimisation in machine learning applications.

Gradient descent typically aims to find a local or global minimum. Conversely, gradient ascent is used to find a local or global maximum, by applying the same technique to $-f(x)$.

Despite the advantages of this optimisation algorithm, it was decided not to apply it due to the following reasons: Firstly, the explicit form of the fitness function F is unknown. This necessitates measuring the photodiode signal F_i for multiple configurations $\mathbf{x}_i$, $\theta_{x,i}$, $\mathbf{y}_i$, and $\theta_{y,i}$. From this dataset F_i, one could introduce small perturbations to the parameters, compute the resulting fitness, and estimate the partial derivative to approximate the gradient. While this perturbative approach to gradient descent is intriguing, it would be computationally expensive and yield only approximate results. [8] [9]

Genetic Algorithm As implied by its name, the genetic algorithm draws inspiration from biological evolution. Analogous to a species, the algorithm operates with a population comprising individuals characterised by diverse genetic configurations. Through mating, these individuals produce offspring that inherit genetic traits from their parents. Over successive generations, natural selection favours the reproduction of more successful individuals while less fit offspring are phased out. This process, driven by the principle of "survival of the fittest," facilitates the algorithm's convergence towards optimal solutions. In other words, the genetic algorithm aims to simulate evolutionary principles within a computational framework. With each generation, the algorithm evaluates and refines the current solutions. The iterative improvements are coupled with mutation to explore new spaces efficiently. As a result, it offers a robust methodology for solving complex optimisation and search problems across various domains. A more detailed description of the genetic algorithm can be found in section 2.3. [11], [12], [13]

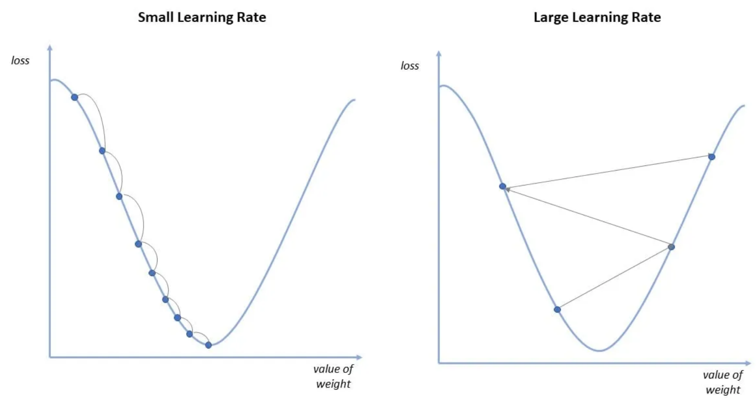


Figure 1.1.: The left plot depicts the iterative process of gradient descent to reach the local minimum. On the right image, one can observe how the large step size α prohibits the algorithm to correctly converge. [10]

Chapter 2

Methods

The project's setup required integration of optical, electrical, and software components, detailed below.

2.1. Electrical Components

The electrical components of this project included 4 servo motors controlled by a micro-controller and servo driver.

2.1.1. Servo Motors

Servo motors are motors controlled by Pulse Width Modulation (PWM), where the rotation angle of the motor's arm is determined by the duration of the electrical pulse applied. In our setup, servo motors are connected to the screws of mirror mounts. Rotating the servo motors adjusts the mirrors within the mounts, thereby controlling the fibre coupling alignment through electronic signals from the ESP32 micro-controller. For this project the servo motor 'A12 610' by KST was used. The servo is fast, robust and has a range of $\pm 50°$.

2.1.2. Electrical Setup

The micro-controller used in this setup is an 'Olimex ESP32-POE-ISO-WROVER' with Power over Ethernet (PoE) capability. It interfaces with an 'SparkFun Qwiic 12 Bit' Analog-to-Digital Converter (ADS1015) and an 'Adafruit 16-Channel 12-bit PWM/Servo Driver' (PCA9685). The Adafruit board connects to the 4 servo motors, allowing for precise control over their movements. The ADC converts the continuous analogue signal from the photodiode into a digital signal, providing feedback to the optimisation algorithm.

2.2. Optical Setup

Central to this project is the optical breadboard, on which a laser, a fibre collimator with an attached optical fibre, and two mirrors are mounted as shown in Figure 2.1. The mirrors are crucial for directing the laser beam into the optical fibre. Each mirror is mounted on a mirror mount that allows tilting along two axes, thereby controlling two of the four degrees of freedom previously discussed. The adjustment screws of the mirror mounts are coupled to the servo motors to facilitate electronic adjustment. The custom

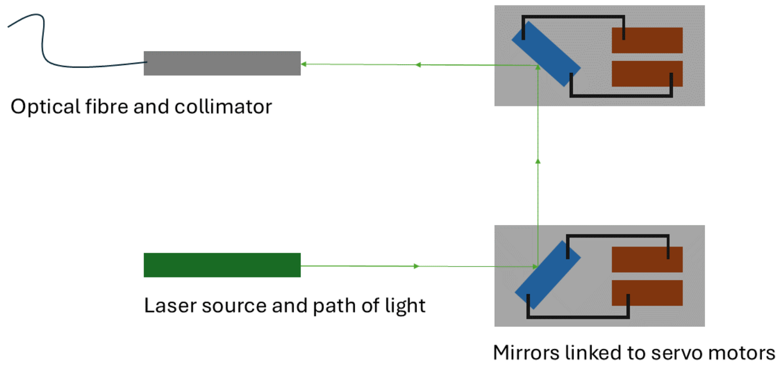


Figure 2.1.: Optical setup: The laser is directed by two mirrors into the fibre (collimator). Each mirror is mounted onto the custom module introduced in Figure 2.2 and connected to two servo motors. This connection allows the mirror to be tilted according to the movement of the servo arms controlled by the micro-controller.

modules were designed specifically for this project to securely fasten servo-controlled mirror mounts. See Figure 2.2 for details.

