COMPARATIVE STUDY OF DIFFERENT OPTIMISATION TECHNIQUES IN LASER CUTTING

A thesis submitted towards partial fulfilment of the requirements for the degree of

Master of Technology in Laser Technology

Course affiliated to Faculty of Engineering and Technology, Jadavpur University

submitted by

RICK SARKAR

Examination Roll No. M4LST22004

Registration No. 154559 of 2020-21

Under the guidance of

Prof. (Dr.) Soumya Sarkar

(Thesis Supervisor)

Department of Production Engineering

Jadavpur University, Kolkata - 700032

School of Laser Science and Engineering

Faculty of Interdisciplinary Studies, Law and Management

Jadavpur University

Kolkata -700032

India

2022

JADAVPUR UNIVERSITY

FACULTY OF INTERDISCIPLINARY STUDIES, LAW AND MANAGEMENT

CERTIFICATE OF RECOMMENDATION

I HERE BY RECOMMEND THAT THE THESIS PREPARED UNDER MY SUPERVISION BY **RICK SARKAR** ENTITLED **"COMPARATIVE STUDY OF DIFFERENT OPTIMISATION TECHNIQUES IN LASER CUTTING"** BE ACCEPTED IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF TECHNOLOGY IN LASER TECHNOLOGY DURING THE ACADEMIC SESSION 2020 - 2021.

THESIS SUPERVISOR
Prof. (Dr.) Soumya Sarkar
Department of Production Engineering
Jadavpur University, Kolkata-700032

Countersigned

DIRECTOR
Dipten Misra
School of Laser Science and Engineering
Jadavpur University, Kolkata-700 032

DEAN
Faculty of Interdisciplinary Studies, Law and Management
Jadavpur University, Kolkata-700 032

JADAVPUR UNIVERSITY

FACULTY OF INTERDISCIPLINARY STUDIES, LAW AND MANAGEMENT

CERTIFICATE OF APPROVAL **

This foregoing thesis is hereby approved as a creditable study of an engineering subject carried out and presented in a manner satisfactory to warrant its acceptance as a prerequisite to the degree for which it has been submitted. It is understood that by this approval the undersigned do not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn therein but approve the thesis only for the purpose for which it has been submitted.

COMMITTEE OF FINAL EXAMINATION FOR EVALUATION OF THESIS

** Only in case the recommendation is concurred

DECLARATION OF ORIGINALITY AND COMPLIANCE OF ACADEMIC ETHICS

The author, hereby declares that this thesis contains original research work by the undersigned candidate, as part of his Master of Technology in Laser Technology studies during academic session 2021-2022.

All information in this document has been obtained and presented in accordance with academic rules and ethical conduct.

The author also declares that as required by this rules and conduct, the author has fully cited and referred all material and results that are not original to this work.

NAME: RICK SARKAR

EXAMINATION ROLL NUMBER: M4LST22004

THESIS TITLE: COMPARATIVE STUDY OF DIFFERENT OPTIMISATION TECHNIQUES IN LASER CUTTING

SIGNATURE: DATE:

ACKNOWLEDGEMENT

First and foremost, I would like to express my sincere gratitude to my supervisor ***Prof. (Dr.) Soumya Sarkar***, Emeritus Professor, Department of Production Engineering, Jadavpur University, for his invaluable guidance, whole-hearted support, and encouragement for accomplishing the present investigation.

I would also like to express my deep sense of thankfulness to ***Shri Dipten Misra*** providing me necessary atmosphere to work on.

I am extremely thankful to my seniors of School of Laser Science and Technology and Production Engineering Department for their inspiration and encouragement and helping me through the research work.

I record my acknowledgement to ***School of Laser Science and Engineering and Production Engineering Department*** for giving me the opportunity to pursue my research work.

Immemorial friendly behaviour and constant support of all my friends, seniors and juniors of School of Laser Science and Engineering are highly acknowledged.

All staff members of School of Laser Science and Technology and Production Department deserve special thanks for their help in diverse ways during the days of my stay in the department.

No words can convey my sense of gratitude for my family.

Lastly, I will remain grateful to all of them who supported me in any way during my entire time span in research activity.

RICK SARKAR

Examination Roll No. M4LST22004

Registration No. 154559 of 2020-2021

This thesis work is dedicated to:

My beloved parents,

my supervisor

and

Science & Technology

Table of Content

List of Figures

List of Tables

Abstract

Laser beam machining (LBM) is the most widely used machining process and can be applied to almost all metallic and non-metallic range of materials. In present research study, attempt has been made for optimization of laser cutting parameters of high strength engineering materials i.e., Ti-6Al-4V super alloy and A653 galvanised steel. Experimental data has been taken from the available literature [3, 16] in order to study the Analysis of variance (ANOVA). Then, the regression equations are developed between the process parameters and the machining characteristics such as, kerf deviation and material removal rate. Finally, the developed regression equations are used for optimisation of the laser cutting process parameters to obtain the minimum kerf deviation and maximum material removal rate using Response Surface Methodology (RSM) and Particle Swarm Optimisation (PSO). For Ti-6Al-4V super alloy, the optimized values found by PSO for minimum kerf deviation (KD) and maximum material removal rate (MRR) are 0.00971 mm and 192.9321 mg/min, respectively, at level of pulse width of 2 ms, pulse energy of 4.64 J, cutting speed of 10 mm/min and gas pressure of 8.17 kg/cm^2. For Ti-6Al-4V super alloy, the optimized values found by RSM for minimum kerf deviation (KD) and maximum material removal rate (MRR) are 0.011 mm and 181.063 mg/min, respectively, at level of pulse width of 1.822 ms, pulse energy of 5.020 J, cutting speed of 10.104 mm/min and gas pressure of 8.176 kg/cm^2. Similarly, the optimal results obtained using Particle Swarm optimisation algorithm and RSM for A653 galvanised steel have also been compared. The optimal results obtained using Particle Swarm optimisation algorithm are compared with the optimum results computed using RSM, and found that the PSO gives better results as it can predict global optima.

CHAPTER – 1

INTRODUCTION

A laser is a device that produces light using an optical amplification technique based on electromagnetic radiation's stimulated emission. "Light amplification by stimulated emission of radiation" is referred to as "laser" in short. Charles Hard Townes and Arthur Leonard Schawlow's theoretical research served as the foundation for Theodore H. Maiman's 1960 creation of the first laser at Hughes Research Laboratories. A laser differs from other sources of light in that it emits light that is coherent. Applications such as laser cutting can be made possible by spatial coherence, which allows a laser to be focused on a small area. Applications like laser pointers and lidar are made possible by spatial coherence, which also permits a laser beam to remain narrow over very long distances (collimation). High temporal coherence in lasers enables them to emit light with an extremely restricted spectrum. Alternately, one can use temporal coherence to create ultrashort light pulses with a broad spectrum and femtosecond durations.

1.1 Types of laser cutting:

Although laser cutting has been present since the 1960s, its increased use in industrial processes means that it is currently as significant as ever. In this non-contact method, a constant laser beam generates heat and pressure that precisely reshapes or bends a variety of materials as the cutting head passes over their surface. Depending on the laser's strength, the main component material it uses to produce the laser beam, and the material it is working against, the laser technology may do a wide range of tasks like cutting, drilling, and engraving. There are different types of laser cutting –

1.1.1 Fusion cutting – melt & blow

This technique involves creating a penetration hole from the edge and blowing the molten material out of the cut kerf with a strong gas jet. When the beam reaches the surface, most of it falls into the hole or kerf, Some of it may reflect off the unmelted surface, and some of it may simply pass straight through. If a beam is thin enough, it can move slowly through the kerf without touching the material, with the melt starting close to the beam's leading edge. There are two mechanisms for absorption at the steeply sloped edge of the cut front (roughly 14° to the vertical): primarily Fresnel absorption - by which the beam interacts directly with the material and secondly, plasma absorption and its radiation. The gas blows the plasma away, thus reducing the build-up on a cut front. As soon as the melt starts, the fast-moving

gas stream, pressure drop, and kerf's depth will cause the melt to blow away. At the kerf bottom, the melt gets thicker due to flow from above, formation of the film, and surface tension blocking the melt from leaving. The gas stream that ejects the molten droplets of the material at the base cut into the atmosphere. In blowing the molten part from the kerf, the gas generates a low-pressure region further up the cut length.

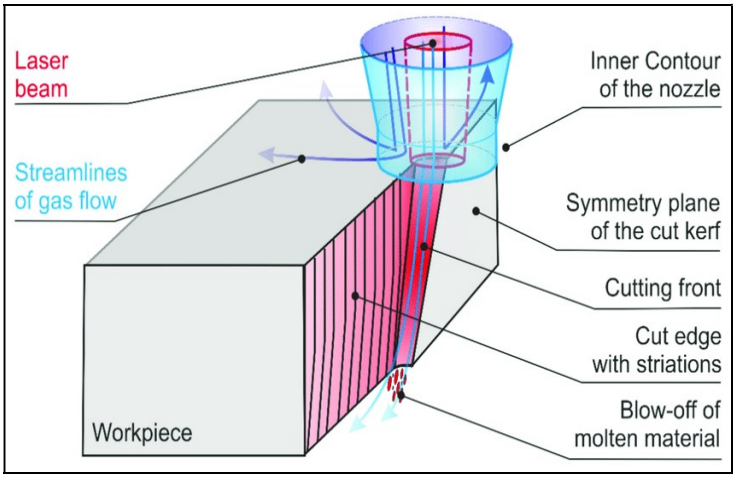

Figure 1.1: Schematic diagram of laser fusion cutting

1.1.2 Vaporization cutting

A laser beam is attached to a static portion of the metal or substance during this cutting operation. If the laser is in place, it heats the metal or substance until it starts to boil, which results in a tiny hole being formed. The metal in this hole produces vapour as it gets deeper, which erodes and destroys the adjacent walls.

Laser vaporization cutting uses a high-energy-density laser beam to heat the workpiece so that the temperature rises rapidly, reaches the boiling point of the material in a very short time, and the material begins to vaporize and form a vapor. These vapors are ejected at a high speed, and a cut is formed in the material while the vapor is ejected.

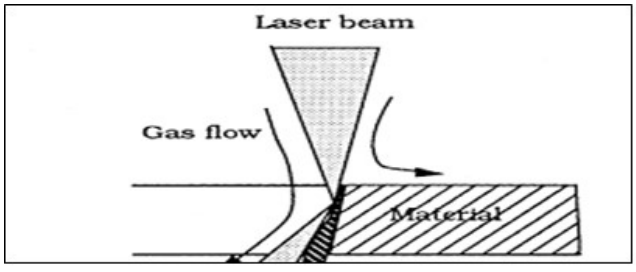

Figure 1.2: Laser vaporized cutting

1.1.3 Reactive fusion cutting

Another heat source is required when the gas and workpiece react exothermically. The cut front becomes the focus of a variety of aspects. The melt and the gas travelling through it interact, pushing the melt away from the kerf. Reactive substances are often composed of oxygen or an oxygen-containing combination. Burning reaction normally starts at the top when temperature approaches ignition temperature. Oxide is blown into the kerf as it forms and covers the melt below. The cutting speed with this method is often at least twice as quick as melt and blow cutting.

Faster cuts typically result in better cuts with less heat penetration. Nevertheless, while a cutting reaction is taking place, the workpiece can experience certain chemical alterations. The advantage of oxide dross over metal dross is that it flows easily and does not stick to the base metal as tightly as metal would.

1.2 Cutting with different types of lasers

1.2.1 ND:YAG laser

The most often used laser for laser cutting is the pulsed Nd-YAG laser. A YAG laser is a solid-state laser that uses an impurity in the host material as the active medium. Thus, the neodymium ion (Nd(+3)) is utilised as a 'dopant,' or intentionally inserted impurity in either a glass crystal, and (Nd(+3)) ions determine the 1064nm result wavelength. The lasing or lasing material With a diameter of 10 mm and a length of 150 mm, the host is formed like a cylinder. The inside and exterior of the cylinder are flattened and paralleled to extremely tight tolerances on both sides, then polished to a high shine on both sides. silvered with an optical finish to provide a reflective surface. The crystal is excited by a krypton atom. The use of a Xenon lamp aids the creation of high average power in the ND:YAG laser system. This was accomplished by combining many separately pulsed laser rods into a single resonator. Pumping and extraction efficiency are both improved as a result of this. Oscillator-amplifier systems can be utilised instead of single multirod oscillators. Nd: YAG lasers can be employed in three modes: continuous output, pulsed output, and Q switched-mode. Lasers have an average output power of 0.35 to 4.5 kW, with a peak power of up to 100 kW. One of the major advantages of the Nd: YAG laser is its ability to transport laser energy across optical fibres.

1.2.2 CO_2 laser

The CO2 laser, which is a type of gas laser that is widely used in industrial processing, is significantly more efficient than the Nd:YAG laser and produces much greater continuous-wave output powers as well as better light quality than the Nd:YAG laser. Gas mixes mostly containing nitrogen and oxygen are used in these lasers. An electric glow discharge is used to combine helium with a little amount of carbon dioxide.CO2 molecules are excited using this method. By continually circulating the gas blend via the optical cavity zone, the gas heating produced in this manner may be adjusted. CO2 lasers with CW output power may produce 3-10 KW. The rapid axial flow and transverse flow CO2 lasers, which offer the highest power outputs, are the most widely employed in the industry. CO2 lasers have several advantages, including excellent electrical efficiency, low operating costs, and the ability to scale up to high powers.

1.2.3 Fibre laser

The active medium in a fiber laser is the core of the laser. The fiber has been doped with a rare earth element. A single-mode optical fiber laser made of silica is the most common type. The pump beam travels the length of the fiber and is guided either by the core itself, as in Single-mode optical lasers, or by an inner cladding around the core. For many materials processing applications, fiber lasers have been touted as a viable substitute for solid-state and CO_2lasers.

In order to form a laser resonator with fibers, one either needs some kind of reflector to form a linear resonator, or one builds a fiber ring laser. Various types of mirrors are used in linear fiber laser resonators:

- In simple laboratory setups, ordinary dielectric mirrors can be butted to the perpendicularly cleaved fiber ends, as shown in Figure 1.3. This approach, however, is not very practical for mass fabrication and not very durable either.

- The Fresnel reflection from a bare fiber end face is often sufficient for the output coupler of a fiber laser.
- It is also possible to deposit dielectric coatings directly on fiber ends, usually with some evaporation method. Such coatings can be used to realize reflectivity's in a wide range.

- For commercial products, it is common to use fiber Bragg gratings, made either directly in the doped fiber, or in an undoped fiber which is spliced to the active fiber.

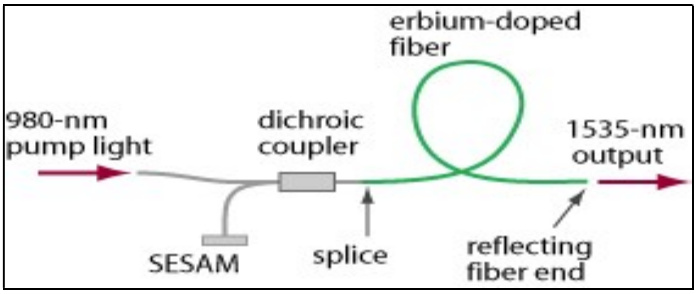

Figure 1.3: A simple erbium-doped femtosecond laser, where the Fresnel reflection from a fiber end is used for output coupling.

Most fiber lasers are pumped with one or several fiber-coupled diode lasers. Whereas the low power fiber lasers could deliver only a few milliwatts of output power, there are now high-power fiber lasers with output powers of hundreds of watts, sometimes even several kilowatts from a single fiber. This potential arises from a high surface-to-volume ratio (avoiding excessive heating) and the guiding effect, which avoids thermo-optical problems even under conditions of significant heating. Nowadays, high-power fiber lasers are widely used e.g., in laser material processing. Examples for processes are laser welding and laser cutting e.g., on metals, but also with many other industrial materials. Many applications use a fiber laser machine with continuous-wave operation; limitations concerning pulse generation e.g., with Q switching are substantial, so that bulk lasers reach clearly superior performance in such domains.

The fiber laser concept is most suitable for the realization of upconversion lasers, as these often must operate on relatively “difficult” laser transitions, requiring high pump intensities. In a fiber laser, such high pump intensities can be easily maintained over a long length, so that the gain efficiency achievable often makes it easy to operate even on low-gain transitions. In most cases, silica glass is not suitable for up conversion fiber lasers, because the up conversion scheme requires relatively long lifetimes of intermediate electronic levels, and such lifetimes are often very small in silica fibers due to the relatively large phonon energy of silica glass (→ multi-phonon transitions). Therefore, one mostly uses certain heavy-metal fluoride fibers such as ZBLAN (a fluorozirconate) with low phonon energies. The probably most popular upconversion fiber lasers are based on thulium-doped fibers for

blue light generation (Figure 1.4), praseodymium-doped lasers (possibly with ytterbium codoping) for red, orange, green or blue output, and green erbium-doped lasers.

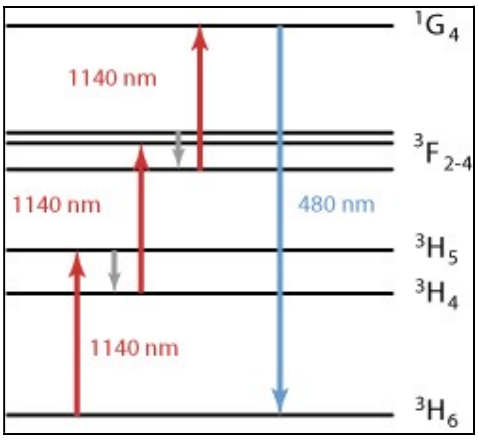

Figure 1.4: Level scheme of thulium (Tm^{3+}) ions in ZBLAN fiber

Fiber lasers can be constructed to operate on a single longitudinal mode (→ single-frequency lasers, single-mode operation) with a very narrow linewidth of a few kilohertz or even below 1 kHz. In order to achieve long-term stable single-frequency operation without excessive requirements concerning temperature stability, one usually has to keep the laser resonator relatively short (e.g., of the order of 5 cm), even though a longer resonator would in principle allow for even lower phase noise and a correspondingly smaller linewidth. The fiber ends have narrow-bandwidth fiber Bragg gratings (→ distributed Bragg reflector lasers, DBR fiber lasers), selecting a single resonator mode. Typical output powers are a few milliwatts to some tens of milliwatts, although single-frequency fiber lasers with up to roughly 1 W output power have also been demonstrated. An extreme form is the distributed-feedback laser (DFB laser), where the whole laser resonator is contained in a fiber Bragg grating with a phase shift in the middle. Here, the resonator is short, which can compromise the output power and linewidth, but single-frequency operation is very stable. Of course, further amplification to much higher power levels in a fiber amplifier is possible.

More sophisticated resonator setups are used particularly for mode-locked fiber lasers (ultrafast fiber lasers), generating picosecond or femtosecond pulses. Here, the laser resonator may contain an active modulator or saturable absorber. An artificial saturable absorber can be constructed using the effect of nonlinear polarization rotation, or a nonlinear fiber loop mirror. A nonlinear loop mirror is used e.g. in a “figure-eight laser”, as shown in Figure 1.5, where there is a main resonator on the left-hand side and a nonlinear fiber loop, which does the amplification, shaping and stabilization of a circulating ultrashort

pulse. Particularly for harmonic mode locking, additional means may be used, such as subcavities acting as optical filters.