2.3. Genetic Algorithm

The fundamental building blocks of a genetic algorithm (GA) are the individuals that form a population, a fitness function for evaluating individuals, the parent selection method, the crossover and mutation methods, and termination conditions. The process of generating a new population based on the previous generation is depicted in figure 2.3. It involves the repetition of parent selection, (genetic) crossover, mutation, and performance evaluation. Various techniques exist for implementing each of these fundamentals. This section delves deeper into the techniques chosen for this project.

Individual The population of a generation consists of individuals, or chromosomes. For this project, each chromosome is structured as an ordered array of 4 angles and a fitness value. As previously discussed, there are 4 degrees of freedom to control in this optical setup, managed by the servo motors that adjust the mirrors to the desired angles. These four angles are translated into the array structure of each chromosome. The fitness of each chromosome is determined by the output signal of the photodiode, converted by an Analog-to-Digital Converter (ADC).

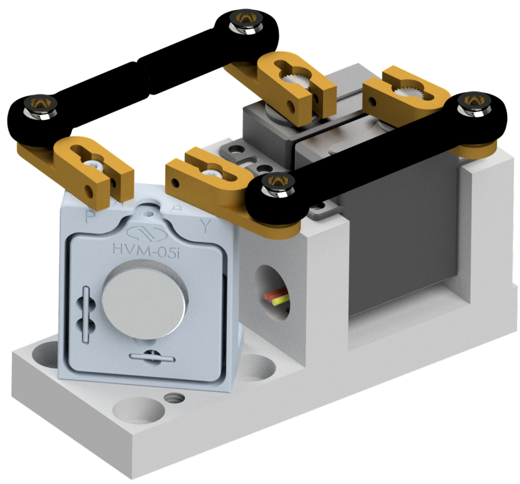


Figure 2.2.: Custom module for servo-controlled mirror mounts: The arms of the servos connect to the mirror mounts, allowing for the servos to control the positioning of the optical mirror and thus control 2 degrees of freedom. The servos are further connected the an electric control system, which is not pictured here.

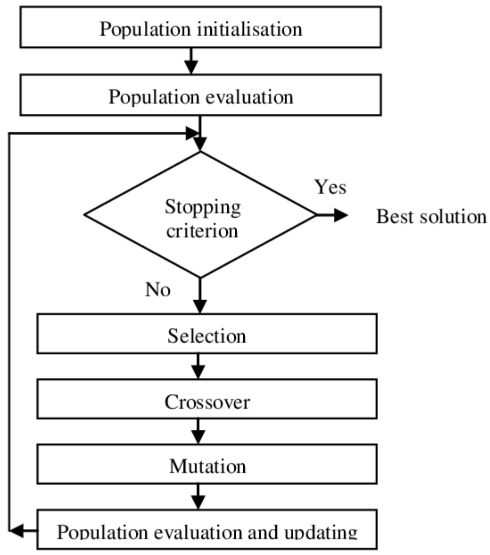


Figure 2.3.: Genetic Algorithm Process: Each individual in the initial population is evaluated based on its performance. Individuals are then selected as parents to produce offspring through mating. The offspring inherit genetic material from their parents, with potential adjustments introduced through mutation. The new generation is subsequently evaluated to determine if the stopping criteria have been met.[14]

Population Initialisation The algorithm is based on the evolution of a population, necessitating an initial population. The goal is to produce a robust algorithm that achieves high fibre coupling quality regardless of the initial population. Therefore, the population initialisation method is based solely on randomness, where 4 random angles are generated within a symmetrical range around 0 degrees of deflection for each chromosome. This is practical since initial manual coupling is performed with servo arms set at 0 degrees, enabling the algorithm to maximise coupling from this baseline.

Parent Selection To produce offspring, the genes of two parents need to be combined. The selection of parents significantly impacts the genes passed through generations, determining the algorithm's success. Various selection methods exist; here the 'Roulette Wheel' method was chosen, which favours parents in proportion to their fitness compared to the total fitness of the generation. This ensures successful individuals are chosen more frequently for reproduction. Additionally, this method can be made more aggressive or elitist by propagating some of the best-performing chromosomes unchanged to the next generation, known as 'Elitism'. Other selection methods include:

- Truncation Selection: Chromosomes are ranked based on fitness, and parents are chosen randomly from the top-performing individuals.
- Rank-Based Selection: Similar to Truncation, but selection probability is distributed according to rank rather than absolute fitness.
- Tournament Selection: A random subset of chromosomes is formed, from which the highest-performing individuals are selected as parents.
- Steady-State Selection: Most chromosomes pass to the new generation unchanged, with only a few parents selected to produce offspring.
- Random Selection: Parents are chosen randomly.

All methods share similarities but differ in the number of unchanged members passed to the new generation and the randomness factor in selecting successful parents. Balancing elitism and exploration is crucial, and this project aimed to achieve this balance by combining the Roulette Wheel method with Elitism. This approach allows fitness-based favouritism while ensuring exploration. Early generations benefit from gene pool variance to avoid early stagnation and local optima convergence.

Crossover After selecting two parents, the gene combination method can vary. This project used 'Single-point crossover', where a random point in the gene string is chosen, resulting in the offspring inheriting the first parent's genetic material up to that point and the second parent's material beyond it.
Multiple crossover points, random gene interchanges, or gene permutations could also be used. However, in this project, the 4 degrees of freedom are treated as independent, rendering permutations ineffective. Nevertheless, the vertical (or horizontal) position

and angle are linked in coupling. This suggests a potential improvement in crossover by exchanging for instance only the 'vertical' (or 'horizontal') genes between parents.