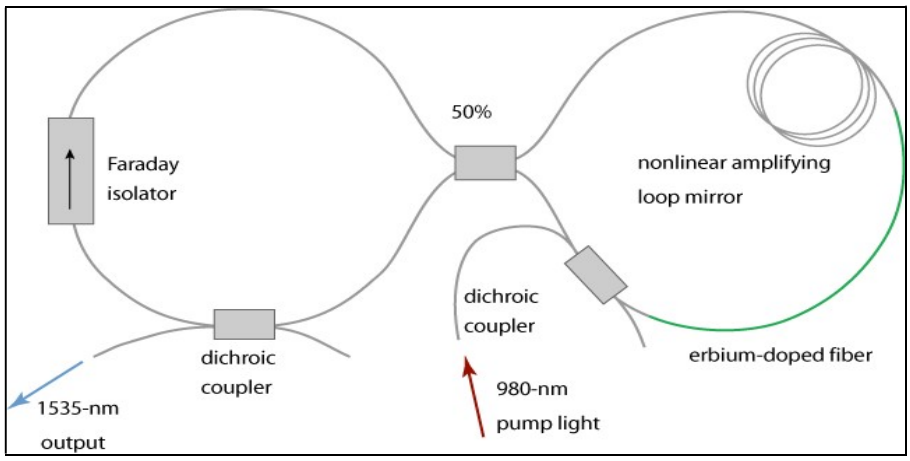

Figure 1.5: Figure-eight fiber laser setup

An advantage of fiber lasers over other types of lasers is that the laser light is both generated and delivered by an inherently flexible medium, which allows easier delivery to the focusing location and target. This can be important for laser cutting, welding, and folding of metals and polymers. Another advantage is high output power compared to other types of laser. Fiber lasers can have active regions several kilometres long, and so can provide very high optical gain. They can support kilowatt levels of continuous output power because of the fiber's high surface area to volume ratio, which allows efficient cooling. The fiber's waveguide properties reduce or eliminate thermal distortion of the optical path, typically producing a diffraction-limited, high-quality optical beam.

1.3 Parameters of laser cutting

Laser cutting has long been a good choice in industrial applications because of the advantage of fast speed. The process parameter settings must be determined in the optimal method to optimize quality with lowest manufacturing cost, time, process safety, and increased productivity. In the fully automated manufacturing process, these ideal parameters are critical for ensuring product quality, increasing productivity, and lowering production costs.

1.3.1 Beam characteristics

Effect of beam type

Both pulsed and continuous beams can be used in laser cutting, with continuous beams being more common. The pulse duty cycle influences the surface roughness, with the surface roughness decreasing as the pulse duty increases. The pulse duty cycle is defined as the ratio

of pulse on time to total pulse time. Figures 12 the continuous and pulsed wave beam shapes, respectively.

Effect of power

Increasing the power has the overall effect of allowing you to cut at faster speeds deeper depths. The potential drawback of raising the power is that the cut breadth expands, the edge finish deteriorates, and sharp corners become rounded. Another feature of high power, which is particularly relevant to high-brightness fiber lasers, is that the temperature of the melt may be high enough to stop the oxygen burning reaction and convert it to a reduction process, resulting in the loss of the extra energy mentioned in reactive fusion cutting. When approaching a corner, some laser cutting systems can switch from continuous wave to pulse mode to prevent side burning or corner rounding. Because the pulse maintains the power intensity, the depth of penetration and pulse rate can be modified in accordance with the speed to control the overall heat input. When cutting fine shapes, other technologies that are not capable of switching during processing are frequently used in the pulsed mode. To cut, the Nd:YAG laser is commonly employed in pulsed mode. There are several types of pulsing: simple power switching, which turns the beam on and switching with excess current, which produces a "super pulse" with an energy 2–3 times that of the CW value; and switching with excess current. Q-switching, which uses a very high-speed switch in the laser cavity – such as a Pocket cell – to provide several thousand times the CW power rating in a very short time, such as a few nanoseconds; and spiking on the CW beam with short power surges to generate a "hyper-pulse," which is available on certain lasers like PRC lasers.

Effect of beam polarization

The cutting in one direction as opposed to one at right angles when cutting with a plane-polarized laser beam, the maximum cutting speed is doubled since the laser beam cuts in one direction rather than at right angles. The amplification of radiation whose electric vector is at right angles to the plane of incidence on the fold mirrors is favored in nearly all high-powered lasers, which contain folded cavities. That example, horizontal folding will result in a vertically polarized beam. The beam will still be plane-polarized if the cavity is not folded or if the folding has near-normal reflections, but the plane of polarization may vary unpredictably over time. As a result, even these lasers now have a fold in the entire reflecting mirror to stabilize the polarization plane from the cavity. The phenomenon is caused by a glancing angle of incidence at the cutting face, and there is a distinct difference in the reflection of a beam at such angles depending on whether the electric vector is at right angles

to the plane of incidence (s-polarization) or in the plane of incidence (p-polarization). It will have a high reflectance if it is s-polarized. It will be absorbed preferentially if it is p-polarized.

Effect of wavelength

For a particular mode of structure and optics train, the shorter the wavelength, the higher is the absorptivity, and the sharper is the focus. As a result, YAG radiation is superior to CO_2 radiation in general; however, the spot diameters for both CO_2 and YAG are similar, with a little advantage for the genuine TEM00 laser due to the poor mode structure of most YAG lasers of any considerable power. Fiber lasers with mono-mode fibers produce TEM00 beams with much higher penetration strengths than other lasers at wavelengths of 1.03 or 1.55 m. The Fulham Laboratory results on cutting plastics with a CO_2 laser emitting 10.6 m radiation and a CO laser emitting 5.4 m radiation were obtained using the same resonator and optics, and thus represent a good comparison between CO_2 10.6-m radiation and CO laser 5.4-m radiation, with the shorter wavelength having an advantage.

The discrepancy is thought to be due to the absorption of the beam on the cut face. Some materials are transparent to specific wavelengths, which creates some intriguing cutting challenges. Human tissue is one such substance. In the ultraviolet, very short wavelengths have energetic photons of energy that are like the bond energy in biological materials (e.g., approximately 4.6 eV). As a result, they can directly break bonds. The material will effectively fall apart if a large flux of photons can concurrently rupture multiple nearby bonds, resulting in a cut without the requirement for heating. This is referred to as "cold cutting." Because some heating has been observed, it is unclear if the mechanism just described is right or not. However, this could be a secondary result rather than the main event. Multiphoton interactions with lower-energy photons can produce a similar effect.

1.3.2 Transport characteristics

Effect of speed

The less time the heat must diffuse sideways and the tighter the HAZ, the faster the cutting. Due to the necessity to deposit a particular amount of energy to promote melting, the kerf is also reduced. As a result, there is a "sharpened pencil" effect with a Gaussian beam, in that as the speed increases, there is only enough energy at the tip of the Gaussian curve, not at the root, to cause melting and hence cutting.

The kerf width changes as the speed increases. The three zones depicted in the diagram are caused by side burning at slow speeds, stable cutting at medium speeds, and dross failure at

higher speeds. Until this last region is reached, the faster the pace, the better the cut finish.

Effect of focal position

The size of the surface spot dictates the intensity of the surface power and whether penetration will occur, although optimal cutting can be achieved by having the smallest spot size below the surface. The issue revolves around energy absorption on the sliced face and how to keep it together. Because the beam stretches out and suffers several reflections, deep cuts are rarely achieved with high quality. There are exceptions, such as a 5-cm cut in the block board with parallel walls. How did the beam manage to do that? It must have been a wave that was guided down a slot with graphite walls - hardly a typical material for reflections.

1.3.3 Gas characteristics

Effect of gas Jet velocity

The gas jet works by drawing the melt out of the cut, according to the description. The faster it can be taken out, the faster the next piece can be melted. The velocity of the gas in the slot on the cut face is critical because the drag force per unit area is 1 2 cfu2 and the drag coefficient cf is a function of the Reynolds number (μd/ (where is the gas density, u is the gas velocity, d is a typical dimension, e.g., kerf width, and is the gas viscosity), Researcher tried a few different angles for directing the jet into the cut front, but it had little effect. Up to a point, increasing the gas jet velocity enhanced the cutting rate. It was perplexing why there was a decrease in cutting speed as nozzle pressure increased. Some early workers speculated that the issue was one of cooling. The tests to show that there was a density gradient field close to the cut surface that may impact the focus at the cut front; however, the density gradient field was shaped like a lens, making the effect difficult to rationalize. They made surface pressure measurements and discovered a variety of shock phenomena linked with high-pressure jets.

Multiple-nozzle systems are employed in a variety of applications. The clean-cut nozzle operates at roughly 1-atm pressure for lens protection on the inner jet and around 5 atm on the outer ring jet. As a result, burr-free, striation-free cuts are produced. Amanda and Prima Industry have invented a ring nozzle with a water spray on the outer nozzle that decreases HAZ, dross, roughness, fume, and smoke, as well as cutting errors caused by expansion when cutting.

Effect of nozzle alignment

The alignment of the nozzle with the laser beam has an impact on the cut quality. A

comprehensive set of trials in which the beam and the jet were methodically mismatched. Both the roughness of the cut and the manner the dross clears the kerf are affected by the alignment.

Effect of gas composition

The gas composition influences the cut quality in some cases. There is a benefit to using pure oxygen; even a 1% impurity will significantly degrade cutting performance when considering the natural mixing that occurs within the kerf during cutting. They demonstrated that cutting at depths more than 15 mm would be problematic due to the reactive gas's efficacy being reduced by mixing. Striations are more likely to appear while cutting reactively. Oxygen can also cause oxide layers to grow on the cut edge. As a result, cutting with inert gas is frequently preferred, especially with stainless steel, to minimize the dross problem caused by the production of high melting point chromium oxides. To increase the drag forces, inert gas cutting is commonly done at very high pressures (14 bar or so) with Laval supersonic nozzles. The cut quality might be exceptional.

1.4 Characteristics of laser cut

The construction of a qualified cutting program, which contains the tested process parameter values or the value range for producing an end user-specified quality level cut, ensures cut quality. The development process can be carried out quickly and easily using pre-existing techniques or predictive methods. Some of them are discussed farther down.

Kerf depth

During through-thickness cutting, the kerf is the slot molded. The kerf width is described as the breadth of the cut at the bottom in optimal laser cutting, and it is usually much larger than the focused beam diameter. The kerf width on the bottom side of the material is typically narrower than on the top side. In gas-assisted thermal cutting procedures, the kerf width is determined by the gas jet width.

Heat affected zone

As the power incident per unit length and the cutting thickness grow, the HAZ widens. When cutting near heat components, the HAZ width is important. As a result, there are times when the maximum limit for the incident beam output or plate thickness is established, or the minimum limit for the cutting speed is set.

Recast layer

The molten slag (solidified oxide) or very adherent form (hardened material) formed at the

bottom of the incision is the re-melted layer. Burrs might appear as prolonged drops or as a rough covering of whiskers. Slag can be mechanically removed after cutting or separated during processing with a gas jet directed from the bottom of the material.

Surface roughness

The presence of sputtering on the surface (remaining molten material ejected from the cutting) and discoloration on the surface characterize the surface condition after cutting.

Taper angle

High laser power can minimize the formation of taper angles, allowing heat to penetrate the thickness of the working material. The material is penetrated more effectively with more laser power, resulting in a greater kerf with less taper. As a result, increasing the laser power increases the material removal rate while decreasing the taper development. The laser beam moves quicker at high cutting speeds, resulting in less time spent at a single point. Larger taper angles were formed at high cutting speeds as a result of the shorter energy input time, which were not adequately conveyed to the thickness of the working material.

1.5 Advantages and disadvantages of laser cutting

There are several advantages and disadvantages of laser cutting they are –

1.5.1 Advantages of laser cutting

High precision and accuracy

precision at which the laser cuts into materials is high, and it is by delivering a focused beam of light. The laser is powerful and small, but the accuracy with which it melts and evaporates materials is unparalleled.

Low cost and more affordable

The economic edge laser cutting has over other CNC machines of the same caliber is one of the advantages of laser technology. There is no need to build a custom tool with the laser cutting technology.

Higher sheet utilization with less waste

use a laser cutter with materials, just a tiny amount of the materials will go to waste. This makes laser cutting stand out from other machines with a considerable portion of the material. With a laser cutter, manufacturers can make maximum use of materials. The cost of

production also decreases due to the higher utilization of the resources and less waste material.

Damage prevention

Another advantage of laser cutting is avoiding damages, even to the narrowest materials.

1.5.2 Disadvantages of laser cutting

Evaporation of some materials

When cut some materials with laser such as plastics, evaporation tends to occur. This is a significant downside to laser cutting.

Limitations to metal thickness

Although laser cutting is compatible with almost every material, including sheets, it is preferable to use another system to cut thick metals.

Production of harmful fumes and gases

We mentioned earlier that laser cutting is compatible with various materials. The compatibility of laser cutting with almost all materials has a downside to it. Each material that will melt during the thermal cutting leads to gases and harmful fumes. The production of these toxic gases is usually typical when the material one working with, is plastics.

1.6 Applications of laser cutting

Cutting is one of the most important and effective uses for industrial lasers. Laser cutting works well with both soft and hard materials. Modern applications of laser cutting technology include medical, engineering, and several disciplines of science. In the electronics industry, laser cutting on ceramic, silicone, and polymer substrates is commonplace. In medical surgery, a laser is used to cut tents that save lives and test tubes. Laser cutting plays a huge role in the medical industry, where extreme precision and tight dimensional tolerances are essential. Because of the medical industry's demand for high volume, this technology aligns with their needs in that it can replicate designs with both accuracy and a quick turnaround.

CHAPTER - 2

MULTI-OBJECTIVE OPTIMISATION

Multi-objective optimization is an area of multiple criteria decision making that is concerned with mathematical optimization problems involving more than one objective function to be optimized simultaneously. Multi-objective optimization has been applied in many fields of science, including engineering, economics, and logistics where optimal decisions need to be taken in the presence of trade-offs between two or more conflicting objectives.

For a nontrivial multi-objective optimization problem, no single solution exists that simultaneously optimizes each objective. In that case, the objective functions are said to be conflicting. A solution is called nondominated, Pareto optimal, Pareto efficient or noninferior, if none of the objective functions can be improved in value without degrading some of the other objective values. Without additional subjective preference information, there may exist a (possibly infinite) number of Pareto optimal solutions, all of which are considered equally good. Researchers study multi-objective optimization problems from different viewpoints and, thus, there exist different solution philosophies and goals when setting and solving them. The goal may be to find a representative set of Pareto optimal solutions, and/or quantify the trade-offs in satisfying the different objectives, and/or finding a single solution that satisfies the subjective preferences of a human decision maker (DM).

2.1 Introduction

A multi-objective optimization problem is an optimization problem that involves multiple objective functions. In mathematical terms, a multi-objective optimization problem can be formulated as-

$$\min_{x \in X} (f_1(x), f_2(x) \ldots\ldots, f_k(x))$$

Where the integer k is less than or equal to 2, is the number of objectives and the set X is the feasible set of decision vectors, which is typically $X \subseteq \mathbb{R}^n$, but it depends on the n-dimensional application domain. The feasible set is typically defined by some constraint functions. In addition, the vector valued objective function is often defined as –

$$f: X \rightarrow \mathbb{R}^k$$

$$x \to \begin{pmatrix} f_1(x) \\ f_k(x) \end{pmatrix}$$

If some objective function is to be maximized, it is equivalent to minimize its negative or its inverse. We denote $X \subseteq \mathbb{R}^n$ the image of X; $x* \in X$ a feasible solution or feasible decision; and $z^* = f(x^*) \in \mathbb{R}^k$ an objective vector or an outcome.

In multi-objective optimization, there does not typically exist a feasible solution that minimizes all objective functions simultaneously. Therefore, attention is paid to Pareto optimal solutions; that is, solutions that cannot be improved in any of the objectives without degrading at least one of the other objectives. In mathematical terms, a feasible solution $x_1 = X$ is said to (Pareto) dominate another solution $x_2 \in X$, if

1. $\forall i \in \{1, \ldots, k\}, f_i(x_i) \leq f_i(x_2)$ and
2. $\exists i \in \{1, \ldots \ldots, k\}, f_i(x_i) < f_i(x_2)$

A solution $x^* \in$ X (and the corresponding outcome f (x*) is called Pareto optimal if there does not exist another solution that dominates it. The set of Pareto optimal outcomes, denoted X*, is often called the Pareto front, Pareto frontier, or Pareto boundary.

The Pareto front of a multi-objective optimization problem is bounded by a so-called nadir objective vector z^{nadir} and an ideal objective vector z^{ideal}, if these are finite. The nadir objective vector is defined as

$$z^{nadir} = \begin{pmatrix} sup_{x^* \in X^*} & f_1(x^*) \\ sup_{x^*} & f_k(x^*) \end{pmatrix}$$

and the ideal objective vector as

$$z^{ideal} = \begin{pmatrix} inf_{x^* \in X^*} & f_1(x^*) \\ inf_{x^* \in X^*} & f_k(x^*) \end{pmatrix}$$

In other words, the components of the nadir and ideal objective vectors define the upper and lower bounds of the objective function of Pareto optimal solutions. In practice, the nadir objective vector can only be approximated as, typically, the whole Pareto optimal set is unknown. In addition, a utopian objective vector z^{utop}, such that $z^{utop} = z^{ideal} - \epsilon$, $\forall i \in \{1 \ldots\ldots, k\}$.

Where $\in>0$ is a small constant, is often defined because of numerical reasons.

2.2 Components of optimization problem

Design variables

The identification of the underlying design variables, which are primarily altered during the optimization process, is the first step in formulating an optimization problem. A design difficulty frequently involves several design factors, some of which are extremely important to the design's proper operation. This set of factors is referred to as design. Other design factors are usually fixed or change in response to the design variables. When formulating an optimization problem, the first rule is to use as few design variables as possible. The results of that optimization technique could indicate whether more design factors should be included in a revised formulation or whether certain previously considered design variables should be replaced with new design variables.

Design constrains

The constraints are functional interactions between design variables and other design factors that fulfil specific physical phenomena and resource limits. It is depended on the user, the type and quantity of constraints to add in the formulation.

Objective function

The objective function in terms of the design variables and other problem parameters is the next step in the formulation phase. Minimization of overall manufacturing costs, minimization of overall component weight, maximizing of a product's complete life cycle, and other engineering goals are prevalent. Although most of the objectives can be quantified (represented in mathematical form), and other objectives (such as the aesthetic aspect of a design, ride characteristics of a car suspension design, and design reliability) may be difficult to define mathematically. In this situation, a mathematical equation is utilized to approximate the solution.

In real-world optimization, the designer may desire to maximize more than one objective at the same time. Multiple objective optimization techniques are complicated and time-consuming to implement. As a result, the most essential goal is chosen as the objective function, and the remaining goals are incorporated as constraints by limiting their values to a specific range. Consider the difficulty of designing the best truss construction. The designer may be concerned with decreasing the total weight of the structure while also minimizing the

deflection of a particular spot in the truss. The weight of the truss (as a function of the cross-sections of the members) may be used as the objective function in the optimization problem formulation, with a restriction on the deflection of the concerned point being smaller than a certain limit.

There are two kinds of objective functions. Either it must be maximized or minimized. The optimization algorithms were usually created to solve minimization or maximization problems. Although modest structural changes in some algorithms allow for either minimization (or) maximizing, this necessitates significant knowledge of the method. The duality principle assists by allowing the same algorithm to be used for both minimization and maximizing with only a tiny modification in the objective function rather than a complete method change. If the algorithm was designed to solve a minimization problem, it can simply be converted to a maximizing problem by multiplying the objective function by one, and vice versa.

2.3 Types of optimization technique

Single variable optimisation functions

These algorithms are classified into two categories

i. Direct methods
ii. Gradient-based methods

Direct approaches do not employ the objective function's derivative information to direct the search process; only the objective function's values are used. Gradient-based approaches, on the other hand, steer the search process using derivative information (first and/or second order). Even though engineering optimization problems typically include several variables, single-variable optimization algorithms are typically utilized as unidirectional search methods in multivariable optimization algorithms.

Multi variable optimisation function

These algorithms show how the search for the best point in several dimensions progresses. These algorithms are divided into direct and gradient-based strategies depending on whether gradient information is used.

Constrained optimisation algorithm

These methods frequently and simultaneously apply single variable and multivariable

optimization algorithms to keep the search effort within the feasible search region. Most of the time, these techniques are employed in engineering optimization issues.

Specialised optimistion algorithm

In engineering design issues, two of these algorithms, integer programming and geometric programming, are frequently utilized. With integer design variables, integer programming methods can be used to tackle optimization problems. Geometric programming approaches are used to solve optimization problems that include objective functions and constraints specified in a certain way.

Non-traditional optimistion algorithm

The optimization strategy is based on natural or natural phenomena. While these methods are quick, they do not guarantee optimal results. Approximation Methods are another name for non-traditional techniques.

2.4 Applications of multiobjective optimisation

2.4.1 Optimal control

In engineering and economics, many problems involve multiple objectives which are not describable as the-more-the-better or the-less-the-better; instead, there is an ideal target value for each objective, and the desire is to get as close as possible to the desired value of each objective. For example, energy systems typically have a trade-off between performance and cost or one might want to adjust a rocket's fuel usage and orientation so that it arrives both at a specified place and at a specified time; or one might want to conduct open market operations so that both the inflation rate and the unemployment rate are as close as possible to their desired values.

Often such problems are subject to linear equality constraints that prevent all objectives from being simultaneously perfectly met, especially when the number of controllable variables is less than the number of objectives and when the presence of random shocks generates uncertainty. Commonly a multi-objective quadratic objective function is used, with the cost associated with an objective rising quadratically with the distance of the objective from its ideal value. Since these problems typically involve adjusting the controlled variables at various points in time and/or evaluating the objectives at various points in time, intertemporal optimization techniques are employed.

2.4.2 Optimal design

Product and process design can be largely improved using modern Modeling, simulation, and optimization techniques The key question in optimal design is the measure of what is good or desirable about a design. Before looking for optimal designs it is important to identify characteristics which contribute the most to the overall value of the design. A good design typically involves multiple criteria/objectives such as capital cost/investment, operating cost, profit, quality and/or recovery of the product, efficiency, process safety, operation time etc. Therefore, in practical applications, the performance of process and product design is often measured with respect to multiple objectives. These objectives typically are conflicting, i.e., achieving the optimal value for one objective requires some compromise on one or more of other objectives.

2.4.3 Process optimization

Multi-objective optimization has been increasingly employed in Laser technology and manufacturing. In 2009, Fiandaca and Fraga used the multi-objective genetic algorithm (MOGA) to optimize the pressure swing adsorption process (cyclic separation process). The design problem involved the dual maximization of nitrogen recovery and nitrogen purity. The results provided a good approximation of the Pareto frontier with acceptable trade-offs between the objectives.

In 2010, Sendín et al. solved a multi-objective problem for the thermal processing of food. They tackled two case studies (bi-objective and triple objective problems) with nonlinear dynamic models and used a hybrid approach consisting of the weighted Tchebycheff and the Normal Boundary Intersection approach. The novel hybrid approach was able to construct a Pareto optimal set for the thermal processing of foods.

In 2013, Ganesan et al. carried out the multi-objective optimization of the combined carbon dioxide reforming and partial-oxidation of methane. The objective functions were methane conversion, carbon monoxide selectivity and hydrogen to carbon monoxide ratio. Ganesan used the Normal Boundary Intersection (NBI) method in conjunction with two swarm-based techniques (Gravitational Search Algorithm (GSA) and Particle Swarm Optimization (PSO)) to tackle the problem.

In 2013, Abakarov et al proposed an alternative technique to solve multi-objective optimization problems arising in food engineering. The Aggregating Functions Approach, the Adaptive Random Search Algorithm, and the Penalty Functions Approach were used to

compute the initial set of the non-dominated or Pareto-optimal solutions. The Analytic Hierarchy Process and Tabular Method were used simultaneously for choosing the best alternative among the computed subset of non-dominated solutions for osmotic dehydration processes.

In 2018, Pearce et al. formulated task allocation to human and robotic workers as a multi-objective optimization problem, considering production time and the ergonomic impact on the human worker as the two objectives considered in the formulation. Their approach used a Mixed-Integer Linear Program to solve the optimization problem for a weighted sum of the two objectives to calculate a set of Pareto optimal solutions. The application of the approach to several manufacturing tasks showed improvements in at least one objective in most tasks and in both objectives in some of the processes.

2.5 Solution of multi-objective optimisation

As there usually exist multiple Pareto optimal solutions for multi-objective optimization problems, what it means to solve such a problem is not as straightforward as it is for a conventional single-objective optimization problem. Therefore, different researchers have defined the term "solving a multi-objective optimization problem" in various ways. This section summarizes some of them and the contexts in which they are used. Many methods convert the original problem with multiple objectives into a single-objective optimization problem. This is called a scalarized problem. If Pareto optimality of the single-objective solutions obtained can be guaranteed, the scalarization is characterized as done neatly.

Solving a multi-objective optimization problem is sometimes understood as approximating or computing all or a representative set of Pareto optimal solutions.

When decision making is emphasized, the objective of solving a multi-objective optimization problem is referred to supporting a decision maker in finding the most preferred Pareto optimal solution according to his/her subjective preferences.

The underlying assumption is that one solution to the problem must be identified to be implemented in practice. Here, a human decision maker (DM) plays an important role. The DM is expected to be an expert in the problem domain.

The most preferred results can be found using different philosophies. Multi-objective optimization methods can be divided into four classes.

1. In so-called no preference methods, no DM is expected to be available, but a neutral compromise solution is identified without preference information. The other classes are so-called a priori, a posteriori and interactive methods and they all involve preference information from the DM in different ways.
2. In a priori methods, preference information is first asked from the DM and then a solution best satisfying these preferences is found.
3. In a posteriori method, a representative set of Pareto optimal solutions is first found and then the DM must choose one of them.
4. In interactive methods, the decision maker is allowed to iteratively search for the most preferred solution. In each iteration of the interactive method, the DM is shown Pareto optimal solution(s) and describes how the solution(s) could be improved. The information given by the decision maker is then considered while generating new Pareto optimal solution(s) for the DM to study in the next iteration. In this way, the DM learns about the feasibility of his/her wishes and can concentrate on solutions that are interesting to him/her. The DM may stop the search whenever he/she wants to.

More information and examples of different methods in the four classes are given in the following sections.

2.5.1 No-preference methods

When a decision maker does not explicitly articulate any preference information the multi-objective optimization method can be classified as no-preference method. A well-known example is the method of global criterion, in which a scalarized problem of the form-

$$\min \left|\left| f(x) - z^{ideal} \right|\right|$$

$$s.t. x \in X$$

is solved. In the above problem, $\left|\left| \cdot \right|\right|$ can be any Lp norm, with common choices including L1, L2 and L_∞. The method of global criterion is sensitive to the scaling of the objective functions, and thus, it is recommended that the objectives are normalized into a uniform, dimensionless scale.

2.5.2 A priori methods

A priori methods require that sufficient preference information is expressed before the solution process. Well-known examples of a priori methods include the utility function method, lexicographic method, and goal programming. In the utility function method, it is assumed that the decision maker's utility function is available. A mapping $u: Y \rightarrow \mathbb{R}$ is a utility function if for all $y^1, y^2 \in Y$ if it holds that $u(y^1) > u(y^2)$ if the decision maker prefers y^1 to y^2, and $u(y^1) = u(y^2)$ if the decision maker is indifferent between y^1 and y^2 .The utility function specifies an ordering of the decision vectors (recall that vectors can be ordered in many ways). Once u is obtained, it suffices to solve.

Max u (f(x)) subject to $x \in X$ but in practice it is very difficult to construct a utility function that would accurately represent the decision maker's preferences - particularly since the Pareto front is unknown before the optimization begins.

The lexicographic method assumes that the objectives can be ranked in the order of importance. We can assume, without loss of generality, that the objective functions are in the order of importance so that f_1 is the most important and f_k the least important to the decision maker. The lexicographic method consists of solving a sequence of single-objective optimization problems of the form

$$\text{Min } f_l(x)$$

$$s.t\, f_j(x) \leq y*_j, j = 1, \ldots\ldots, l-1,$$

$$x \in X$$

Where y^*_j is the optimal value of the above problem with l=j. Thus,

$$y*_1 = \min \{f_1(x), where\ x \in X\}$$

and each new problem of the form in the above problem in the sequence adds one new constraint as l goes from 1 to k. Note that a goal or target value is not specified for any objective here, which makes it different from the Lexicographic Goal Programming method.

2.5.3 Scalarizing

Scalarizing a multi-objective optimization problem is an a priori method, which means formulating a single-objective optimization problem such that optimal solutions to the single-objective optimization problem are Pareto optimal solutions to the multi-objective optimization problem. In addition, it is often required that every Pareto optimal solution can be reached with some parameters of the scalarization. With different parameters for the

scalarization, different Pareto optimal solutions are produced. A general formulation for a scalarization of a multiobjective optimization is thus-

$$\min g(f_1(x), \ldots\ldots f_k(x), \theta) y *_1 = \min \{f_1(x), x \in X\}$$

Where θ is a vector parameter, the set $X_\theta \subseteq X$ is a set depending on the parameter θ and $g : \mathbb{R}^{k+1} \rightarrow \mathbb{R}$ is a function.

Very well-known examples are the so-called

- linear scalarization

  $$\min_{x \in X} \sum_{i=1}^{k} w_i f_i(x)$$

  where the weights of the objectives $w_i > 0$ are the parameters of the scalarization, and the

- $\in$ - constraint method

  $$\min_{s.t} f_i(x)$$

  $$where\ x \in X\ and\ f_i(x) \leq \epsilon_i\ \ for\ i \in \{1, \ldots\ldots k\}\ \{j\},$$

Where upper bounds w_j are parameters as above and f_j is the objective to be minimized. Somewhat more advanced examples are the:

- achievement scalarizing problems of Wierzbicki. One example of the achievement scalarizing problems can be formulated as

  min $$max_{i=1,\ldots,k}\left[\frac{f_i(x) - z_i}{f_i^{nad} - f_i^{utopian}}\right] + \rho \sum_{i=1}^{k} \frac{f_i(x)}{f_i^{nad} - f_i^{utopian}}$$

  $$subject\ to\ x \in X$$

  $\rho > 0$ is a small constant, and z^{nad} and $z^{utopian}$ are the *nadir* and *utopian* vectors, respectively. In the above problem, the parameter is the so-called *reference point* z which represents objective function values preferred by the decision maker.

- Sen's Multi-Objective Programming:

  $$\max \frac{\sum_{j=1}^{r} z_j}{w_j} - . \frac{\sum_{j=r+1}^{r} z_j}{w_{r+1}}$$
  $$s.t\ AX = b$$
  and X≥0

Where W_j is individual optima (Absolute) for objectives of maximization r and minimisation r+1 to s.

2.6 Particle swarm optimization

Particle swarm optimization (PSO) is a computational method that optimizes a problem by iteratively trying to improve a candidate solution with regard to a given measure of quality.

It solves a problem by having a population of candidate solutions, here dubbed particles, and moving these particles around in the search-space according to simple mathematical formula over the particle's position and velocity. Each particle's movement is influenced by its local best known position, but is also guided toward the best known positions in the search-space, which are updated as better positions are found by other particles. This is expected to move the swarm toward the best solutions.

2.6.1 Algorithm

A basic variant of the PSO algorithm works by having a population (called a swarm) of candidate solutions (called particles). These particles are moved around in the search-space according to a few simple formulae. The movements of the particles are guided by their own best-known position in the search-space as well as the entire swarm's best-known position. When improved positions are being discovered these will then come to guide the movements of the swarm.

Formally, let $f: \mathbb{R}^n \to \mathbb{R}$ be the cost function which must be minimized. The function takes a candidate solution as an argument in the form of a vector of real numbers and produces a real number as output which indicates the objective function value of the given candidate solution. The gradient of f is not known. The goal is to find a solution "a" for which $f(a) \leq f(b)$ for all "b" in the search-space, which would mean "a" is the global minimum.

Let S be the number of particles in the swarm, A basic PSO algorithm is then:

for each particle i = 1, ..., S do

Initialize the particle's position with a uniformly distributed random vector: xi ~ $U(b_{lo}, b_{up})$

Initialize the particle's best known position to its initial position: $p_i \leftarrow x_i$

if f(p_i) < f(g) then

update the swarm's best known position: g ← p_i

Initialize the particle's velocity: vi ~ U(-|b_{up}- b_{lo}|, |b_{up}- b_{lo} |)

while a termination criterion is not met do:

for each particle i = 1, ..., S do

for each dimension d = 1, ..., n do

Pick random numbers: r_p, r_g ~ U(0,1)

Update the particle's velocity: $v_{i,d} \leftarrow w\ v_{i,d} + \varphi_p\ r_p\ (p_{i,d} - x_{i,d}) + \varphi_g\ r_g\ (g_d - x_{i,d})$

Update the particle's position: $x_i \leftarrow x_i + v_i$

if f(x_i) < f(p_i) then

Update the particle's best known position: $p_i \leftarrow x_i$

if f(p_i) < f(g) then

Update the swarm's best known position: g ← p_i

The values b_{lo} and b_{up} represent the lower and upper boundaries of the search-space respectively. The w parameter is the inertia weight. The parameters φ_p and φ_g are often called cognitive coefficient and social coefficient.

2.6.2 Parameter selection

The choice of PSO parameters can have a large impact on optimization performance. Selecting PSO parameters that yield good performance has therefore been the subject of much research. The PSO parameters can also be tuned by using another overlaying optimizer, a concept known as meta-optimization or even fine-tuned during the optimization, e.g., by means of fuzzy logic. Parameters have also been tuned for various optimization scenarios.

2.6.3 Inner working of PSO

There are several schools of thought as to why and how the PSO algorithm can perform optimization.

A common belief amongst researchers is that the swarm behaviour varies between exploratory behaviour, that is, searching a broader region of the search-space, and exploitative behaviour, that is, a locally oriented search to get closer to a (possibly local) optimum. This school of thought has been prevalent since the inception of PSO. This school of thought contends that the PSO algorithm and its parameters must be chosen so as to

properly balance between exploration and exploitation to avoid premature convergence to a local optimum yet still ensure a good rate of convergence to the optimum. This belief is the precursor of many PSO variants.

Another school of thought is that the behaviour of a PSO swarm is not well understood in terms of how it affects actual optimization performance, especially for higher-dimensional search-spaces and optimization problems that may be discontinuous, noisy, and time-varying. This school of thought merely tries to find PSO algorithms and parameters that cause good performance regardless of how the swarm behaviour can be interpreted in relation to e.g., exploration and exploitation. Such studies have led to the simplification of the PSO algorithm.

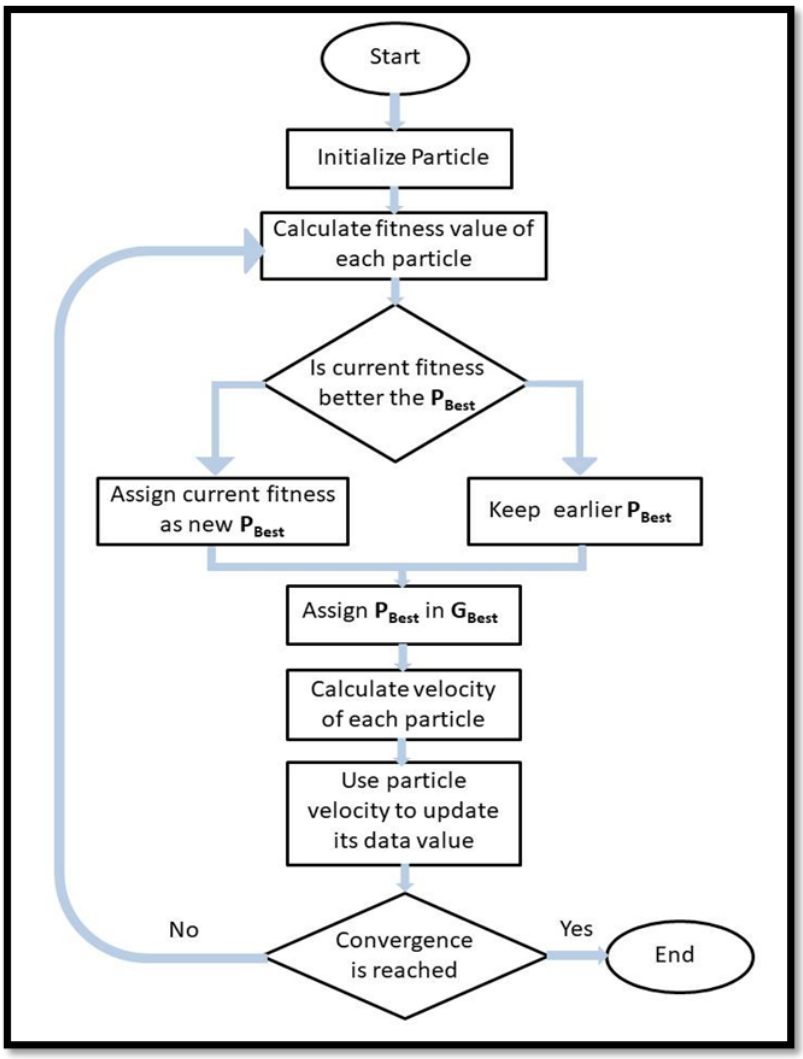

Figure 2.1: Algorithm of PSO Optimisation

2.6.4 Variants of PSO algorithm

Numerous variants of even a basic PSO algorithm are possible. For example, from the above algorithm, there are different ways to initialize the particles and velocities (e.g., start with zero velocities instead), how to dampen the velocity, only update p_i and g after the entire swarm has been updated, etc.

2.6.4.1 Hybridization

New and more sophisticated PSO variants are also continually being introduced to improve optimization performance. There are certain trends in that research; one is to make a hybrid optimization method using PSO combined with other optimizers e.g., combined PSO with biogeography-based optimization and the incorporation of an effective learning method.

2.6.4.2 Gradient based PSO algorithms

The ability of the PSO algorithm to efficiently explore multiple local minimums can be combined with the ability of gradient based local search algorithms to effectively compute an accurate local minimum to produce gradient based PSO algorithms. In gradient based PSO algorithms, the PSO algorithm is used to explore many local minima and locate a point in the basin of attraction of a deep local minimum. Then efficient gradient based local search algorithms are used to accurately locate the deep local minimum. The computation of gradients and Hessians of complex high-dimensional cost functions is often computationally costly and manually impossible in many cases preventing the widespread adoption of gradient based PSO algorithms.

2.6.4.3 Multi objective optimisation

PSO has also been applied to multi-objective problems, in which the objective function comparison takes pareto dominance into account when moving the PSO particles and non-dominated solutions are stored so as to approximate the pareto front.

2.6.4.4 Binary, discrete, and combinatorial

As the PSO equations given above work on real numbers, a commonly used method to solve discrete problems is to map the discrete search space to a continuous domain, to apply a classical PSO, and then to de map the result. Such a mapping can be very simple (for example by just using rounded values) or more sophisticated.

2.7 Response surface methodology

In statistics, Response surface methodology (RSM) explores the relationships between several explanatory variables and one or more response variables. The main idea of RSM is to use a sequence of designed experiments to obtain an optimal response. Box and Wilson suggest using a second-degree polynomial model to do this. They acknowledge that this model is only an approximation, but they use it because such a model is easy to estimate and apply, even when little is known about the process.

Statistical approaches such as RSM can be employed to maximize the production of a special substance by optimization of operational factors. Of late, for formulation optimization, the RSM, using proper design of experiments (DoE), has become extensively used. In contrast to conventional methods, the interaction among process variables can be determined by statistical techniques.

2.7.1 Basic approach of response surface methodology

An easy way to estimate a first-degree polynomial model is to use a factorial experiment or a fractional factorial design. This is sufficient to determine which explanatory variables affect the response variable(s) of interest. Once it is suspected that only significant explanatory variables are left, then a more complicated design, such as a central composite design can be implemented to estimate a second-degree polynomial model, which is still only an approximation at best. However, the second-degree model can be used to optimize (maximize, minimize, or attain a specific target for) the response variable(s) of interest.

2.7.2 Important response surface methodology properties

Orthogonality

The property that allows individual effects of the k-factors to be estimated independently without (or with minimal) confounding. Also, orthogonality provides minimum variance estimates of the model coefficient so that they are uncorrelated.