Mutation Similar to biological evolution, mutation introduces diversity by randomly altering a gene. Mutation occurs with a probability defined by the mutation rate. If a random generated number is smaller than the mutation rate, a mutation happens. The probability of mutation is thus steered by the mutation rate. The mutation rate can be updated during the algorithm, according to the current performance of the fitness history. For instance, if stagnation of sub-sequential generations is observed, the mutation rate can be increased to allow for a more diverse gene pool again. In this sense, the adjustment of mutation rates directly influences the balance between stagnation and convergence. Other mutation strategies include gene (order) shuffling or (order) swapping. Those methods couple to crossover functions however. Thus, this project opted for generating a new gene with a random angle, using a uniform distribution without favouring any specific angle within the range.

Termination The genetic algorithm is such that there is no natural definite convergence and thus termination, meaning that the stopping criteria have to be set manually, or artificially. While this offers the benefits of more control over the algorithm's behaviour, is also poses the difficult question of appropriate criteria. In this semester thesis, a myriad of stopping conditions were combined. Firstly, the algorithm ensures sufficient exploration of the parameter space by running for a minimum of 5 generations. Secondly, the algorithm stops if a chromosome achieves a predefined fitness, or if previous generations converge within an acceptable range of the target fitness, termed the 'leniency' range. Additionally, a maximum number of generations serves as a safety exit strategy.

2.4. Programming

The algorithm was implemented in C++ within the 'platform.io' environment using the Arduino framework.

Inter-Integrated Circuit protocol In 2.1, the various electrical components have been detailed, many of which necessitate specific communication protocols with the ESP32 micro-controller. The Inter-Integrated Circuit (I2C) protocol is employed for communication between the micro-controller ESP32 and both the ADC and the Servo Driver Printed Circuit Board (PCB). I2C utilises a two-wire serial communication protocol: The Serial Data Line (SDA) transmits data, while the Serial Clock Line (SCL) allows for timing control by the controller. This facilitates bidirectional communication where devices with unique addresses can be accessed by the ESP32, enabling interaction with multiple peripherals over a single bus. [15]

Physical interfacing involves connecting the ESP32's SDA and SCL pins to corresponding pins on the ADC and servo driver module. On the software side, the communication is handled by the Arduino library 'Wire'. For detailed information on this library, refer to the Wire - Arduino Reference.

ADC Signal Readout As stated above, the micro-controller communicates with the Analog-to-Digital Converter via the I2C protocol. To facilitate this communication, the 'SparkFun ADS1015 Arduino Library' abstracts communication protocols and converts raw digital data from the ADC into meaningful signal readouts. This library simplifies configuration tasks such as setting I2C addresses, selecting signal input modes (single-ended or differential), and adjusting gain amplification or voltage reference points. Further details can be found on the github repository SparkFun ADS1015 Arduino Library.

PWM control The ESP32 controls servo motors connected to the servo driver module via I2C communication. The 'Adafruit PWM Servo Driver Library' simplifies communication between the ESP32 and the servo driver, enabling precise PWM control over the servos. The source code for this library is available on the Github repository Adafruit PWM Servo Driver Library.

Chapter 3

Results

The outcome of this semester project includes the electrical and optical setups, as well as a genetic algorithm that optimises fibre coupling, starting from initial alignment.

Before discussing the performance of the algorithm, it is important to elaborate on the choice of parameters and their values.

3.1. Parameters

Range The range of the servo is limited to a deflection of $\pm 50°$ degrees from 0 degrees. However, the connection of the servo arms to the mirror mount screws reduces this range to approximately $\pm 35°$. Given that the two servos in a module are mounted in an anti-parallel configuration and their range of motion is limited on only one side, it is practical to assume a symmetric range around 0 degrees in the algorithm. These modules are designed to be flexible and interchangeable within a setup. If the range were not symmetric around the zero point, users would need to measure the motion range for each servo individually and hard-code this into the genetic algorithm.

Stopping Signal As discussed, the genetic algorithm requires artificial termination conditions. In this project, one such condition is achieving a fitness, i.e. ADC output signal, higher than a parameter *Stopping Signal*. Meeting this criterion designates the termination type of the algorithm run as *Fitness*, distinct from the other types, *Convergence* or *Max. Generation*, which will be further discussed later. It is reiterated that the objective of this algorithm is to re-couple a laser beam into fibre, after coupling has been performed but lost due to mechanical stress or drift. This implies that even though coupling is not at an optimal level, it has not been lost entirely. It is relevant, as the performance of the genetic algorithm does depend on its initial setup. For example, a laser beam that does not impinge on the fibre collimator will surely not be able to maximise coupling. This dependence poses the challenge of finding an optimal *Stopping Signal*. A suitable stopping signal must be adjusted for every setup, as a too-low value results in premature termination, while unrealistic values drive the algorithm to over-fitting and worse performance. Over-fitting in this context means the algorithm continues to optimise beyond the point where best solutions were found, causing the performance to deteriorate. Furthermore, the signal will depend on the photodiode's gain, which must be set appropriately to avoid saturating the ADC. In this project, a gain of 40 has been found to be appropriate.

The algorithm was tested with 3 different setups, for which different maximal fitness values were recorded. The parameter *Stopping Signal* was set to 1250 for all measurements, unless stated otherwise.

Leniency and Convergence In addition to the *Stopping Signal*, convergence within a lower fitness regime than the desired *Stopping Signal* was used as an alternative terminating condition. The algorithm's parameters include both the *Leniency* value, which determines the lower fitness regime, and *Convergence.* It represents the maximum fitness difference between successive generations to qualify as convergence. The *Leniency* was set to 10% of the *Stopping Signal*, though this parameter has not been tested. If the algorithm does not terminate under the *Fitness* condition but achieves generational convergence within the lenient fitness regime, its termination type is recorded as *Convergence* and *Max. Generation* otherwise.

Population Size The population size significantly influences the algorithm's outcome. Larger population sizes encourage exploration but require more generations to converge, risking over-fitting or saturation while consuming more computational resources. To find the optimal number of individuals per generation, the algorithm was tested with different population sizes between 10 and 100. For each parameter value, the algorithm was applied 10 times to compare average ending fitness, ending generation, and termination type. The results in Figure 3.1 indicate that a population size of 70 is optimal for achieving the best ending fitness, earliest algorithm termination, and minimal errors.

Mutation Rate The *Mutation Rate* supports or hinders variance in the algorithm. A *Mutation Rate* = 0.1 resulted in the best algorithm performance, as shown in Figure 3.2. This figure highlights the importance of the correct amount of exploration, realised by mutation. Too little exploration limits the search, preventing higher fitness values. Conversely, a mutation rate as high as 0.2 is too high, resulting in suboptimal fitness values and longer convergence times. Further testing with mutation rates of 0.075 or 0.125 could increase precision. Although the possibility of updating the mutation rate during the algorithm was discussed, it was not further developed due to inconclusive preliminary results.

Elitism Elitism is a method of favouring the best-performing individuals in a population by passing them unchanged to the next generation. This can be adjusted by choosing the number of elite individuals to pass on, where a parameter value of $elitism = 0$ corresponds to no elitism. In this project, different values for the parameter *elitism* were tested. It is important to balance stagnation, i.e., convergence to a premature result, and convergence at a desired fitness. A high *elitism* value may result in a faster fitness increase over generations but limits the exploration of possible angles. Based on the results presented in Figure 3.3, it is recommended to set $elitism = 5$ for a population size of 70, which is approximately 7.1% of the population size.

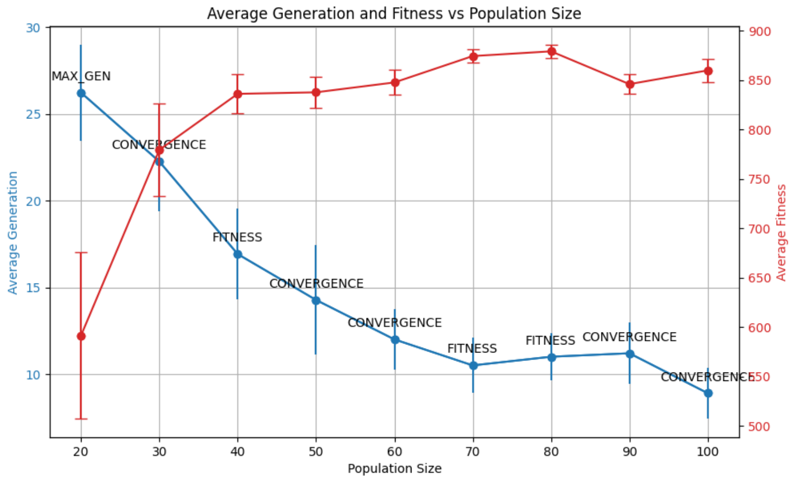


Figure 3.1.: Population Size: The average ending fitness increases with population size until approximately *Population Size* = 70 as fitness fluctuations diminish. The average terminating generation also decreases as desired, with termination types shifting from largely *Convergence* to *Fitness*. Please note that for these measurements, the *Stopping Signal* was set to 880, corresponding to the highest recorded fitness value in this setup.

Number of Generations In theory, setting a maximum number of generations is unnecessary. However, practically, it makes sense to stop the algorithm if it evidently does not reach the desired fitness value and to allow for a restart. As shown in Figure 3.1, with a population size of 70, the terminating generation is well below 15 on average. With the discussed parameter values, setting a maximum generation of 30 should guarantee a successful outcome if the algorithm performs well.

Resulting Parameter Set Based on the results above, the following parameter set is recommended for optimal performance:

- Range: $\pm 35°$
- Stopping Signal: To be adjusted per setup (Gain = 40)
- Leniency: 10% of Stopping Signal
- Population Size: 70 individuals
- Mutation Rate: 0.1
- Elitism: 5
- Maximum Generations: 30

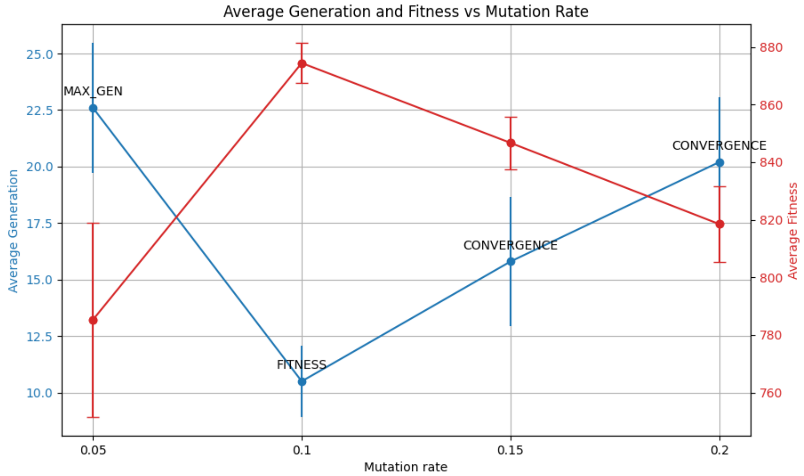


Figure 3.2.: Mutation Rate: This plot displays the effect of different mutation rates on the ending fitness and generation count. A *Mutation Rate* $= 0.1$ provides a balance between exploration and convergence, resulting in better performance and faster termination. Please note that for these measurements, the *Stopping Signal* was set to 880, corresponding to the highest recorded fitness value in this setup.