Rotatability

The property of rotating points of the design about the center of the factor space. The moments of the distribution of the design points are constant.

Uniformity

A third property of CCD designs used to control the number of center points is uniform precision (or Uniformity).

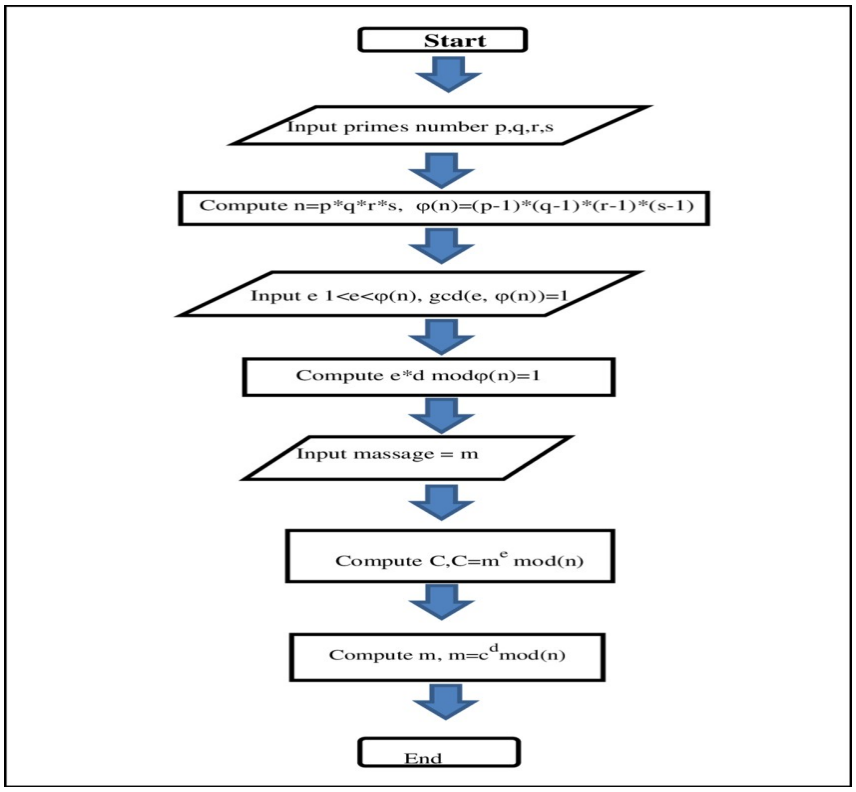

Figure 2.2: Flow chart of Enhanced RSM based Multi Objective Optimisation

2.7.3 RSM as a multi objective functions

Some extensions of response surface methodology deal with the multiple response problem. Multiple response variables create difficulty because what is optimal for one response may not be optimal for other responses. Other extensions are used to reduce variability in a single response while targeting a specific value, or attaining a near maximum or minimum while preventing variability in that response from getting too large.

CHAPTER – 3

LITERATURE REVIEW

3.1 Past research in the field of optimisation of laser cutting

1. The hybrid Taguchi technique and Response Surface Method (TMRSM) was utilised by **Kumar Dubey A. and Yadava V *et al.*** for the multi-response optimization of laser beam cutting. Approach first determines the optimum value of input cutting parameters, such as assist Gas Pressure, Pulse Width, Pulse Frequency, and Cutting Speed, using the Taguchi quality loss function. The second-order response model is developed and optimised using the best input parameter values as the response surface method's central values. For simultaneous optimization, the two separate quality parameters Kerf width (KW) and material removal rate (MRR)—where KW is of the smaller-the-better type and MRR is of the higher-the-better type—have been chosen. The results show considerable improvement in both the quality characteristics when the hybrid approach is used, as compared the results of a single approach.

2. The prospects of obtaining high quality characteristics (surface roughness and kerf width) using pulsed CO2 laser cutting of Al6061/Al2O3 composite were examined by **Adalarasan R., Santhanakumar M., and Thileepan S *et al.*** . The non-contact cutting process's variables, such as the beam power, assist gas pressure, flow rate, pulse frequency, and spot movement speed, were changed in accordance with a structured L27 orthogonal array, and the quality characteristics, such as surface finish and kerf width, were examined. In order to create quadratic models for the observed quality characteristics, Response Surface Methodology (RSM) was employed. Desirability analysis was then used to determine the ideal laser cutting parameter setting. There were fewer cutting trials because the research was based on Taguchi's L27 orthogonal array rather than the central composite design. Application of the Taguchi approach in RSM optimisation process was found to be both time and cost efficient (reduced experimentation).

3. **Rajamani D. and Tamilarasan A. *et al.*** suggest a multi-response optimization strategy for the titanium superalloy sheet's Nd: YAG laser cutting parameters (Ti6Al-4V). The studies were planned using the Box-Behnken design, and experimental models were created using the response surface methodology. During the experiment,

four input parameters—pulse width, pulse energy, cutting speed, and gas pressure—were set, and the performance metrics—kerf deviation and metal removal rate—were considered. The performance characteristic was optimised using a search optimization approach based on the use of desire function after the appropriate mathematical models had been developed.

4. **Chaki Sudipto, Bose Digbalay and Bathe N. Ravi *et al.*** proposed in their study to evaluate the performance of an entropy-based ANN-PSO model that combines Artificial Neural Networks (ANN) and Particle Swarm Optimization (PSO) for quality characteristic estimation and optimization in pulsed Nd:YAG laser cutting of aluminium alloy. In the ANN-PSO model, the objective function value is computed and estimated using an ANN that has been trained using backpropagation and the Bayesian regularisation approach. When formulating the combined goal function, the real weight of various output quality criteria is determined using the entropy technique. Cutting speed, pulse energy, and pulse breadth are considered as controllable input parameters while kerf width, kerf deviation, surface roughness, and material removal rate are monitored as output parameters in an experiment based on a full factorial experimental design. The ANN-PSO model has also optimised and predicted using the experimental dataset. Low mean absolute error indicates the ANN module's prediction accuracy. The experimental validation of the improved output yields an error rate of only 2% or less. According to an ANOVA analysis, cutting speed is the main deciding factor.
5. **Prasant Kumar Shrivastava, Arun Kumar Pandey *et al***. has done the experimental research while cutting Inconel-718 with a pulsed Nd: YAG laser. The carefully thought-out orthogonal array L27 has been used to conduct the trials. The bottom kerf deviation (KD), bottom kerf width (KW), and kerf taper (KT) second order regression models were created using the empirically measured values of various quality criteria. Using the particle swarm optimization (PSO) method, the constructed models of various quality attributes have been used as a quality function for single-objective optimization. The best outcomes produced by the suggested hybrid methodology have been contrasted with those of experiments. The bottom kerf deviation, bottom kerf width, and Kerf taper all individually improved by 75%, 12.67%, and 33.70% when the optimised results were compared to the experimental data.

6. **M.M.Noor, K.Kadirgama and M.M.Rahman *et al.*** developed an artificially intelligent model using partial swarm optimization (PSO) to forecast the optimum level of surface roughness when laser-beam cutting acrylic sheets (LBC). They conducted as few experiments as possible; the response surface method (RSM) was applied. Investigated were the effects of cutting speed, material thickness, tip gap, and power on surface roughness. It was discovered that the tip distance, power demand, cutting speed, and material thickness have the biggest impacts on surface roughness. Low power, a close tip distance, and thicker material all increase surface roughness. Fine surface roughness is produced by a combination of low cutting speed, high power, tip distance, and material distance. Optimised predicted value of parameter was found using PSO.
7. **Manish Kumar Gupta, P.K.Sood and Vishal S. Sharma *et al.*** showed how to use particle swarm optimization (PSO) and response surface methodology (RSM) to optimise the machining factors when turning titanium (Grade-II) alloy with a cubic boron nitride insert tool in a minimal quantity lubricant (MQL) environment. Cutting speed (Vc), feed rate (f), and side cutting edge angle (approach angle) are the three machining factors that are designed as three factors by using RSM design. Tangential force (Fc), tool wear (VBmax), surface roughness (Ra), and tool-chip contact length are some of the constraints that also apply to this design (L). The multiple regression technique was used to establish the interaction between input parameters and given responses. Additionally, the findings have been made public, and the improved process parameters are obtained using a multi-response optimization method using the desirability function as PSO technique. Additionally, it was noted that the PSO-based responses that were selected are significantly closer to that the values obtained when using the desirability function approach. From this point forward, PSO has the potential to eliminate the necessary machining parameters when turning titanium (Grade-II) alloys under MQL conditions.
8. **Rashmi Laxmikant Malghan, Karthik M.C. Rao, Arun Kumar Shettigar, Shrikantha S. Rao, R. J, D'Souza *et al.*** illustrated an application of response surface methodology (RSM) and particle swarm optimization (PSO) technique for optimizing the process parameters of milling and provides a comparison study among desirability and PSO techniques. The experimental investigations are carried out on metal matrix composite material AA6061-4.5%Cu-5%SiCp to study the effect of process parameters such as feed rate, spindle speed and depth of cut on the cutting force,

surface roughness and power consumption. The process parameters are analysed using RSM central composite face-centered design to study the relationship between the input and output responses. The interaction between the process parameters was identified using the multiple regression technique, which showed that spindle speed has major contribution on all the responses followed by feed rate and depth of cut. It has shown good prediction for all the responses. The optimized process parameters are acquired through multiresponse optimization using the desirability approach and the PSO technique. The results obtained from PSO are closer to the values of the desirability function approach and achieved signifcant improvement.

9. **Prasant Kumar Shrivastava, Bhagat Singh and Yogesh Shrivastava *et al*.** developed a reliable prediction model that can offer recommendations for the ideal range of cutting parameters to achieve improved cutting quality, precision, and geometrical accuracy. A 300 W (CNC-PCT 300) pulsed Nd: YAG laser cutting system has been put through tests with a range of input cutting parameters, including gas pressure, standoff distance, cutting speed, and laser power. Then, using the input cutting parameters for top kerf width and bottom kerf width, mathematical models for geometrical quality attributes have been developed using response surface approach. By contrasting the projected values with the experimental ones, these generated models have been proven to be accurate. Further, these models have been optimized using the multiobjective genetic algorithm in order to ascertain the optimal range of cutting parameters pertaining to better quality cut with high precision and geometrical accuracy.mized using the multiobjective genetic algorithm in order to ascertain the optimal range of cutting parameters pertaining to better quality cut with high precision and geometrical accuracy.
10. **Muhamad Nur Rohman, Jeng-Rong Ho, Pi-Cheng Tung, Chin-Te Lin, Chih-Kuang Lin *et al.*** analysed dross formation using random forest method and response surface method showed that the cutting environment, laser power, pulse frequency, and cutting speed had a significant influence on the dross formation. In addition, cutting in oil leads to less dross formation than in alcohol and air. A stacked autoencoder method combined with a multi-objective grey wolf optimizer was employed to generate a pre-trained DNN, followed by a fine-tuning process to obtain the final DNN. The I-GWO was used to determine the optimal combination of process parameters for minimum dross formation. The DNN model proved its efficacy by showing very low values of mean absolute percentage error and very high values of

absolute fraction of variation for the training, validation, and testing datasets. Moreover, the accuracy of the developed DNN model was higher than that of other artificial intelligence based methods, namely random vector functional link and support vector machine for regression, as evaluated by nine statistical criteria. The predicted optimal process parameters by the DNN and I-GWO al- gorithms were verified by validation experiments in which the minimum dross formation was generated.

11. **A D Tura, H B Mamo, and D G Desisa *et al.*** used a genetic algorithm (GA) in conjunction with response surface approaches to improve surface roughness in laser beam cutting CO2 with a continuous wave of SS 304 stainless steel. The effects of the machining parameters, such as cutting speed, nitrogen gas pressure, and focal point location, were investigated quantitatively, and optimized. The optimum results of each method were compared, as the results the response surface approach is less promising than the genetic algorithm method.
12. **Senthilkumar Vagheesan and Jayaprakash Govindarajalu *et al*.** proposed using a hybrid artificial neural network (ANN)-genetic algorithm (GA) and artificial neural network (ANN)-particle swarm optimization (PSO) approach to optimise the multi-response characteristics that relate to CO2 laser cutting of aluminium 6061 alloys. The outcomes demonstrate the effectiveness of the hybrid ANN-GA and ANN-PSO model as a tool for process parameter optimization in CO2 laser cutting of the challenging material, aluminium. From the results of the optimization, it can be inferred that the proposed ANN-GA method can be effectively used to optimise the parameters for achieving minimum roughness, kerf width, and kerf taper.
13. **D. Rajamani, M. Siva Kumar, E. Balasubramanian and A. Tamilarasan (2021) *et al*.** hybrid approach through combining genetic algorithm (GA) and adaptive neuro-fuzzy inference system (ANFIS) for modeling the correlation of laser beam cutting (LBC) parameters and enhancing the quality performance characteristics of machined Hastelloy C276 is emphasized. The LBC experiments are performed by considering gas pressure (GP), cutting speed (CS), pulse energy (PE) and stand-off distance (SOD) as input parameters. The output responses are material removal rate (MRR), kerf taper (KT) and surface roughness (Ra) for the present investigation. The optimal ANFIS training variables are obtained through GA. The training, testing errors, and statistical validation parameter results exposed that the ANFIS learned by GA is outperformed in forecasting LBC responses. In addition, to obtain the optimal

combinations of LBC parameters, the multi-response optimization based on maximizing MRR and minimizing KT and Ra was performed using a trained ANFIS network coupled with a whale optimization algorithm (WOA).

14. **J. Ciurana, G. Arias and T. Ozel *et al*.** focused on modeling and optimizing process parameters in pulsed laser micromachining. Artificial neural networks have been used to simulate the relationship between process parameters and quality criteria .Experimental research has been contrasted with predictions made using ANNs. Particle swarm optimization (PSO) with multiple objectives is used to determine the process parameters for the minimum available surface roughness and volume error. This finding demonstrates that the suggested models and swarm optimization approach can be used to obtain the optimal process settings.
15. **Prasant Kumar Shrivastava, Bhagat Singh and Yogesh Shrivastava *et al.*** created a reliable prediction model that can offer recommendations for the necessary range of cutting parameters to achieve higher cutting quality, more precision, and more accurate geometrical results. On a 300 W (CNC-PCT 300) pulsed Nd: YAG laser cutting system, experiments have been conducted at different levels of input cutting parameters, including gas pressure, standoff distance, cutting speed, and laser power. After that, the Top Kerf Width, Bottom Kerf Width, and Kerf Taper input cutting parameters for geometrical quality features were used to create mathematical models using the artificial neural network technique. Furthermore, in order to determine the ideal range of cutting parameters referring to superior quality cuts with high precision and geometrical accuracy, these models have been refined utilising a multiobjective genetic algorithm and particle swarm optimization approach.
16. **Vipin Kumar Sharma and Vinod Kumar *et al***. has done experimental work, power, gas pressure, pulse frequency and feed were taken as input parameters and kerf width, kerf deviation and material removal rate were taken as output responses. The experiments were planned using Box–Behnken design, and empirical models have been developed by using response surface methodology. Scanning electron microscope was used to determine the surface characteristics of sample. Confirmation tests were conducted to validate predicted optimum results. Considerable reduction in kerf deviation and increase in material removal rate were revealed by confirmation tests. The percentage prediction error in the case of kerf width was 5.12% and in the case of kerf deviation was 5.88%, which shows that model is adequate. The experimental and statistical results revealed that the most signifcant laser cutting

parameters for kerf width, kerf deviation and MRR is feed followed by gas pressure, frequency, and power.

17. **Yadvinder Singh *et al.*** focused on optimising various quality characteristics for Kerf geometry. He has done cutting trials designed using L_9 orthogonal array and the process optimisation was conducted on Response Surface Methedology (RSM).
18. **Vi Nguyen, Faisal Altarazi and Thanh Tran *et al.*** presented that Taguchi Method can be a pratical approach for optimisation problems, it can help reduce cost and time and achieve the desired optimal outputs, but with cutting problem with high precision RSM method is highly recommended compare to Taguchi method.
19. **Konstantinos Ninikas, John kechagias, and Konstantinos Salonitis *et al.*** observed that during the cutting of PMMA thin sheet using CO_2 laser cutting optimisation process RSM has been used to optimised the Process Parameters and optimal value of output has been found by him.

3.2 Objective of the present research

From the review of the past literatures, it is observed that some research studies have already been carried out in the field of modelling and optimization of laser cutting of advanced engineering materials. But it is observed that so far, no research work has been reported which can maximize the laser cutting speed while maintaining the required laser cutting characteristics. In view of this fact, the objective of the present research study has been framed as follows:

i) To investigate the existing optimisation technique that has been used in different laser cutting by the past research.

ii) To explore different optimisation strategies and compare their relative performances.

ii) To develop advanced optimization strategy that can improve the productivity of laser cutting.

CHAPTER - 4

MACHINING OF TI -6AL -4V SUPER ALLOY SHEET

4.1 Modelling for laser cutting of Ti-Al-4V superalloy sheet:

4.1.1 Material properties of Ti-Al-4V material:

Titanium and nickel superalloys with intricately formed, extremely complex features benefit greatly from the current manufacturing environment. Ti-6Al-4V alloys are among these materials and are widely used in the aerospace sectors due to their inherent exceptional properties, such as excellent corrosion resistance, high temperature withstanding capability, and low thermal conductivity at higher temperatures. Ti-6Al-4V alloys are incredibly difficult to machine because of their strong chemical affinity and poor heat conductivity. In these circumstances, high-precision laser cutting is a practical option for cutting these alloys.

Table 4.1: Chemical properties of Ti-6Al-4V:

	V	Al	Fe	O	C	N	H	Y	Ti	Remainder Each	Reminder Total
Min	3.5	5.5	--	--	--	--	--	--	--	--	--
Max	4.5	6.75	0.3	0.2	0.08	0.5	0.015	0.005	Balance	0.1	0.3

Physical and mechanical Properties

The alpha (SG: P63/mmc) and beta (SG: Im-3m) phases of the titanium alloy Ti-6Al-4V, both of which have hcp crystal structures, are the most common states of this material. Typical property ranges for well-processed Ti-6Al-4V are listed below, though the mechanical properties of an alloy may vary based on its properties and the heat treatment state of the alloy. Aluminium helps to stabilize the alpha phase while vanadium stabilises the beta phase.

Table 4.2: Mechanical and physical properties of Ti-6Al-4V:

	Density, g/cm^3	Young's Modulus, GPa	Shear Modulus, GPa	Bulk Modulus, GPa	Poisson's Ratio	Yield Stress, MPa (Tensile)	Ultimate Stress, MPa (Tensile)	Hardness, Rockwell C	Uniform Elongation%
Min	4.429	104	40	96.8	0.31	880	900	36	5
Max	4.512	113	45	153	0.37	920	950	--	18

Ti-6Al-4V has a poor machinability due to its extremely low thermal conductivity, which ranges from 6.7 to 7.5 W/mK at ambient temperature.

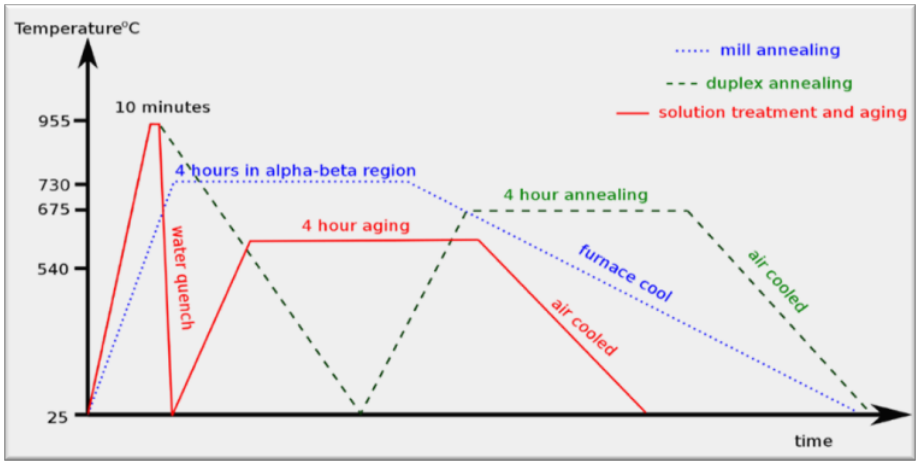

Figure 4.1: Mill anneal, duplex anneal, and solution treatment and aging heat treatment processes for Ti-6Al-4V. Exact times and temperatures will vary by manufacturer.