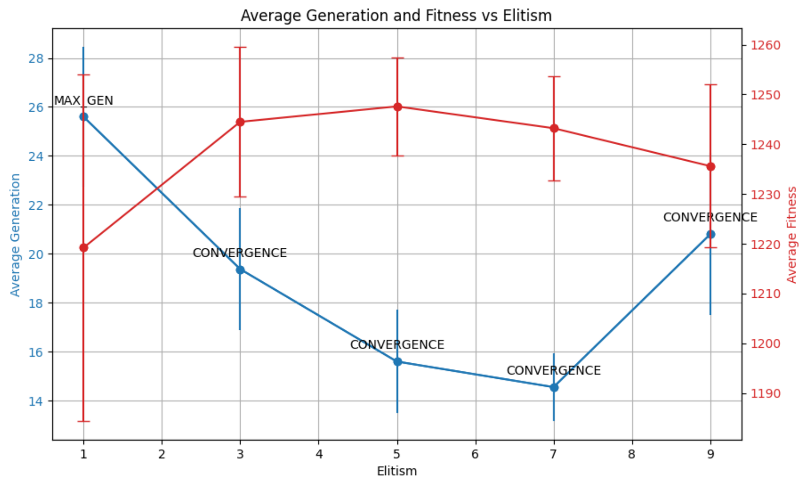


Figure 3.3.: Elitism: This plot shows the averaged ending fitness and terminating generation for *elitism* values set to $1, 3, 5, 7$, and 9. For *elitism* $= 5$, the fluctuations over multiple measurements are minor, and the terminating generation drops significantly compared to the result for *elitism* $= 3$.

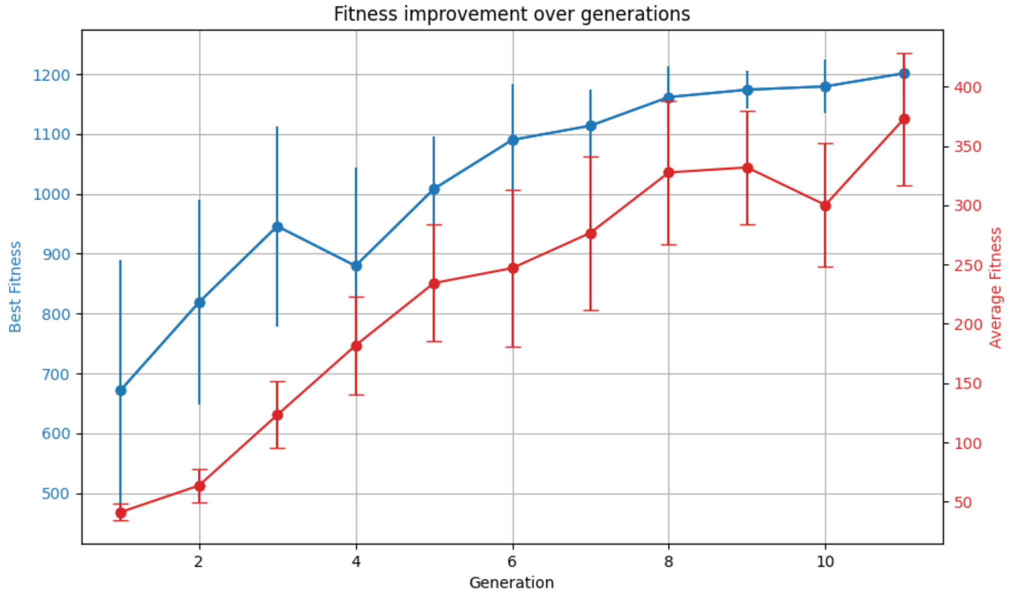


Figure 3.4.: Fitness improvements per generation: With each generation, the best fitness, as well as the average fitness increase (almost perfectly) monotonously.

3.2. Genetic Algorithm Performance

The performance of the genetic algorithm is evaluated through three distinct tests. Firstly, the improvement in the best fitness per generation is demonstrated. Secondly, the algorithm's stability across multiple measurements given an identical parameter set is assessed. Lastly, a comparison of the algorithm's performance against manual fibre coupling is conducted.

It is important to note that all measurements presented below are based on the aforementioned parameter set.

Improvement over generations In figure 3.4, it is evident how both the average fitness and the final best fitness improve progressively with each generation, indicating the algorithm successfully converges towards optimal fitness levels.

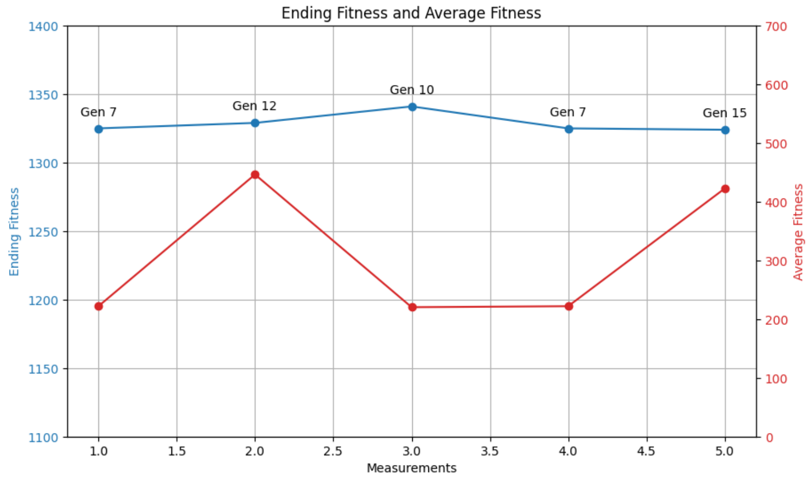
Multiple measurements To assess the algorithm's robustness, its outcomes were compared across multiple measurements of the same parameter set. Figures 3.5 present the results of two measurement sets (with differing *Stopping Signals*) of 5 measurements each. In both plots, the final fitness remains largely consistent, indicating robustness of the algorithm, especially considering the initial random population initialisation. Out of 10 measurements, only one (measurement 5 in Figure 3.5b) shows a slight deviation.

The average fitness fluctuates across multiple measurements, which is expected as the algorithms terminate at varying generations. Generally, longer algorithm runs result in higher average fitness levels, as observed in both Figure 3.5a and 3.5b.

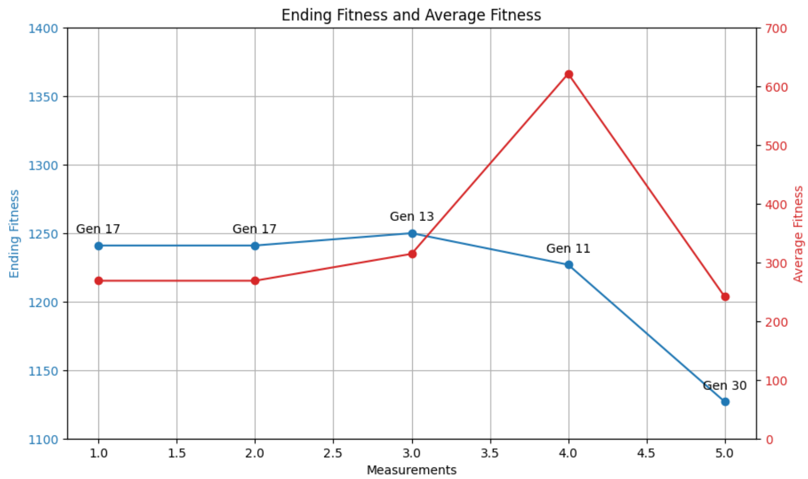
Evaluation of Algorithm Performance Against Manual Coupling The algorithm's results were measured against those of manual coupling to assess the algorithm's overall performance and usability. Manual alignment produced a maximum power of 49 µW with a corresponding ADC signal of 1450. The algorithm yielded similar results, with a measured power of 48 µW and an ADC signal of 1480. These findings demonstrate that the project successfully met its objectives.

Further, the laser beam was measured with a power meter in free space, yielding 0.930 mW. This indicates a very low coupling efficiency, which is expected. The measurements were conducted using a green laser with a wavelength of 532 nm. The laser's suboptimal performance, due to its highly non-Gaussian beam profile, accounts for the low coupling efficiency.

Fluctuations Variations in the algorithm's outcomes are expected due to the diverse angular positions generated for the servos, resulting in differing final fitness levels. However, even when the servos remain static, the output signal exhibits fluctuations, as shown in Figure 3.6. For these five measurements, the fitness was measured over approximately one minute each. Although the standard deviation per measurement is small, as seen in Figure 3.6a, an individual measurement can fluctuate considerably over the duration of approximately one minute, as depicted in Figure 3.6b.

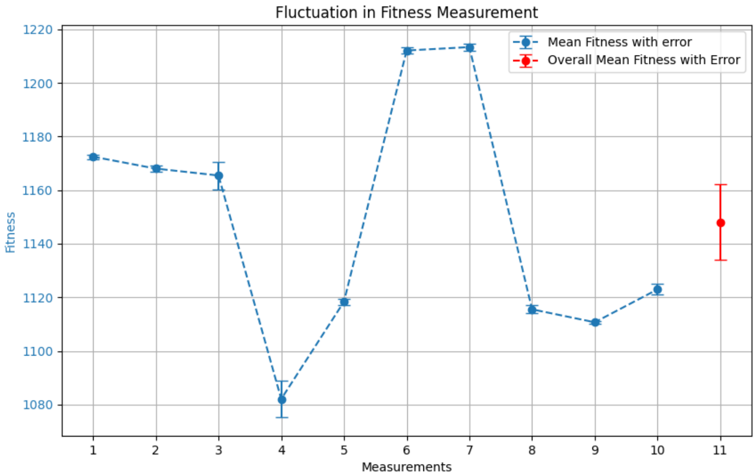


a First set of 5 measurements with *Stopping Signal* = 1350.

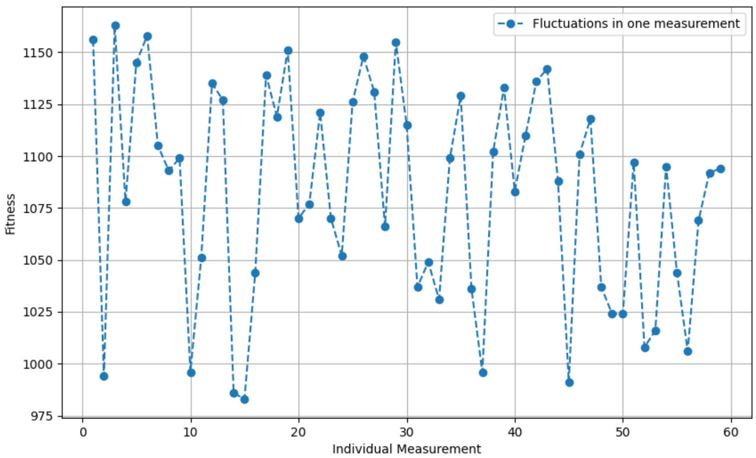


b Second set of 5 measurements with *Stopping Signal* = 1250.