4.1.2 Experimental plan of laser cutting of Ti-6Al-4V material

Tamilarasan A. and Rajamani D. *et al*. in "Multi response optimisation of Nd: YAG laser cutting parameters of Ti-6Al-4V superalloy sheet" performed the experiments on $200 \times 200 \times 1\ mm$ Ti-6Al-4V superalloy sheet. The designed the experiments with Box-Behnken experimental design considering the process parameters, including pulse width, pulse energy, cutting speed and gas pressure with each in three levels:

Table 4.3: Level value of each process parameter

Process Parameter (unit)	Selected symbol	Low value (-1)	Middle value (0)	High value (+1)
Pulse width (ms)	A	1.5	1.75	2
Pulse energy (J)	B	2.5	4	5.5
Cutting speed (mm/min)	C	10	14	18
Gas pressure(kg/cm^2)	D	6	7.5	9

Levels of input parameters range from -1 to +1, hence level -1 is indicating the lower values and level +1 indicating the higher values.

The response variables investigated in this thesis work done by Tamilarasan A. and Rajamani D. et al. Box Behnken design and results of experiments value were given in the follow table:

Table 4.4 : Box Behnken design layout and results

Run	Pulse Width	Pulse Energy	Cutting Speed	Gas Pressure	Kerf Deviation	MRR
Unit	ms	J	mm/min	Kg/cm^2	mm	mg/min
1	2.00	4.00	18	7.5	0.0146	160.93
2	2.00	4.00	14	6.0	0.0155	88.29
3	1.75	5.5	10	7.5	0.0130	166.38
4	1.75	2.5	10	7.5	0.0139	132.33
5	1.75	4.0	14	7.5	0.0123	161.38
6	1.75	5.5	14	9.0	0.0141	56.29
7	2.00	5.5	14	7.5	0.0142	168.19
8	1.75	4.0	18	6.0	0.0121	155.48
9	1.75	4.0	14	7.5	0.0121	160.02
10	1.50	4.0	14	6.0	0.0168	84.89
11	2.00	4.0	14	9.0	0.0144	66.73
12	1.75	4.0	14	7.5	0.0123	174.55
13	1.75	5.5	18	7.5	0.0129	164.33
14	1.50	4.0	14	9.0	0.0197	28.37
15	1.75	4.0	10	9.0	0.0131	156.62
16	1.50	2.5	14	7.5	0.0185	66.05
17	1.75	4.0	10	6.0	0.0145	110.54
18	1.50	4.0	18	7.5	0.0125	74.68
19	1.75	4.0	14	7.5	0.0129	165.92
20	1.75	4.0	14	7.5	0.0126	171.37
21	1.75	4.0	18	9.0	0.0145	66.73
22	2.00	2.5	14	7.5	0.0149	66.73
23	2.00	4.0	10	7.5	0.0101	161.38
24	1.75	2.5	18	7.5	0.0133	123.48
25	1.50	5.5	14	7.5	0.0175	57.20
26	1.75	2.5	14	9.0	0.0187	43.81
27	1.75	2.5	14	6.0	0.0147	77.17
28	1.50	4.0	10	7.5	0.018	147.76
29	1.75	5.5	14	6.0	0.0167	141.41

Table 4.4 shows that the experiment design and outputs after laser cutting of Ti-6Al-4V using response surface methodology. The kerf deviation (KD) is the difference between the top kerf width at their highest and minimum along the cutting length. The KD can be expressed as –

Kerf deviation (KD) = (Max Top kerf width – Min top kerf width)

Tamilarasan A. and Rajamani D. et.al in "Multi response optimisation of Nd: YAG laser cutting parameters of Ti-6Al-4V superalloy sheet" used the empirical formula (shown below) to determine the MRR:

$$\text{Metal removal rate, (MRR)(mg/min)} = \left(\frac{\textit{Loss of mass during each cut} * \textit{cutting speed}}{\textit{length of each cut}}\right)$$

By weighing the specimen using a Contech electronic balance (Model CAS 234) before and after the cutting of Ti-6Al-4V superalloy sheet.

At each stage and each case of response variable regression model were compared with the help of R^2 value adjusted R^2 value, predicted R^2 value, lack of fit value and p value from the ANOVA table, the most suitable regression model was considered. The regression models were obtained from regression analysis method in design expert 13 version software.

4.1.3 Software identification

Design Expert: Design Expert 13 product version package is used to evaluate the effects of process parameters, data analysis, and quadratic model generation.

MATLAB: MATLAB was used to implement the PSO algorithm. Pseudo code of MOPSO algorithm is given below –

MOPSO code

```
% ------------------------------------------------------------------------ %
% Function MOPSO performs a Multi-Objective Particle Swarm Optimization   %
% over continous functions.                                               %
%                                                                         %
%   Input parameters:                                                     %
%       - params:   Struct that contains the customized parameters.       %
%           * params.Np:        Number of particles.                      %
%           * params.Nr:        Repository size (in particles).           %
%           * params.maxgen:    Maximum number of generations.            %
%           * params.W:         Inertia coefficient.                      %
%           * params.C1:        Personal confidence factor.               %
%           * params.C2:        Swarm confidence factor.                  %
%           * params.ngrid:     Number of hypercubes in each dimension.   %
%           * params.maxvel:    Maximum velocity (search space percentage)%
%           * params.u_mut:     Uniform mutation percentage.              %
%       - MultiObj: Struct that contains the parameters relative to the   %
%                   optimization functions.                               %
%           * MultiObj.fun:     Anonymous multi-obj function to minimize. %
%           * MultiObj.nVar:    Number of variables.                      %
%           * MultiObj.var_min: Vector that indicates the minimum values  %
%                               of the search space in each dimension.    %
%           * MultiObj.var_max: Same than 'var_min' with the maxima.      %
% ------------------------------------------------------------------------ %
%   For an example of use, run 'example.m'.                               %
% ------------------------------------------------------------------------ %
%   Author:  Victor Martinez Cagigal                                      %
%   Date:    17/03/2017                                                   %
```

```
%   E-mail:  vicmarcag (at) gmail (dot) com                          %
%   Version: 1.1                                                     %
%   Log:                                                             %
%          - 1.0:  Initial version without mutation [1] (15/03/2017).    %
%          - 1.1:  Crowding and mutation are implemented [2].            %
% ------------------------------------------------------------------------ %
%   References:                                                      %
%    [1]Coello, C. A. C., Pulido, G. T., & Lechuga, M. S. (2004). Handling%
%       multiple objectives with particle swarm optimization. IEEE Tran-  %
%       sactions on evolutionary computation, 8(3), 256-279.              %
%                                                                        %
%    [2]Sierra, M. R., & Coello, C. A. C. (2005, March). Improving PSO-   %
%       based multi-objective optimization using crowding, mutation and ?-%
%       dominance. In International Conference on Evolutionary Multi-Crite%
%       rion Optimization (pp. 505-519). Springer Berlin Heidelberg.      %
% ------------------------------------------------------------------------ %
function REP = MOPSO(params,MultiObj)

    % Parameters
    Np      = params.Np;
    Nr      = params.Nr;
    maxgen  = params.maxgen;
    W       = params.W;
    C1      = params.C1;
    C2      = params.C2;
    ngrid   = params.ngrid;
    maxvel  = params.maxvel;
    u_mut   = params.u_mut;
    fun     = MultiObj.fun;
    nVar    = MultiObj.nVar;
    var_min = MultiObj.var_min(:);
    var_max = MultiObj.var_max(:);

    % Initialization
    POS = repmat((var_max-var_min)',Np,1).*rand(Np,nVar) + repmat(var_min',Np,1);
    VEL = zeros(Np,nVar);
    POS_fit  = fun(POS);
    if size(POS,1) ~= size(POS_fit,1)
       warning(['The objective function is badly programmed. It is not returning' ...
          'a value for each particle, please check it.']);
    end
    PBEST    = POS;
    PBEST_fit= POS_fit;
    DOMINATED= checkDomination(POS_fit);
    REP.pos  = POS(~DOMINATED,:);
    REP.pos_fit = POS_fit(~DOMINATED,:);
    REP      = updateGrid(REP,ngrid);
    maxvel   = (var_max-var_min).*maxvel./100;
    gen      = 1;
```

```
    % Plotting and verbose
    if(size(POS_fit,2)==2)
        h_fig = figure(1);
        h_par = plot(POS_fit(:,1),POS_fit(:,2),'or'); hold on;
        h_rep = plot(REP.pos_fit(:,1),REP.pos_fit(:,2),'ok'); hold on;
        try
            set(gca,'xtick',REP.hypercube_limits(:,1)','ytick',REP.hypercube_limits(:,2)');
            axis([min(REP.hypercube_limits(:,1)) max(REP.hypercube_limits(:,1)) ...
                min(REP.hypercube_limits(:,2)) max(REP.hypercube_limits(:,2))]);
            grid on; xlabel('f1'); ylabel('f2');
        end
        drawnow;
    end
    if(size(POS_fit,2)==3)
        h_fig = figure(1);
        h_par = plot3(POS_fit(:,1),POS_fit(:,2),POS_fit(:,3),'or'); hold on;
        h_rep = plot3(REP.pos_fit(:,1),REP.pos_fit(:,2),REP.pos_fit(:,3),'ok'); hold on;
        try

set(gca,'xtick',REP.hypercube_limits(:,1)','ytick',REP.hypercube_limits(:,2)','ztick',REP.hyper
cube_limits(:,3)');
            axis([min(REP.hypercube_limits(:,1)) max(REP.hypercube_limits(:,1)) ...
                min(REP.hypercube_limits(:,2)) max(REP.hypercube_limits(:,2))]);
        end
        grid on; xlabel('f1'); ylabel('f2'); zlabel('f3');
        drawnow;
        axis square;
    end
    display(['Generation #0 - Repository size: ' num2str(size(REP.pos,1))]);

    % Main MPSO loop
    stopCondition = false;
    while ~stopCondition

        % Select leader
        h = selectLeader(REP);

        % Update speeds and positions
        VEL = W.*VEL + C1*rand(Np,nVar).*(PBEST-POS) ...
                + C2*rand(Np,nVar).*(repmat(REP.pos(h,:),Np,1)-POS);
        POS = POS + VEL;

        % Perform mutation
        POS = mutation(POS,gen,maxgen,Np,var_max,var_min,nVar,u_mut);

        % Check boundaries
        [POS,VEL] = checkBoundaries(POS,VEL,maxvel,var_max,var_min);

        % Evaluate the population
        POS_fit = fun(POS);
```

```
        % Update the repository
        REP = updateRepository(REP,POS,POS_fit,ngrid);
        if(size(REP.pos,1)>Nr)
            REP = deleteFromRepository(REP,size(REP.pos,1)-Nr,ngrid);
        end

        % Update the best positions found so far for each particle
        pos_best = dominates(POS_fit, PBEST_fit);
        best_pos = ~dominates(PBEST_fit, POS_fit);
        best_pos(rand(Np,1)>=0.5) = 0;
        if(sum(pos_best)>1)
           PBEST_fit(pos_best,:) = POS_fit(pos_best,:);
           PBEST(pos_best,:) = POS(pos_best,:);
        end
        if(sum(best_pos)>1)
           PBEST_fit(best_pos,:) = POS_fit(best_pos,:);
           PBEST(best_pos,:) = POS(best_pos,:);
        end

        % Plotting and verbose
        if(size(POS_fit,2)==2)
           figure(h_fig); delete(h_par); delete(h_rep);
           h_par = plot(POS_fit(:,1),POS_fit(:,2),'or'); hold on;
           h_rep = plot(REP.pos_fit(:,1),REP.pos_fit(:,2),'ok'); hold on;
           try
              set(gca,'xtick',REP.hypercube_limits(:,1)','ytick',REP.hypercube_limits(:,2)');
              axis([min(REP.hypercube_limits(:,1)) max(REP.hypercube_limits(:,1)) ...
                   min(REP.hypercube_limits(:,2)) max(REP.hypercube_limits(:,2))]);
           end
           if(isfield(MultiObj,'truePF'))
              try delete(h_pf); end
              h_pf = plot(MultiObj.truePF(:,1),MultiObj.truePF(:,2),'.','color',0.8.*ones(1,3));
hold on;
           end
           grid on; xlabel('f1'); ylabel('f2');
           drawnow;
           axis square;
        end
        if(size(POS_fit,2)==3)
           figure(h_fig); delete(h_par); delete(h_rep);
           h_par = plot3(POS_fit(:,1),POS_fit(:,2),POS_fit(:,3),'or'); hold on;
           h_rep = plot3(REP.pos_fit(:,1),REP.pos_fit(:,2),REP.pos_fit(:,3),'ok'); hold on;
           try

set(gca,'xtick',REP.hypercube_limits(:,1)','ytick',REP.hypercube_limits(:,2)','ztick',REP.hyper
cube_limits(:,3)');
              axis([min(REP.hypercube_limits(:,1)) max(REP.hypercube_limits(:,1)) ...
                   min(REP.hypercube_limits(:,2)) max(REP.hypercube_limits(:,2)) ...
                   min(REP.hypercube_limits(:,3)) max(REP.hypercube_limits(:,3))]);
```

```
            end
            if(isfield(MultiObj,'truePF'))
               try delete(h_pf); end
               h_pf =
plot3(MultiObj.truePF(:,1),MultiObj.truePF(:,2),MultiObj.truePF(:,3),'.','color',0.8.*ones(1,3)
); hold on;
            end
            grid on; xlabel('f1'); ylabel('f2'); zlabel('f3');
            drawnow;
            axis square;
         end
         display(['Generation #' num2str(gen) ' - Repository size: ' num2str(size(REP.pos,1))]);

         % Update generation and check for termination
         gen = gen + 1;
         if(gen>maxgen), stopCondition = true; end
      end
      hold off;
end

% Function that updates the repository given a new population and its
% fitness
function REP = updateRepository(REP,POS,POS_fit,ngrid)
      % Domination between particles
      DOMINATED  = checkDomination(POS_fit);
      REP.pos    = [REP.pos; POS(~DOMINATED,:)];
      REP.pos_fit= [REP.pos_fit; POS_fit(~DOMINATED,:)];
      % Domination between nondominated particles and the last repository
      DOMINATED  = checkDomination(REP.pos_fit);
      REP.pos_fit= REP.pos_fit(~DOMINATED,:);
      REP.pos    = REP.pos(~DOMINATED,:);
      % Updating the grid
      REP        = updateGrid(REP,ngrid);
end

% Function that corrects the positions and velocities of the particles that
% exceed the boundaries
function [POS,VEL] = checkBoundaries(POS,VEL,maxvel,var_max,var_min)
      % Useful matrices
      Np = size(POS,1);
      MAXLIM   = repmat(var_max(:)',Np,1);
      MINLIM   = repmat(var_min(:)',Np,1);
      MAXVEL   = repmat(maxvel(:)',Np,1);
      MINVEL   = repmat(-maxvel(:)',Np,1);

      % Correct positions and velocities
      VEL(VEL>MAXVEL) = MAXVEL(VEL>MAXVEL);
      VEL(VEL<MINVEL) = MINVEL(VEL<MINVEL);
      VEL(POS>MAXLIM) = (-1).*VEL(POS>MAXLIM);
      POS(POS>MAXLIM) = MAXLIM(POS>MAXLIM);
```

```
    VEL(POS<MINLIM) = (-1).*VEL(POS<MINLIM);
    POS(POS<MINLIM) = MINLIM(POS<MINLIM);
end

% Function for checking the domination between the population. It
% returns a vector that indicates if each particle is dominated (1) or not
function dom_vector = checkDomination(fitness)
    Np = size(fitness,1);
    dom_vector = zeros(Np,1);
    all_perm = nchoosek(1:Np,2);    % Possible permutations
    all_perm = [all_perm; [all_perm(:,2) all_perm(:,1)]];

    d = dominates(fitness(all_perm(:,1),:),fitness(all_perm(:,2),:));
    dominated_particles = unique(all_perm(d==1,2));
    dom_vector(dominated_particles) = 1;
end

% Function that returns 1 if x dominates y and 0 otherwise
function d = dominates(x,y)
    d = all(x<=y,2) & any(x<y,2);
end

% Function that updates the hypercube grid, the hypercube where belongs
% each particle and its quality based on the number of particles inside it
function REP = updateGrid(REP,ngrid)
    % Computing the limits of each hypercube
    ndim = size(REP.pos_fit,2);
    REP.hypercube_limits = zeros(ngrid+1,ndim);
    for dim = 1:1:ndim
        REP.hypercube_limits(:,dim) =
linspace(min(REP.pos_fit(:,dim)),max(REP.pos_fit(:,dim)),ngrid+1)';
    end

    % Computing where belongs each particle
    npar = size(REP.pos_fit,1);
    REP.grid_idx = zeros(npar,1);
    REP.grid_subidx = zeros(npar,ndim);
    for n = 1:1:npar
        idnames = [];
        for d = 1:1:ndim
            REP.grid_subidx(n,d) = find(REP.pos_fit(n,d)<=REP.hypercube_limits(:,d)',1,'first')-
1;
            if(REP.grid_subidx(n,d)==0), REP.grid_subidx(n,d) = 1; end
            idnames = [idnames ',' num2str(REP.grid_subidx(n,d))];
        end
        REP.grid_idx(n) = eval(['sub2ind(ngrid.*ones(1,ndim)' idnames ');']);
    end

    % Quality based on the number of particles in each hypercube
    REP.quality = zeros(ngrid,2);
```

```
    ids = unique(REP.grid_idx);
    for i = 1:length(ids)
        REP.quality(i,1) = ids(i);  % First, the hypercube's identifier
        REP.quality(i,2) = 10/sum(REP.grid_idx==ids(i)); % Next, its quality
    end
end

% Function that selects the leader performing a roulette wheel selection
% based on the quality of each hypercube
function selected = selectLeader(REP)
    % Roulette wheel
    prob    = cumsum(REP.quality(:,2));     % Cumulated probs
    sel_hyp = REP.quality(find(rand(1,1)*max(prob)<=prob,1,'first'),1); % Selected hypercube

    % Select the index leader as a random selection inside that hypercube
    idx      = 1:1:length(REP.grid_idx);
    selected = idx(REP.grid_idx==sel_hyp);
    selected = selected(randi(length(selected)));
end

% Function that deletes an excess of particles inside the repository using
% crowding distances
function REP = deleteFromRepository(REP,n_extra,ngrid)
    % Compute the crowding distances
    crowding = zeros(size(REP.pos,1),1);
    for m = 1:1:size(REP.pos_fit,2)
        [m_fit,idx] = sort(REP.pos_fit(:,m),'ascend');
        m_up     = [m_fit(2:end); Inf];
        m_down   = [Inf; m_fit(1:end-1)];
        distance = (m_up-m_down)./(max(m_fit)-min(m_fit));
        [~,idx]  = sort(idx,'ascend');
        crowding = crowding + distance(idx);
    end
    crowding(isnan(crowding)) = Inf;

    % Delete the extra particles with the smallest crowding distances
    [~,del_idx] = sort(crowding,'ascend');
    del_idx = del_idx(1:n_extra);
    REP.pos(del_idx,:) = [];
    REP.pos_fit(del_idx,:) = [];
    REP = updateGrid(REP,ngrid);
end

% Function that performs the mutation of the particles depending on the
% current generation
function POS = mutation(POS,gen,maxgen,Np,var_max,var_min,nVar,u_mut)
    % Sub-divide the swarm in three parts [2]
    fract     = Np/3 - floor(Np/3);
    if(fract<0.5), sub_sizes =[ceil(Np/3) round(Np/3) round(Np/3)];
    else           sub_sizes =[round(Np/3) round(Np/3) floor(Np/3)];
```

```
    end
    cum_sizes = cumsum(sub_sizes);

    % First part: no mutation
    % Second part: uniform mutation
    nmut = round(u_mut*sub_sizes(2));
    if(nmut>0)
        idx = cum_sizes(1) + randperm(sub_sizes(2),nmut);
        POS(idx,:) = repmat((var_max-var_min)',nmut,1).*rand(nmut,nVar) +
repmat(var_min',nmut,1);
    end

    % Third part: non-uniform mutation
    per_mut = (1-gen/maxgen)^(5*nVar);     % Percentage of mutation
    nmut    = round(per_mut*sub_sizes(3));
    if(nmut>0)
        idx = cum_sizes(2) + randperm(sub_sizes(3),nmut);
        POS(idx,:) = repmat((var_max-var_min)',nmut,1).*rand(nmut,nVar) +
repmat(var_min',nmut,1);
    end
end
```