Figure 3.5.: Performance comparison: Based on the same parameter set, the algorithm output is measured 5 times each. The plot shows that the ending fitness remains stable and thus proves the robustness of the algorithm. The terminating generation is indicated above each ending fitness measurement. The average fitness fluctuates more strongly than the ending fitness.



a Fluctuations over multiple measurements with mean fitness and error.



b Fluctuations over one single measurement.

Figure 3.6.: Measurement Fluctuations: Over five measurements, the fitness of an optimal solution was assessed, with each measurement conducted over one minute. In the first plot, the blue graph shows the mean recorded signals for each measurement along with their errors. The combined error for all measurements, shown in red, is ≈ 14.5. The lower figure depicts the variation in fitness over a single measurement.

Chapter 4

Discussion

4.1. Results

Based on our parameter value measurements, the algorithm operates using the parameter set 3.1. With this configuration, the algorithm consistently demonstrates incremental improvements per generation until termination, primarily driven by achieving the desired fitness level. The algorithm's robustness is evident in its ability to consistently produce the same final fitness results, as illustrated in Figure 3.5. As outlined in paragraph 3.2, the algorithm proved to perform well with respect to the manual coupling process. Therefore, this setup can confidently replace manual coupling in experimental procedures.

4.2. Challenges

Multiple components to genetic algorithm While developing the genetic algorithm, tracing unsatisfactory results proved to be challenging. This issue is inherent to genetic algorithms, as the various fundamental building blocks collectively and similarly influence the behaviour of the algorithm. For instance, attributing the lack of performance to the parent selection method instead of the crossover technique is almost impossible. This highlights the importance of coordination between the algorithm's components.

Stopping Criteria As previously mentioned, the genetic algorithm lacks inherent convergence criteria, necessitating the artificial setting of stopping conditions. In this project, criteria were employed such as the maximum number of generations, the achievement of a desired fitness level, or the observation of fitness convergence over multiple generations within a specified fitness regime. A notable challenge lies in determining the optimal *Stopping Signal*, which varies depending on setup and alignment precision. Eliminating the *Stopping Signal* entirely in favour of relying solely on fitness convergence poses challenges as well. Early convergence to local optima and defining closeness between fitness values then become critical issues, requiring a thorough understanding of optimal fitness ranges.

4.3. Improvements

Laser and measurement improvements All measurements were performed using the same laser, which exhibited a diffraction pattern instead of the desired Gaussian profile. This profile negatively impacts coupling efficiency and causes fluctuations. For future work, it is recommended to update the laser to a beam with a proper Gaussian profile.

Other critical factors in the measurements include keeping the mirrors clean and ensuring proper alignment between the fibre and the fibre collimator. Properly cleaned mirrors significantly increased the power output. Additionally, the setup is extremely sensitive, making it crucial to correctly mount the fibre into the fibre collimator. Incorrect mounting can result in an inaccurate system focal length, further affecting performance.

Fluctuations In paragraph 3.2, significant signal fluctuations in the measurements were detailed, while the computed error remained within a reasonable range. To enhance the algorithm's precision, it is proposed to calculate the fitness of an individual multiple times and continue with the mean value. This approach addresses the unreliability of single-shot measurements, resulting in more accurate and consistent outcomes. However, eliminating signal fluctuations entirely proves challenging, as the variance likely stems from vibrations of the servo motors. To mitigate this issue, a strategy known as pre-loading has been implemented to reduce noise associated with servo movements by effectively addressing backlash. Backlash occurs when the servos fail to engage their gears or linkages fully, causing them to miss the commanded movement entirely or move by an incorrect degree. To counteract this, the servos are initially directed to move in the opposite direction of the final intended angle. This amplifies the overall movement, ensuring any gaps or backlash are closed. Moreover, the mechanical setup can be improved by very securely fastening all components, thereby reducing play that would otherwise be continuously exacerbated by the frequent movement of the servo motors.

Mutation rate A promising avenue for enhancing the genetic algorithm involves dynamically adjusting the mutation rate based on algorithm performance across generations. For instance, increasing the mutation rate during early convergence phases and decreasing it during periods of significant fitness improvement could optimise exploration and exploitation phases. Another potential enhancement involves modularly adjusting mutation rates for individual candidates—lower rates for top performers and higher rates for less successful individuals. Implementing these strategies would require careful testing and analysis due to their potential impact on algorithm behaviour and outcomes.

Server interface The current server interface is basic and could benefit from further development. Future iterations might include creating a user-friendly starting page listing available endpoints, or integrating the web server with more advanced frameworks like Pydase for enhanced functionality and interactivity.

Modularity Expanding modularity beyond beam walking—where servo numbers up to four can be adjusted with minimal code modifications—could significantly enhance the project's versatility. Future developments might explore implementing multiple alignment systems concurrently, allowing optimisation of several modules simultaneously for more complex experimental setups.

Algorithm combinations In section 1.4, three optimisation algorithms are introduced: grid search, gradient descent, and the genetic algorithm. Future work could involve implementing the former two algorithms and comparing their performance and computational costs with those of the genetic algorithm. Additionally, there is potential to combine these techniques. For instance, the genetic algorithm could be used to identify a region of interest, within which grid search could be then applied.

Chapter 5

Conclusion

The optical, electrical, and software components, including the developed genetic algorithm, of this Master's semester project constitute a flexible module for laser-to-fibre re-coupling. This module can be easily integrated into an experimental setup, replacing manual coupling procedures and thereby saving valuable time.

Within the scope of this thesis, the algorithm's parameters were tested and adjusted to ensure optimal outcomes. Measurements of the genetic algorithm applied to fibre coupling demonstrated its robustness and ability to consistently achieve high-quality coupling. Finally, the genetic algorithm performs at least as well as manual coupling, validating its usability in experimental work.

Appendix A

Applying the Automated Fibre Coupling Module

This practical project culminates in a setup featuring an algorithm designed to be flexibly incorporated into experimental setups to optimise fibre coupling. Therefore, the practical application of the genetic algorithm is explained.

Usage of Genetic Algorithm The source code of the genetic algorithm can be accessed on the github repository Automated Fibre coupling.

Before applying the genetic algorithm, consideration should be given to the setup it can be applied to. In an optical configuration where the laser beam is partially coupled into the optical fibre, four servos are connected to the servo driver module. The ADC connected to the ESP32 micro-controller reads signals from a photodiode measuring the optical fibre's output.

Furthermore, the ESP32 is equipped with Ethernet connectivity, enabling it to host a web server for remote command execution. The available endpoints for these commands are outlined below. When the genetic algorithm has been engaged and has terminated successfully, the final mirror configuration will be saved to a file on the micro-controller. This configuration can then be requested, and the servos can be directed to assume these positions.

- endpoint /: Starting page
- endpoint $/get\_latest\_config$: Returns the latest configuration of servo positions saved on the ESP32.
- endpoint $/use\_latest\_config$: Sets the 4 servos to the most recently saved configuration on the ESP32.
- endpoint */optimise*: Initiates the genetic algorithm for optimising servo positions. The resulting positions are automatically saved to the ESP32 configuration.
- endpoint */status_optimisation*: Provides the current status of the optimisation process, indicating whether the algorithm is still in progress or has completed.

Beam Walking This setup provides modularity, enabling the algorithm to selectively address fewer than all four servos simultaneously. This flexibility supports Beam Walking—a aforementioned optimisation process for two interrelated degrees of freedom. During the upload of the source code to the ESP32, unnecessary servos can be commented out. This configuration directs the algorithm to utilise only the remaining two servos for optimisation, thereby facilitating Beam Walking.

Bibliography

1. Addanki, S., Amiri, I. & Yupapin, P. Review of optical fibers-introduction and applications in fiber lasers. *Results in Physics* **10,** 743–750. ISSN: 2211-3797. https://www.sciencedirect.com/science/article/pii/S2211379718314268 (2018).
2. Personick, S. Review of Fundamentals of Optical Fiber Systems. *IEEE Journal on Selected Areas in Communications* **1,** 373–380 (1983).
3. Dufour, J.-M. & Neves, J. Conceptual Econometrics Using R. *Handbook of Statistics* **41,** 3–31. https://www.sciencedirect.com/science/article/pii/S0169716119300367 (2019).
4. Bergstra, J., Bardenet, R., Bengio, Y. & Kégl, B. *Algorithms for Hyper-Parameter Optimization* in *Advances in Neural Information Processing Systems* **24** (Curran Associates, Inc., 2011). https://proceedings.neurips.cc/paper_files/paper/2011/file/86e8f7ab32cfd12577bc2619bc635690-Paper.pdf.
5. Feurer, M. & Hutter, F. Hyperparameter Optimization. *Automated Machine Learning: Methods, Systems, Challenges,* 3–33 (2019).
6. Bergstra, J. & Bengio, Y. Random Search for Hyper-Parameter Optimization. *Journal of Machine Learning Research* **13,** 281–305. https://jmlr.org/papers/volume13/bergstra12a/bergstra12a.pdf (2012).
7. Wu, J. *et al.* Hyperparameter Optimization for Machine Learning Models Based on Bayesian Optimization. *Journal of Electronic Science and Technology* **17,** 26–40. https://www.sciencedirect.com/science/article/pii/S1674862X19300047 (2019).
8. Ruder, S. *An overview of gradient descent optimization algorithms* 2017.
9. Mustapha, A., Mohamed, L. & Ali, K. *An Overview of Gradient Descent Algorithm Optimization in Machine Learning: Application in the Ophthalmology Field* in *Smart Applications and Data Analysis* (eds Hamlich, M., Bellatreche, L., Mondal, A. & Ordonez, C.) (Springer International Publishing, Cham, 2020), 349–359.
10. Saxena, P. Guide to Gradient Descent in Machine Learning. https://heartbeat.comet.ml/guide-to-gradient-descent-in-machine-learning-ab5e72909c7a (2022).
11. Mirjalili, S. in *Evolutionary Algorithms and Neural Networks: Theory and Applications* 43–55 (Springer International Publishing, Cham, 2019). https://doi.org/10.1007/978-3-319-93025-1_4.
12. Lambora, A., Gupta, K. & Chopra, K. *Genetic Algorithm - A Literature Review* in *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon)* (2019), 380–384.

13. Kumar, M., Husain, M., Upreti, N. & Gupta, D. Genetic Algorithm: Review and Application. *International Journal of Information Technology and Knowledge Management* **2,** 451–454. https://ssrn.com/abstract=3529843 (2010).

14. Ammeri, A., Hachicha, W., Chabchoub, H. & Masmoudi, F. Iterative simulation optimisation approach-based genetic algorithm for lot-sizing problem in make-to-order sector. *International Journal of Business Performance and Supply Chain Modelling* **6,** 376–394. https://www.researchgate.net/publication/266146665_Iterative_simulation_optimisation_approach-based_genetic_algorithm_for_lot-sizing_problem_in_make-to-order_sector (2014).

15. Wu, J. A Basic Guide to I2C. https://www.ti.com/lit/an/sbaa565/sbaa565.pdf?ts=1719410123691 (2022).