MachiningP Code :

```
case 'Poloni'
        GW = @(x,y,z,w) 0.199819 - 0.141653.*x - 0.001124.*y - 0.004140.*z – 0.007017.*w+
0.034120.*x.*x + 0.000709.*y.*y - 0.000056.*z.*z + 0.000842.*w.*w + 0.000200.*x.*y+
0.00250.*x.*z - 0.002667.*x.*w + 0.000021.*y.*z - 0.000733.*y.*w + 0.000158.*z.*w;
        TA = @(x,y,z,w) - 3319 + 1990.*x + 42.88607.*y - 13.94.*z + 459.63028.*w –
701.29733.*x.*x - 14.35993.*y.*y + 0.771807.*z.*z - 24.75104.*w.*w + 73.54000.*x.*y +
18.15750.*x.*z - 5.75111.*y.*w - 5.61792.*z.*w;
        %
        MultiObj.fun = @(x) [GW(x(:,1),x(:,2),x(:,3),x(:,4)), TA(x(:,1),x(:,2),x(:,3),x(:,4))];
        MultiObj.nVar = 4;
        MultiObj.var_min = [1.5 2.5 10 6 ];
        MultiObj.var_max = [2.0 5.5 18 9];
end

% Parameters
params.Np = 300;        % Population size
params.Nr = 300;        % Repository size
params.maxgen =250;     % Maximum number of generations
params.W = 0.4;         % Inertia weight
params.C1 = 1;          % Individual confidence factor
params.C2 = 1;          % Swarm confidence factor
params.ngrid = 20;      % Number of grids in each dimension
params.maxvel = 5;      % Maxmium vel in percentage
params.u_mut = 0.5;     % Uniform mutation percentage
```

```
% MOPSO
REP = MOPSO(params,MultiObj);

% Display info
display('Repository fitnez values are stored in REP.pos_fit');
display('Repository particles positions are store in REP.pos');
```

4.1.4 Statistical analysis

Software known as design expert 13 was employed to analyse data, create quadratic models, and analyse the effect of process factors. The model F-value of 198.65 implies the model is significant. There is only a 0.01% chance that an F-value this large could occur due to noise. P-values less than 0.0500 indicate model terms are significant. In this case A, B, C, D, AC, AD, BD, CD, A^2, B^2, C^2, D^2 are significant model terms. Values greater than 0.1000 indicate the model terms are not significant. If there are many insignificant model terms (not counting those required to support hierarchy), model reduction may improve model.

The lack of fit F-value of 0.71 implies the lack of fit is not significant relative to the pure error. There is a 70.17% chance that a lack of fit F-value this large could occur due to noise. Non-significant lack of fit is good -- we want the model to fit.

Table 4.5: ANOVA table for kerf deviation (KD)

Source	Sum of Squares	df	Mean Square	F-value	p-value	
Model	0.0001	14	0.0000	198.65	< 0.0001	significant
A-Pulse width	0.0000	1	0.0000	577.25	<0.0001	
B-Pulse energy	2.613E-06	1	2.613E-06	48.60	<0.0001	
C-Cutting speed	6.075E-07	1	6.075E-07	11.30	0.0047	
D-Gas pressure	1.470E-06	1	1.470E-06	27.34	0.0001	
AB	2.250E-08	1	2.250E-08	0.4184	0.5282	
AC	0.0000	1	0.0000	464.91	<0.0001	
AD	4.000E-06	1	4.000E-06	74.39	<0.0001	
BC	6.250E-08	1	6.250E-08	1.16	0.2992	
BD	0.0000	1	0.0000	202.51	<0.0001	
CD	3.610E-06	1	3.610E-06	67.13	<0.0001	
A^2	0.0000	1	0.0000	548.55	<0.0001	
B^2	0.0000	1	0.0000	306.87	<0.0001	
C^2	5.167E-06	1	5.167E-06	96.08	<0.0001	
D^2	0.0000	1	0.0000	433.17	<0.0001	
Residual	7.528E-07	14	5.377E-08			
Lack of Fit	4.808E-07	10	4.808E-08	0.7071	0.7017	not significant
Pure Error	2.720E-07	4	6.800E-08			
Cor Total	0.0002	28				

Table 4.6: Model fit statistics of kerf deviation (KD)

Std. Dev	0.0002
Mean	0.0145
C.V%	1.60
R^2	0.9950
Adjusted R^2	0.9900
Predicted R^2	0.9787
Adeq. Precision	57.7373

The predicted R^2 of 0.9787 is in reasonable agreement with the adjusted R^2 of 0.9900; i.e., the difference is less than 0.2. Adeq. precision measures the signal to noise ratio. A ratio greater than 4 is desirable. Model ratio of 57.737 indicates an adequate signal. This model can be used to navigate the design space.

Table 4.7: Coefficients in terms of coded Factors

Factor	Coefficient Estimate	df	Standard Error	95% CI Low	95% CI High	VIF
Intercept	0.0126	1	0.0001	0.0123	0.0128	
A-Pulse width	-0.0016	1	0.0001	-0.0018	-0.0015	1.0000
B-Pulse energy	-0.0005	1	0.0001	-0.0006	-0.0003	1.0000
C-Cutting Speed	-0.0002	1	0.0001	-0.0004	-0.0001	1.0000
D-Gas Pressure	0.0004	1	0.0001	0.0002	0.0005	1.0000
AB	0.0001	1	0.0001	-0.0002	0.0003	1.0000
AC	0.0025	1	0.0001	0.0023	0.0027	1.0000
AD	-0.0010	1	0.0001	-0.0012	-0.0008	1.0000
BC	0.0001	1	0.0001	-0.0001	0.0004	1.0000
BD	-0.0017	1	0.0001	-0.0019	-0.0014	1.0000
CD	0.0010	1	0.0001	0.0007	0.0012	1.0000
A^2	0.0021	1	0.0001	0.0019	0.0023	1.08
B^2	0.0016	1	0.0001	0.0014	0.0018	1.08
C^2	-0.0009	1	0.0001	-0.0011	-0.0007	1.08
D^2	0.0019	1	0.0001	0.0017	0.0021	1.08

The coefficient estimate represents the expected change in response per unit change in factor value when all remaining factors are held constant. The intercept in an orthogonal design is the overall average response of all the runs. The coefficients are adjustments around that average based on the factor settings. When the factors are orthogonal the VIFs are 1; VIFs greater than 1 indicate multi-collinearity, the higher the VIF the more severe the correlation of factors. As a rough rule, VIFs less than 10 are tolerable.

Table 4.8: Final equation in terms of actual factors:

Kerf deviation	=
+0.199819	
-0.141653	Pulse width
-0.001124	Pulse energy
-0.004140	Cutting speed
-0.007017	Gas pressure
+0.000200	Pulse width * Pulse energy
+0.002500	Pulse width * Cutting speed
-0.002667	Pulse width * Gas pressure
+0.000021	Pulse energy * Cutting speed
-0.000733	Pulse energy * Gas pressure
+0.000158	Cutting speed * Gas pressure
+0.034120	Pulse width^2
+0.000709	Pulse energy^2
-0.000056	Cutting speed^2
+0.000842	Gas pressure^2

The equation in terms of actual factors can be used to make predictions about the response for given levels of each factor. Here, the levels should be specified in the original units for each factor. This equation should not be used to determine the relative impact of each factor because the coefficients are scaled to accommodate the units of each factor and the intercept is not at the centre of the design space.

Table 4.9: ANOVA table for MRR

Source	Sum of Squares	df	Mean Square	F-value	p-value	
Model	62024.93	12	5168.74	33.77	<0.0001	significant
A-Pulse width	5346.74	1	5346.74	34.94	<0.0001	
B-Pulse energy	4970.69	1	4970.69	32.48	<0.0001	
C-Cutting Speed	1394.93	1	1394.93	9.11	0.0081	
D-Gas Pressure	4769.25	1	4769.25	31.16	<0.0001	
AB	3042.07	1	3042.07	19.88	0.0004	
AC	1318.78	1	1318.78	8.62	0.0097	
BD	669.77	1	669.77	4.38	0.0527	
CD	4544.78	1	4544.78	29.70	<0.0001	
A^2	12461.60	1	12461.60	81.43	<0.0001	
B^2	6771.41	1	6771.41	44.25	<0.0001	
C^2	989.16	1	989.16	6.46	0.0217	
D^2	20116.91	1	20116.91	131.45	<0.0001	
Residual	2448.61	16	153.04			
Lack of Fit	2291.66	12	190.97	4.87	0.0695	not significant
Pure Error	156.95	4	39.24			
Cor Total	64473.54	28				

The model F-value of 33.77 implies the model is significant. There is only a 0.01% chance that an F-value this large could occur due to noise. P-values less than 0.0500 indicate model terms are significant. In this case A, B, C, D, AB, AC, CD, A^2, B^2, C^2, D^2 are significant model terms. Values greater than 0.1000 indicate the model terms are not significant. If there are many insignificant model terms (not counting those required to support hierarchy), model reduction may improve model. The lack of fit F-value of 4.87 implies there is a 6.95% chance that a lack of fit F-value this large could occur due to noise. Lack of fit is bad -- we want the model to fit. This relatively low probability (<10%) is troubling.

Table 4.10: Model fit statistics of MRR

Std. deviation	12.37
Mean	117.21
C.V%	10.55
R^2	0.9620
Adjusted R^2	0.9335
Predicted R^2	0.8483
Adeq Precision	18.3199

Table 4.11: Coefficients in terms of coded factors

Factor	Coefficient Estimate	df	Standard Error	95% CI Low	95% CI High	VIF
Intercept	166.65	1	5.53	154.92	178.38	
A-Pulse width	21.11	1	3.57	13.54	28.68	1.0000
B-Pulse energy	20.35	1	3.57	12.78	27.92	1.0000
C-Cutting Speed	-10.78	1	3.57	-18.35	-3.21	1.0000
D-Gas Pressure	-19.94	1	3.57	-27.51	-12.37	1.0000
AB	27.58	1	6.19	14.46	40.69	1.0000
AC	18.16	1	6.19	5.04	31.27	1.0000
BD	-12.94	1	6.19	-26.05	0.1725	1.0000
CD	-33.71	1	6.19	-46.82	-20.59	1.0000
A^2	-43.83	1	4.86	-54.13	-33.53	1.08
B^2	-32.31	1	4.86	-42.61	-22.01	1.08
C^2	12.35	1	4.86	2.05	22.65	1.08
D^2	-55.69	1	4.86	-65.99	-45.39	1.08

The coefficient estimate represents the expected change in response per unit change in factor value when all remaining factors are held constant. The intercept in an orthogonal design is the overall average response of all the runs. The coefficients are adjustments around that

average based on the factor settings. When the factors are orthogonal the VIFs are 1; VIFs greater than 1 indicate multi-collinearity, the higher the VIF the more severe the correlation of factors. As a rough rule, VIFs less than 10 are tolerable.

Table 4.12: Final equation in terms of actual factors

MRR	=
-3319.19800	
+1990.60900	Pulse width
+42.88607	Pulse energy
-13.94727	Cutting speed
+459.63028	Gas pressure
+73.54000	Pulse width * Pulse energy
+18.15750	Pulse width * Cutting speed
-5.75111	Pulse energy * Gas pressure
-5.61792	Cutting speed * Gas pressure
-701.29733	Pulse width^2
-14.35993	Pulse energy^2
+0.771807	Cutting speed^2
-24.75104	Gas pressure^2

The equation in terms of actual factors can be used to make predictions about the response for given levels of each factor. Here, the levels should be specified in the original units for each factor. This equation should not be used to determine the relative impact of each factor because the coefficients are scaled to accommodate the units of each factor and the intercept is not at the centre of the design space.

4.1.5 Model and analysis

The second order regression models have been developed for the analysis of quality characteristics such as KD, and MRR. The efficiency and confidence level of second order regression model is high compared to first order regression model. Keeping this fact in mind, the second order regression models for different responses have been developed by using experimental data. The general equation for the second order regression model is given as-

$$Y - \beta o + \sum_{i=1}^{n} \beta_i X_i + \sum_{i=1}^{n} \beta_{ii} X_i^{\,2} + \sum_{i=1}^{n} \sum_{i=1+1}^{n} \beta_{ij} X_i X_j$$

The regression coefficients or partial regression coefficients in this equation are denoted by the letters β. Because these coefficients demonstrate the average change in Y caused by changing X_i while other factors remain fixed, they are known as partial regression coefficients. By minimising the mean square error or by applying the least squares approach, the values of these coefficients can be determined.

From equation, X_i (i = 1, 2, 3...n), are the input (Independent) or process variables and n is the total number of the input variables or control factors (independent). The *Y* is the output/response. For finding of the values of regression coefficients β's, Design Expert 13 software has been used. The second-order response surface models or regression models for KD and MRR are-

KD (kerf width deviation) = 0.199819 - 0.141653 A - 0.001124 B - 0.004140 C – 0.007017 D + 0.034120 A^2 + 0.000709B^2 - 0.000056C^2 + 0.000842 D^2 + 0.000200 AB + 0.00250AC - 0.002667 AD + 0.000021BC - 0.000733 BD + 0.000158 CD. -------Eq$^{n.}$ (1)

MRR (Material removal rate) = - 3319 + 1990 A + 42.88607 B - 13.94 C + 459.63028 D – 701.29733 A^2 - 14.35993 B^2 + 0.771807 C^2 - 24.75104 D^2 + 73.54000 AB + 18.15750 AC - 5.75111 BD - 5.61792 CD. -------Eq$^{n.}$ (2)

4.2 Multi objective optimisation on laser cutting parameters of Ti-6Al-4V super alloy

A single optimal solution is generated by a single-response optimization process. However, most multi-response problems principally yield a set of optimal solutions instead of a single optimal solution. Ultimately, the multi-response optimization algorithm functions in maintaining a balance between two or more responses in terms of quality and productivity. The present work considers two responses, namely, KD (quality consideration) and MRR (productivity consideration). We observed that KD increases with increasing MRR. Given the production purpose, the optimal combination of parameter levels should produce the maximum MRR and the minimum KD. Particle swamp optimisation and response surface methodology is an improved and unique numerical optimization technique used in the industry for the optimization of multiple quality characteristics. The optimization module searches for a combination of process parameter levels that simultaneously satisfy the requirements placed on each of the responses and process parameters. The developed KD and MRR model equations (1 and 2) were simultaneously solved to yield the optimal process variables.

4.2.1 Particle swarm optimization (optimal solution) and quality improvement

The hybrid approach of regression and particle swarm optimization has been used for the optimization of kerf deviation (KD), material removal rate (MRR). The second order regression model developed for the kerf deviation (KD); material removal rate (MRR) has been utilized for the particle swarm optimization. These models have been used as objective functions for the optimization. The PSO optimization coding has been done in MATLAB and after selecting the critical parameters, the program has been run. The optimization result is shown the relationship between the kerf deviation (KD) and MRR (material removal rate) represent with the help of figure 4.2. The selected operator setting is shown in table 4.13.

Table 4.13: Critical operating parameters

Parameter setting for PSO				
C_1	C_2	W	Population size	Maximum Iteration
1	1	0.4	300	250

The optimized values by PSO are 0.00971 mm for KD (kerf deviation) and 192.9321 mg/min for MRR (material removal rate). The PSO optimization gives the predicted values for the improvement at optimum control parameter setting for individual quality characteristics. The optimize value found for kerf deviation (KD) and material removal rate (MRR) at optimum setting as pulse width is 2 ms, pulse energy is 4.64 J, cutting speed is 10 mm/min and gas pressure is 8.17 kg/cm^2.

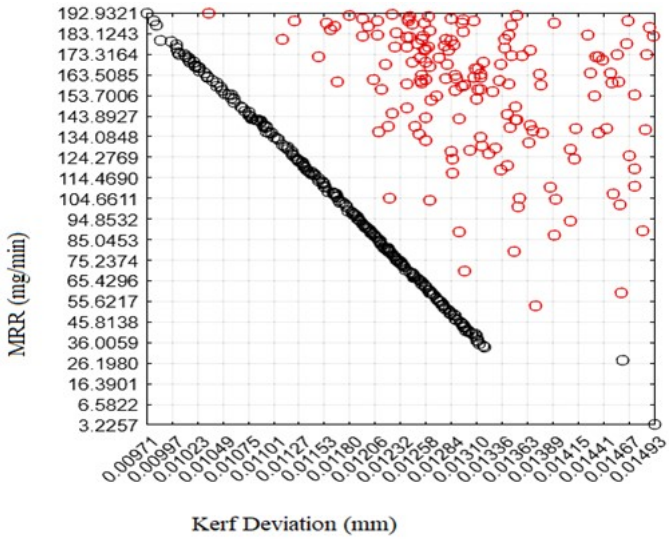

Figure 4.2: MOPSO optimisation of kerf deviation (KD) and material removal rate (MRR)

And the graph in figure 4.2 shows the relation between the kerf deviation (KD) and the material removal rate (MRR). The graph shows that the depth of the MRR is maximum when the KD is minimum and thus the relation is inversely proportional, between the KD and the MRR.

4.2.2 RSM optimisation of laser cutting ofTi-6Al-4V material

Response surface methodology has been used for the optimization of kerf deviation (KD), material removal rate (MRR). The second order regression model developed for the kerf deviation (KD); material removal rate (MRR) has been utilized for the response surface methodology. These models have been used as objective functions for the optimization. In the study, optimization of KD and MRR has been done during Nd: YAG laser beam cutting of Ti-6Al-4V–sheet using RSM where response optimizer module of design expert 13 software is used. The goal set, lower limits, upper limits, weights used, and importance of the factors given are shown in table 4.14.

Table 4.14: Target and goal settings in RSM optimisation

Name	Goal	Lower Limit	Upper Limit	Lower Weight	Upper Weight	Importance
A: Pulse width	is in range	1.5	2	1	1	3
B: Pulse energy	is in range	2.5	5.5	1	1	3
C: Cutting Speed	is in range	10	18	1	1	3
D: Gas Pressure	is in range	6	9	1	1	3
kerf Deviation (KD)	minimize	0.0101	0.0197	1	1	3
Material Removal Rate (MRR)	maximize	28.37	174.55	1	1	3

The minimum and maximum levels were provided for each parameter included. The goals used for kerf deviation and MRR are “minimize” and “maximize”, and the goal used for the factors or process parameters is “within range”. A weight was assigned to each goal to adjust the shape of its desirability function. The importance of each goal was changed in relation to other goals. optimization was carried out for a combination of goals.

Figure 4.3 represents the multi objective optimisation analysis for minimizing the kerf deviation (KD) and maximising the MRR based on the developed mathematical model as

given by Eq. (1) and Eq. (2). For linear desirability function (d) the value of weight is considered 1 for kerf deviation (KD) and 1 for material removal rate (MRR), from table 4.14. This figure 4.3 represents the simultaneous optimization of both quality characteristics in one setting.

Table 4.15: Solution of optimal data of RSM optimisation:

Solution	Pulse width (ms)	Pulse energy (J)	Cutting speed (mm/min)	Gas pressure (kg/cm^2)	MRR (mg/min) Fit	KD (mm) Fit	CD*
1	1.822	5.020	10.104	8.176	181.063	0.011	1

*CD mean composite desirability.

The predicted optimum values (given in Table 4.15) of kerf deviation (KD) and material removal rate (MRR) during multi objective optimisation are 0.011 mm and 181.063 mg/min, respectively at Pulse width 1.822 ms, pulse energy 5.020 J, cutting speed 10.104 mm/min and gas pressure 8.176 kg/cm^2.

Table 4.16: Optimise input parameters for KD and MRR

Variable	Setting
Pulse width(ms)	1.822
Pulse energy(J)	5.020
Cutting speed(mm/min)	10.104
Gas pressure (kg/cm^2)	8.176

The value of composite desirability, (D) is 1.000 and individual desirability, (d) for kerf deviation (KD) and material removal rate (MRR) are 1.0000 and 1.0000.

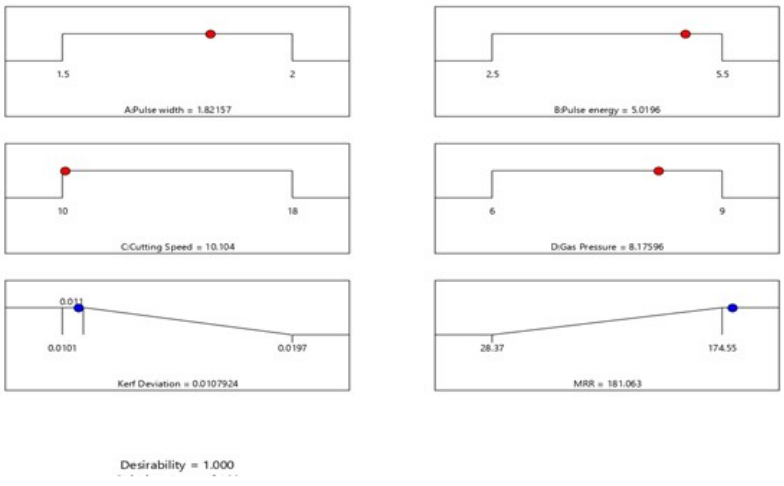

Figure 4.3: Multi parameter multi objective optimisation plot after the RSM optimisation during Nd: YAG laser beam cutting of Ti-6Al-4V.

Now, by multi objective technique of response surface methodology optimisation technique, the goal is to achieve the target minimum kerf deviation (KD) when maximize the cutting speed (mm/min) with all the other parameter like pulse width, pulse energy and gas pressure remain in between their range value. Table 4.17, shown below shows goal of maximise cutting speed when kerf deviation (KD) targeted at minimised value.

Table 4.17 :Constraints of maximise scanning speed when kerf deviation fixed at minimise value

Name	Goal	Lower Limit	Upper Limit	Lower Weight	Upper Weight	Importance
A: Pulse width	is in range	1.5	2	1	1	3
B: Pulse energy	is in range	2.5	5.5	1	1	3
C: Cutting Speed	maximize	10	18	1	1	3
D: Gas Pressure	is in range	6	9	1	1	3
kerf deviation	is target = 0.011	0.0101	0.0197	1	1	3
MRR	none	28.37	174.55	1	1	3

RSM based multi objective optimisation was performed to get the best combination of parameters to get the desired value for targeted minimised kerf width (KD) when cutting speed (mm/min) is set to the maximised value. Results after optimising in the design expert given in the table. Table 4.18 shows that when kerf deviation targeted at a minimised value of 0.011 mm value, the optimise maximise value of cutting speed will be 18 mm/min, when importance is set to +++ and desirability is 0.967.

Table 4.18: Solution of optimal values when cutting speed is fixed at maximum value

No	Pulse width	Pulse energy	Cutting speed	Gas pressure	Kerf deviation	Desirability	
1	1.66908	3.79234	18.000	6.76605	0.011006	0.967	Selected

Simiarly, some other minimised target value was choosen for kerf deviation , maximised MRR values are also choosen and taking scanning speed as maximise condition, the following table 4.19 has been made.

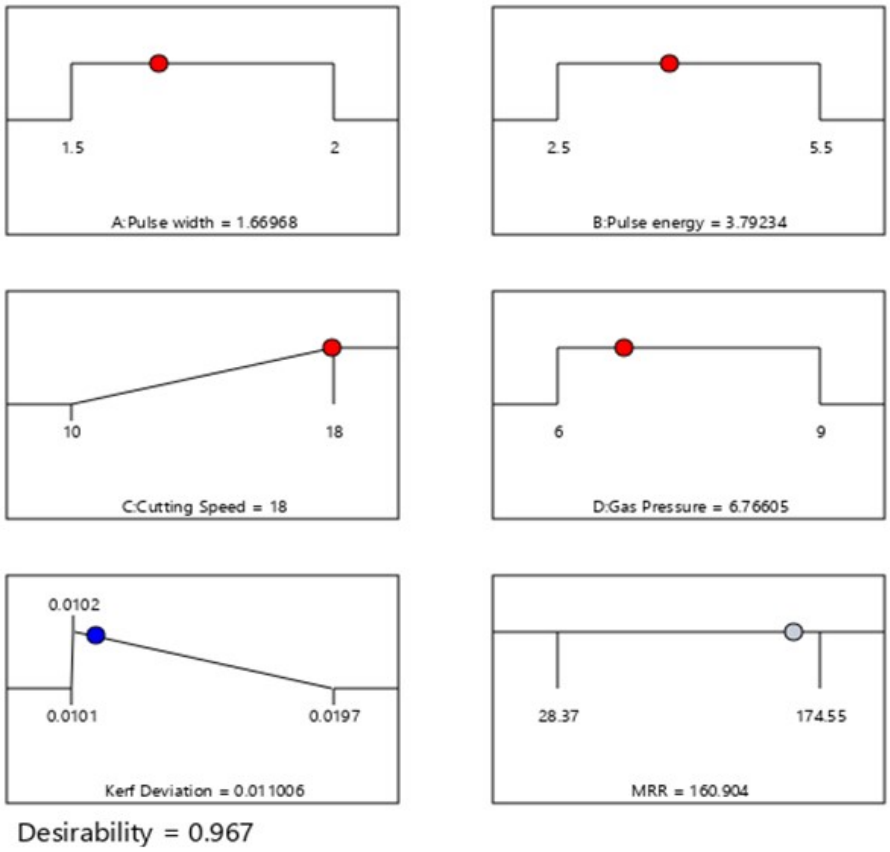

Figure 4.4: When kerf deviation is fixed at minimum of 0.011 mm and cutting speed is set as maximise value.

Table 4.19: Percentage of deviation from required data for Ti-6Al-4V superalloy

Required		Predicted		Percentage of Deviation From Requirement	
Kerf deviation (mm)	MRR (mm/min)	Kerf deviation (mm)	MRR (mm/min)	Kerf deviation (%)	MRR (%)
0.0112	171.5	0.0113	170.303	0.892	0.69
0.012	169	0.012556	168.153	0.0556	0.501
0.0125	165	0.012752	164.618	2.016	0.231
0.0129	164	0.013261	163.93	2.79	0.042
0.0131	163	0.013297	162.877	1.51	0.04
0.0133	162	0.013382	161.802	0.61	0.123

4.2.3 Comparison of two optimization

Table 4.20: Comparison of predicted optimal conditions of RSM and PSO optimisation

Controlling Factor	Unit	PSO optimal Set	RSM optimal Set
Pulse width	ms	2	1.822
Pulse energy	J	4.64	5.020
Cutting speed	mm/min	10	10.104
Gas pressure	Kg/cm^2	8.17	8.176

Table 4.21: Comparison of predicted optimal conditions responses of RSM and PSO optimisation

Responses	PSO optimal set	RSM optimal set
Kerf deviation (mm)	0.00971	0.011
Material removal rate (mg/min)	192.9321	181.063

CHAPTER – 5

MACHINING OF A653 GALVANIZED STEEL SHEET

5.1 Modelling for laser cutting of A653 galvanized steel sheet

5.1.1 Material properties of A653 galvanized steel material

The galvanized steel sheet (thickness 0.5 mm) with grade A653 was used by Vipin Kumar Sharma and Vinod Kumar et. al. in their paper "Study on material transfer and surface properties during fiber laser cutting of A653 galvanized steel sheet". Galvanized steel sheet basically a carbon steel sheet which is coated with zinc on both sides. When corrosion resistance is required, it is frequently used since it is simple to fabricate and less expensive than stainless steel. Galvanized steel sheet is used for playground equipment, air conditioning systems, steel water tanks, electrical boxes, automobile parts, roofing, and safety railings.

Chemical Properties:

Work material contained C: 0.065%, Mn: 0.407%, P: 0.147%, S: 0.099%, Al: 0.302%, Cu: 0.029%, Ni: 0.128%, Cr: 0.055%, Ti: 0.022% and Fe: 97.7%.

Mechanical properties:

Atmospheric corrosion resistance of zinc or zinc-iron alloy-coated sheet products is a direct function of coating thickness. A thinner coating will result in almost linearly reduced corrosion performance of the coating. Heavier galvanized coatings perform adequately in bold atmospheric exposure whereas the lighter coatings are often further coated with paint or a similar barrier coating for increased corrosion resistance.

5.1.2 Experimental plan of laser cutting of A653 galvanised steel sheet material

Vipin Kumar Sharma and Vinod Kumar et. al. in "Study on material transfer and surface properties during fiber laser cutting of A653 galvanized steel sheet" performed the experiments on $40\ mm * 40\ mm * 0.5\ mm$ thickness A653 galvanised steel sheet. The designed the experiments with Box-Behnken experimental design considering the process parameters, including gas pressure, power, frequency, and feed with each in three levels:

Table 5.1: Level value of each process parameter

Process Parameter (unit)	Selected symbol	Low value (-1)	Middle value (0)	High value (+1)
Gas Pressure (kgf/cm^2)	A	6	8	10
Power (J)	B	80	90	100
Frequency (Hz)	C	50	70	90
Feed (mm/min)	D	150	300	450

Levels of input parameters range from -1 to +1, hence level -1 is indicating the lower values and level +1 indicating the higher values.

Vipin Kumar Sharma and Vinod Kumar et. al. has done the experiments on 100 W fiber laser beam machine (Scantech Laser Pvt. Ltd., Patriot 100) 1 ft* 1ft CNC workable fully integrated with hardware and software, parameters kept constant during machining of the material work material, assist gas type oxygen, diameter nozzle 1 mm and distance between workpiece and nozzle tip (standoff distance) maintained at 2 mm, continuous mode laser beam machine was used. With 10* magnification, Nikon profile Projector (model V 10 A) was used whose least count is 0.001 mm.

Outcome of pilot experiments was used as basis for the selection of process parameters with various levels for the plan of main experiments. A one-factor-at-a-time approach was used to conduct pilot experiments. The design of various experimentation runs was done using Box–Behnken design:

Table 5.2: Box - Behnken Design layout and Results:

Gas Pressure (Kgf/cm^2)	Power (w)	Frequency (Hz)	Feed (mm/min)	KD (mm)	MRR (mg/min)
8	100	90	300	0.267	545
10	90	70	150	0.385	338
6	90	70	150	0.389	219
10	100	70	300	0.367	511
6	100	70	300	0.375	602
8	80	90	300	0.238	611
6	90	50	300	0.326	615
6	90	90	300	0.302	321
8	90	70	300	0.391	541
8	100	70	450	0.281	802
10	90	90	300	0.253	693

10	90	50	300	0.416	490
8	90	70	300	0.378	505
10	90	70	450	0.348	760
8	90	50	150	0.325	267
8	90	70	300	0.372	530
8	80	50	300	0.371	314
10	80	70	300	0.367	685
6	90	70	450	0.351	664
8	100	50	300	0.297	608
8	90	90	450	0.248	667
8	100	70	150	0.348	278
8	90	70	300	0.397	520
6	80	70	300	0.387	334
8	90	90	150	0.272	280
8	90	70	300	0.383	550
8	80	70	450	0.278	487
8	90	50	450	0.324	675
8	80	70	150	0.357	267

Table 5.2 shows the experiment design and outputs after laser cutting of A653 galvanised steel sheet using Response Surface Methodology. Twenty-nine experiments were performed based on BBD design taking four control factors at three levels as given in Table 5.1.

5.1.3 Software identification:

Design Expert: Design Expert 13 product version package is used to evaluate the effects of process parameters, data analysis, and quadratic model generation.

MATLAB: MATLAB was used to implement the PSO algorithm. Pseudo code of MOPSO algorithm is given below –

MOPSO code:

```
% ------------------------------------------------------------------------ %
% Function MOPSO performs a Multi-Objective Particle Swarm Optimization   %
% over continuous functions.                                               %
%                                                                          %
%   Input parameters:                                                      %
%       - params:   Struct that contains the customized parameters.        %
%           * params.Np:        Number of particles.                       %
%           * params.Nr:        Repository size (in particles).            %
%           * params.maxgen:    Maximum number of generations.             %
%           * params.W:         Inertia coefficient.                       %
%           * params.C1:        Personal confidence factor.                %
%           * params.C2:        Swarm confidence factor.                   %
%           * params.ngrid:     Number of hypercubes in each dimension.    %
%           * params.maxvel:    Maximum velocity (search space percentage)%
```

```
%          * params.u_mut:     Uniform mutation percentage.              %
%       - MultiObj: Struct that contains the parameters relative to the   %
%                   optimization functions.                               %
%          * MultiObj.fun:     Anonymous multi-obj function to minimize. %
%          * MultiObj.nVar:    Number of variables.                      %
%          * MultiObj.var_min: Vector that indicates the minimum values  %
%                              of the search space in each dimension.    %
%          * MultiObj.var_max: Same than 'var_min' with the maxima.      %
% ------------------------------------------------------------------------ %
%   For an example of use, run 'example.m'.                                %
% ------------------------------------------------------------------------ %
%   Author:  Victor Martinez Cagigal                                       %
%   Date:    17/03/2017                                                    %
%   E-mail:  vicmarcag (at) gmail (dot) com                                %
%   Version: 1.1                                                          %
%   Log:                                                                  %
%          - 1.0:  Initial version without mutation [1] (15/03/2017).     %
%          - 1.1:  Crowding and mutation are implemented [2].             %
% ------------------------------------------------------------------------ %
%   References:                                                           %
%    [1]Coello, C. A. C., Pulido, G. T., & Lechuga, M. S. (2004). Handling%
%       multiple objectives with particle swarm optimization. IEEE Tran-  %
%       sactions on evolutionary computation, 8(3), 256-279.              %
%                                                                         %
%    [2]Sierra, M. R., & Coello, C. A. C. (2005, March). Improving PSO-   %
%       based multi-objective optimization using crowding, mutation and ?-%
%       dominance. In International Conference on Evolutionary Multi-Crite%
%       rion Optimization (pp. 505-519). Springer Berlin Heidelberg.      %
% ------------------------------------------------------------------------ %
function REP = MOPSO(params,MultiObj)

    % Parameters
    Np      = params.Np;
    Nr      = params.Nr;
    maxgen  = params.maxgen;
    W       = params.W;
    C1      = params.C1;
    C2      = params.C2;
    ngrid   = params.ngrid;
    maxvel  = params.maxvel;
    u_mut   = params.u_mut;
    fun     = MultiObj.fun;
    nVar    = MultiObj.nVar;
    var_min = MultiObj.var_min(:);
    var_max = MultiObj.var_max(:);

    % Initialization
    POS = repmat((var_max-var_min)',Np,1).*rand(Np,nVar) + repmat(var_min',Np,1);
    VEL = zeros(Np,nVar);
    POS_fit  = fun(POS);
```

```
    if size(POS,1) ~= size(POS_fit,1)
       warning(['The objective function is badly programmed. It is not returning' ...
          'a value for each particle, please check it.']);
    end
    PBEST    = POS;
    PBEST_fit= POS_fit;
    DOMINATED= checkDomination(POS_fit);
    REP.pos  = POS(~DOMINATED,:);
    REP.pos_fit = POS_fit(~DOMINATED,:);
    REP      = updateGrid(REP,ngrid);
    maxvel   = (var_max-var_min).*maxvel./100;
    gen      = 1;

    % Plotting and verbose
    if(size(POS_fit,2)==2)
       h_fig = figure(1);
       h_par = plot(POS_fit(:,1),POS_fit(:,2),'or'); hold on;
       h_rep = plot(REP.pos_fit(:,1),REP.pos_fit(:,2),'ok'); hold on;
       try
          set(gca,'xtick',REP.hypercube_limits(:,1)','ytick',REP.hypercube_limits(:,2)');
          axis([min(REP.hypercube_limits(:,1)) max(REP.hypercube_limits(:,1)) ...
               min(REP.hypercube_limits(:,2)) max(REP.hypercube_limits(:,2))]);
          grid on; xlabel('f1'); ylabel('f2');
       end
       drawnow;
    end
    if(size(POS_fit,2)==3)
       h_fig = figure(1);
       h_par = plot3(POS_fit(:,1),POS_fit(:,2),POS_fit(:,3),'or'); hold on;
       h_rep = plot3(REP.pos_fit(:,1),REP.pos_fit(:,2),REP.pos_fit(:,3),'ok'); hold on;
       try

set(gca,'xtick',REP.hypercube_limits(:,1)','ytick',REP.hypercube_limits(:,2)','ztick',REP.hyper
cube_limits(:,3)');
             axis([min(REP.hypercube_limits(:,1)) max(REP.hypercube_limits(:,1)) ...
                  min(REP.hypercube_limits(:,2)) max(REP.hypercube_limits(:,2))]);
       end
       grid on; xlabel('f1'); ylabel('f2'); zlabel('f3');
       drawnow;
       axis square;
    end
    display(['Generation #0 - Repository size: ' num2str(size(REP.pos,1))]);

    % Main MPSO loop
    stopCondition = false;
    while ~stopCondition

       % Select leader
       h = selectLeader(REP);
```

```
        % Update speeds and positions
        VEL = W.*VEL + C1*rand(Np,nVar).*(PBEST-POS) ...
                + C2*rand(Np,nVar).*(repmat(REP.pos(h,:),Np,1)-POS);
        POS = POS + VEL;

        % Perform mutation
        POS = mutation(POS,gen,maxgen,Np,var_max,var_min,nVar,u_mut);

        % Check boundaries
        [POS,VEL] = checkBoundaries(POS,VEL,maxvel,var_max,var_min);

        % Evaluate the population
        POS_fit = fun(POS);

        % Update the repository
        REP = updateRepository(REP,POS,POS_fit,ngrid);
        if(size(REP.pos,1)>Nr)
            REP = deleteFromRepository(REP,size(REP.pos,1)-Nr,ngrid);
        end

        % Update the best positions found so far for each particle
        pos_best = dominates(POS_fit, PBEST_fit);
        best_pos = ~dominates(PBEST_fit, POS_fit);
        best_pos(rand(Np,1)>=0.5) = 0;
        if(sum(pos_best)>1)
           PBEST_fit(pos_best,:) = POS_fit(pos_best,:);
           PBEST(pos_best,:) = POS(pos_best,:);
        end
        if(sum(best_pos)>1)
           PBEST_fit(best_pos,:) = POS_fit(best_pos,:);
           PBEST(best_pos,:) = POS(best_pos,:);
        end

        % Plotting and verbose
        if(size(POS_fit,2)==2)
           figure(h_fig); delete(h_par); delete(h_rep);
           h_par = plot(POS_fit(:,1),POS_fit(:,2),'or'); hold on;
           h_rep = plot(REP.pos_fit(:,1),REP.pos_fit(:,2),'ok'); hold on;
           try
              set(gca,'xtick',REP.hypercube_limits(:,1)','ytick',REP.hypercube_limits(:,2)');
              axis([min(REP.hypercube_limits(:,1)) max(REP.hypercube_limits(:,1)) ...
                   min(REP.hypercube_limits(:,2)) max(REP.hypercube_limits(:,2))]);
           end
           if(isfield(MultiObj,'truePF'))
              try delete(h_pf); end
              h_pf = plot(MultiObj.truePF(:,1),MultiObj.truePF(:,2),'.','color',0.8.*ones(1,3));
hold on;
           end
           grid on; xlabel('f1'); ylabel('f2');
           drawnow;
```

```
            axis square;
        end
        if(size(POS_fit,2)==3)
            figure(h_fig); delete(h_par); delete(h_rep);
            h_par = plot3(POS_fit(:,1),POS_fit(:,2),POS_fit(:,3),'or'); hold on;
            h_rep = plot3(REP.pos_fit(:,1),REP.pos_fit(:,2),REP.pos_fit(:,3),'ok'); hold on;
            try

set(gca,'xtick',REP.hypercube_limits(:,1)','ytick',REP.hypercube_limits(:,2)','ztick',REP.hyper
cube_limits(:,3)');
                axis([min(REP.hypercube_limits(:,1)) max(REP.hypercube_limits(:,1)) ...
                    min(REP.hypercube_limits(:,2)) max(REP.hypercube_limits(:,2)) ...
                    min(REP.hypercube_limits(:,3)) max(REP.hypercube_limits(:,3))]);
            end
            if(isfield(MultiObj,'truePF'))
                try delete(h_pf); end
                h_pf =
plot3(MultiObj.truePF(:,1),MultiObj.truePF(:,2),MultiObj.truePF(:,3),'.','color',0.8.*ones(1,3)
); hold on;
            end
            grid on; xlabel('f1'); ylabel('f2'); zlabel('f3');
            drawnow;
            axis square;
        end
        display(['Generation #' num2str(gen) ' - Repository size: ' num2str(size(REP.pos,1))]);

        % Update generation and check for termination
        gen = gen + 1;
        if(gen>maxgen), stopCondition = true; end
    end
    hold off;
end

% Function that updates the repository given a new population and its
% fitness
function REP = updateRepository(REP,POS,POS_fit,ngrid)
    % Domination between particles
    DOMINATED  = checkDomination(POS_fit);
    REP.pos    = [REP.pos; POS(~DOMINATED,:)];
    REP.pos_fit= [REP.pos_fit; POS_fit(~DOMINATED,:)];
    % Domination between nondominated particles and the last repository
    DOMINATED  = checkDomination(REP.pos_fit);
    REP.pos_fit= REP.pos_fit(~DOMINATED,:);
    REP.pos    = REP.pos(~DOMINATED,:);
    % Updating the grid
    REP        = updateGrid(REP,ngrid);
end

% Function that corrects the positions and velocities of the particles that
% exceed the boundaries
```

```
function [POS,VEL] = checkBoundaries(POS,VEL,maxvel,var_max,var_min)
   % Useful matrices
   Np = size(POS,1);
   MAXLIM   = repmat(var_max(:)',Np,1);
   MINLIM   = repmat(var_min(:)',Np,1);
   MAXVEL   = repmat(maxvel(:)',Np,1);
   MINVEL   = repmat(-maxvel(:)',Np,1);

   % Correct positions and velocities
   VEL(VEL>MAXVEL) = MAXVEL(VEL>MAXVEL);
   VEL(VEL<MINVEL) = MINVEL(VEL<MINVEL);
   VEL(POS>MAXLIM) = (-1).*VEL(POS>MAXLIM);
   POS(POS>MAXLIM) = MAXLIM(POS>MAXLIM);
   VEL(POS<MINLIM) = (-1).*VEL(POS<MINLIM);
   POS(POS<MINLIM) = MINLIM(POS<MINLIM);
end

% Function for checking the domination between the population. It
% returns a vector that indicates if each particle is dominated (1) or not
function dom_vector = checkDomination(fitness)
   Np = size(fitness,1);
   dom_vector = zeros(Np,1);
   all_perm = nchoosek(1:Np,2);    % Possible permutations
   all_perm = [all_perm; [all_perm(:,2) all_perm(:,1)]];

   d = dominates(fitness(all_perm(:,1),:),fitness(all_perm(:,2),:));
   dominated_particles = unique(all_perm(d==1,2));
   dom_vector(dominated_particles) = 1;
end

% Function that returns 1 if x dominates y and 0 otherwise
function d = dominates(x,y)
   d = all(x<=y,2) & any(x<y,2);
end

% Function that updates the hypercube grid, the hypercube where belongs
% each particle and its quality based on the number of particles inside it
function REP = updateGrid(REP,ngrid)
   % Computing the limits of each hypercube
   ndim = size(REP.pos_fit,2);
   REP.hypercube_limits = zeros(ngrid+1,ndim);
   for dim = 1:1:ndim
      REP.hypercube_limits(:,dim) =
linspace(min(REP.pos_fit(:,dim)),max(REP.pos_fit(:,dim)),ngrid+1)';
   end

   % Computing where belongs each particle
   npar = size(REP.pos_fit,1);
   REP.grid_idx = zeros(npar,1);
   REP.grid_subidx = zeros(npar,ndim);
```

```
    for n = 1:1:npar
        idnames = [];
        for d = 1:1:ndim
            REP.grid_subidx(n,d) = find(REP.pos_fit(n,d)<=REP.hypercube_limits(:,d)',1,'first')-
1;
            if(REP.grid_subidx(n,d)==0), REP.grid_subidx(n,d) = 1; end
            idnames = [idnames ',' num2str(REP.grid_subidx(n,d))];
        end
        REP.grid_idx(n) = eval(['sub2ind(ngrid.*ones(1,ndim)' idnames ');']);
    end

    % Quality based on the number of particles in each hypercube
    REP.quality = zeros(ngrid,2);
    ids = unique(REP.grid_idx);
    for i = 1:length(ids)
        REP.quality(i,1) = ids(i);  % First, the hypercube's identifier
        REP.quality(i,2) = 10/sum(REP.grid_idx==ids(i)); % Next, its quality
    end
end

% Function that selects the leader performing a roulette wheel selection
% based on the quality of each hypercube
function selected = selectLeader(REP)
    % Roulette wheel
    prob    = cumsum(REP.quality(:,2));     % Cumulated probs
    sel_hyp = REP.quality(find(rand(1,1)*max(prob)<=prob,1,'first'),1); % Selected hypercube

    % Select the index leader as a random selection inside that hypercube
    idx      = 1:1:length(REP.grid_idx);
    selected = idx(REP.grid_idx==sel_hyp);
    selected = selected(randi(length(selected)));
end

% Function that deletes an excess of particles inside the repository using
% crowding distances
function REP = deleteFromRepository(REP,n_extra,ngrid)
    % Compute the crowding distances
    crowding = zeros(size(REP.pos,1),1);
    for m = 1:1:size(REP.pos_fit,2)
        [m_fit,idx] = sort(REP.pos_fit(:,m),'ascend');
        m_up     = [m_fit(2:end); Inf];
        m_down   = [Inf; m_fit(1:end-1)];
        distance = (m_up-m_down)./(max(m_fit)-min(m_fit));
        [~,idx]  = sort(idx,'ascend');
        crowding = crowding + distance(idx);
    end
    crowding(isnan(crowding)) = Inf;

    % Delete the extra particles with the smallest crowding distances
    [~,del_idx] = sort(crowding,'ascend');
```

```
    del_idx = del_idx(1:n_extra);
    REP.pos(del_idx,:) = [];
    REP.pos_fit(del_idx,:) = [];
    REP = updateGrid(REP,ngrid);
end

% Function that performs the mutation of the particles depending on the
% current generation
function POS = mutation(POS,gen,maxgen,Np,var_max,var_min,nVar,u_mut)
    % Sub-divide the swarm in three parts [2]
    fract     = Np/3 - floor(Np/3);
    if(fract<0.5), sub_sizes =[ceil(Np/3) round(Np/3) round(Np/3)];
    else           sub_sizes =[round(Np/3) round(Np/3) floor(Np/3)];
    end
    cum_sizes = cumsum(sub_sizes);

    % First part: no mutation
    % Second part: uniform mutation
    nmut = round(u_mut*sub_sizes(2));
    if(nmut>0)
        idx = cum_sizes(1) + randperm(sub_sizes(2),nmut);
        POS(idx,:) = repmat((var_max-var_min)',nmut,1).*rand(nmut,nVar) +
repmat(var_min',nmut,1);
    end

    % Third part: non-uniform mutation
    per_mut = (1-gen/maxgen)^(5*nVar);     % Percentage of mutation
    nmut    = round(per_mut*sub_sizes(3));
    if(nmut>0)
        idx = cum_sizes(2) + randperm(sub_sizes(3),nmut);
        POS(idx,:) = repmat((var_max-var_min)',nmut,1).*rand(nmut,nVar) +
repmat(var_min',nmut,1);
    end
end
```

MachiningP Code :

```
case 'Poloni'
        GW =@(x,y,z,w) - 2.08770 + 0.008463.*x + 0.041718.*y + 0.016195.*z + 0.000719.*w
- 0.000869.*x.*z + 0.000129.*y.*z + 0.003287.*x.*x - 0.000285.*y.*y - 0.000163.*z.*z-
1.41556E-06.*w.*w;

       TA = @(x,y,z,w) - 4558.82353 + 309.89583.*x + 65.900.*y + 16.26667.*z – 1.82046.*w
– 5.52500.*x.*y + 3.10625.*x.*z – 0.45000.*y.*z + 0.050667.*y.*w – 0.002338.*w.*w;

        %
        MultiObj.fun = @(x) [GW(x(:,1),x(:,2),x(:,3),x(:,4)), TA(x(:,1),x(:,2),x(:,3),x(:,4))];
        MultiObj.nVar = 4;
        MultiObj.var_min = [6 80 50 150];
        MultiObj.var_max = [10 100 90 450];
```

```
end

% Parameters
params.Np = 300;        % Population size
params.Nr = 300;        % Repository size
params.maxgen =250;     % Maximum number of generations
params.W = 0.4;         % Inertia weight
params.C1 = 1;          % Individual confidence factor
params.C2 = 1;          % Swarm confidence factor
params.ngrid = 10;      % Number of grids in each dimension
params.maxvel = 5;      % Maxmium vel in percentage
params.u_mut = 0.5;     % Uniform mutation percentage

% MOPSO
REP = MOPSO(params,MultiObj);

% Display info
display('Repository fitnez values are stored in REP.pos_fit');
display('Repository particles positions are store in REP.pos');
```

5.1.4 Statistical analysis:

Software known as design expert 13 was employed to analyse data, create quadratic models, and analyse the effect of process factors.

Table 5.3: ANOVA table for kerf deviation (KD)

Source	Sum of Squares	df	Mean Square	F-value	p-value	
Model	0.0698	10	0.0070	27.99	<0.0001	significant
A-Gas pressure	3.000E-06	1	3.000E-06	0.0120	0.9139	
B-Power	0.0003	1	0.0003	1.33	0.2646	
C-Frequency	0.0191	1	0.0191	76.66	<0.0001	
D-Feed	0.0050	1	0.0050	20.22	0.0003	
AC	0.0048	1	0.0048	19.37	0.0003	
BC	0.0027	1	0.0027	10.63	0.0043	
A^2	0.0011	1	0.0011	4.50	0.0481	
B^2	0.0053	1	0.0053	21.09	0.0002	
C^2	0.0276	1	0.0276	110.65	<0.0001	
D^2	0.0066	1	0.0066	26.38	<0.0001	
Residual	0.0045	18	0.0002			
Lack of Fit	0.0041	14	0.0003	2.93	0.1544	not significant
Pure Error	0.0004	4	0.0001			
Cor Total	0.0743	28				

The model F-value of 27.99 implies the model is significant. There is only a 0.01% chance that an F-value this large could occur due to noise. P-values less than 0.0500 indicate model terms are significant. In this case C, D, AC, BC, A^2, B^2, C^2, D^2 are significant model terms. Values greater than 0.1000 indicate the model terms are not significant. If there are many insignificant model terms (not counting those required to support hierarchy), model reduction may improve your model.

The lack of fit F-value of 2.93 implies the lack of fit is not significant relative to the pure error. There is a 15.44% chance that a lack of fit F-value this large could occur due to noise. Non-significant lack of fit is good -- we want the model to fit.

Table 5.4: Fit statistics of kerf deviation:

Std. Dev.	0.0158		R^2	0.9396
Mean	0.3377		Adjusted R^2	0.9060
C.V. %	4.68		Predicted R^2	0.8209
			Adeq Precision	18.566

The predicted R^2 of 0.8209 is in reasonable agreement with the adjusted R^2 of 0.9060; i.e., the difference is less than 0.2. Adeq precision measures the signal to noise ratio. A ratio greater than 4 is desirable. Model ratio of 18.567 indicates an adequate signal. This model can be used to navigate the design space.

Table 5.5: Coefficients in terms of coded factors

Factor	Coefficient Estimate	df	Standard Error	95% CI Low	95% CI High	VIF
Intercept	0.3842	1	0.0071	0.3694	0.3990	
A-Gas pressure	0.0005	1	0.0046	-0.0091	0.0101	1.0000
B-Power	-0.0053	1	0.0046	-0.0148	0.0043	1.0000
C-Frequency	-0.0399	1	0.0046	-0.0495	-0.0303	1.0000
D-Feed	-0.0205	1	0.0046	-0.0301	-0.0109	1.0000
AC	-0.0347	1	0.0079	-0.0513	-0.0182	1.0000
BC	0.0258	1	0.0079	0.0092	0.0423	1.0000
A^2	0.0131	1	0.0062	0.0001	0.0262	1.08
B^2	-0.0285	1	0.0062	-0.0415	-0.0154	1.08
C^2	-0.0652	1	0.0062	-0.0783	-0.0522	1.08
D^2	-0.0318	1	0.0062	-0.0449	-0.0188	1.08

The coefficient estimate represents the expected change in response per unit change in factor value when all remaining factors are held constant. The intercept in an orthogonal design is the overall average response of all the runs. The coefficients are adjustments around that average based on the factor settings. When the factors are orthogonal the VIFs are 1; VIFs greater than 1 indicate multicollinearity, the higher the VIF the more severe the correlation of factors. As a rough rule, VIFs less than 10 are tolerable.

Table 5.6: Final equation in terms of actual factors

Kerf Deviation	=
-2.08770	
+0.008463	Gas pressure
+0.041718	Power
+0.016195	Frequency
+0.000713	Feed
-0.000869	Gas pressure * Frequency
+0.000129	Power * Frequency
+0.003287	Gas pressure^2
-0.000285	Power^2
-0.000163	Frequency^2
-1.41556E-06	Feed^2

The equation in terms of actual factors can be used to make predictions about the response for given levels of each factor. Here, the levels should be specified in the original units for each factor. This equation should not be used to determine the relative impact of each factor because the coefficients are scaled to accommodate the units of each factor and the intercept is not at the centre of the design space.

Table 5.7: ANOVA table of MRR

Source	Sum of Squares	df	Mean Square	F-value	p-value	
Model	7.482E+05	9	83136.27	57.33	<0.0001	significant
A-Gas pressure	43440.33	1	43440.33	29.96	<0.0001	
B-Power	34992.00	1	34992.00	24.13	<0.0001	
C-Frequency	1825.33	1	1825.33	1.26	0.2759	
D-Feed	4.824E+05	1	4.824E+05	332.67	<0.0001	
AB	48841.00	1	48841.00	33.68	<0.0001	
AC	61752.25	1	61752.25	42.59	<0.0001	
BC	32400.00	1	32400.00	22.34	0.0001	
BD	23104.00	1	23104.00	15.93	0.0008	
D^2	19468.53	1	19468.53	13.43	0.0016	
Residual	27551.69	19	1450.09			
Lack of Fit	26308.89	15	1753.93	5.65	0.0533	not significant
Pure Error	1242.80	4	310.70			
Cor Total	7.758E+05	28				

The model F-value of 57.33 implies the model is significant. There is only a 0.01% chance that an F-value this large could occur due to noise. P-values less than 0.0500 indicate model terms are significant. In this case A, B, D, AB, AC, BC, BD, D^2 are significant model terms. Values greater than 0.1000 indicate the model terms are not significant. If there are many insignificant model terms (not counting those required to support hierarchy), model reduction may improve model.

The lack of fit F-value of 5.65 implies there is a 5.33% chance that a lack of fit F-value this large could occur due to noise. Lack of fit is bad -- we want the model to fit. This relatively low probability (<10%) is troubling.

Table 5.8: Fit statistics for MRR

Std. Dev.	38.08		R^2	0.9645
Mean	506.17		Adjusted R^2	0.9477
C.V. %	7.52		Predicted R^2	0.8812
			Adeq Precision	26.437

The predicted R^2 of 0.8812 is in reasonable agreement with the adjusted R^2 of 0.9477; i.e., the difference is less than 0.2.

Adeq precision measures the signal to noise ratio. A ratio greater than 4 is desirable. Model ratio of 26.437 indicates an adequate signal. This model can be used to navigate the design space.

Table 5.9: Coefficients in terms of coded factors of MRR

Factor	Coefficient Estimate	df	Standard Error	95% CI Low	95% CI High	VIF
Intercept	527.94	1	9.24	508.61	547.27	
A-Gas pressure	60.17	1	10.99	37.16	83.17	1.0000
B-Power	54.00	1	10.99	30.99	77.01	1.0000
C-Frequency	12.33	1	10.99	-10.67	35.34	1.0000
D-Feed	200.50	1	10.99	177.49	223.51	1.0000
AB	-110.50	1	19.04	-150.35	-70.65	1.0000
AC	124.25	1	19.04	84.40	164.10	1.0000
BC	-90.00	1	19.04	-129.85	-50.15	1.0000
BD	76.00	1	19.04	36.15	115.85	1.0000
D^2	-52.61	1	14.36	-82.66	-22.56	1.0000

The coefficient estimate represents the expected change in response per unit change in factor value when all remaining factors are held constant. The intercept in an orthogonal design is the overall average response of all the runs. The coefficients are adjustments around that average based on the factor settings. When the factors are orthogonal the VIFs are 1; VIFs greater than 1 indicate multicollinearity, the higher the VIF the more severe the correlation of factors. As a rough rule, VIFs less than 10 are tolerable.

Table 5.10: Final equation in Terms of actual factors

MRR	=
-4558.82353	
+309.89583	Gas pressure
+65.90000	Power
+16.26667	Frequency
-1.82046	Feed
-5.52500	Gas pressure * Power
+3.10625	Gas pressure * Frequency
-0.450000	Power * Frequency
+0.050667	Power * Feed
-0.002338	Feed²

The equation in terms of actual factors can be used to make predictions about the response for given levels of each factor. Here, the levels should be specified in the original units for each factor. This equation should not be used to determine the relative impact of each factor because the coefficients are scaled to accommodate the units of each factor and the intercept is not at the centre of the design space.

5.1.5 Model and analysis

The second order regression models have been developed for the analysis of quality characteristics such as KD, and MRR. The efficiency and confidence level of the second order regression model is high compared to the first order regression model. Keeping this fact in mind, the second order regression models for different responses have been developed by using experimental data. The general equation for the second order regression model is given as-

$$Y = \beta o + \sum_{i=1}^{n} \beta_i X_i + \sum_{i=1}^{n} \beta_{ii} X_i^2 + \sum_{i=1}^{n} \sum_{i=1+1}^{n} \beta_{ij} X_i X_j$$

The regression coefficients or partial regression coefficients in this equation are denoted by the letters β. Because these coefficients demonstrate the average change in Y caused by changing X_i while other factors remain fixed, they are known as partial regression coefficients. By minimising the mean square error or by applying the least squares approach, the values of these coefficients can be determined.

X_i (i = 1, 2, 3...n), are the input (Independent) or process variables and n is the total number of the input variables or control factors (independent). The Y is the output/response. For finding the values of regression coefficients β's, Design Expert 13 software has been used. The second-order response surface models or regression models for KD and MRR are-

Kerf deviation (KD) = - 2.08770 + 0.008463 A + 0.041718 B + 0.016195 C + 0.000719 D - 0.000869 AC + 0.000129 BC + 0.003287 A^2 - 0.000285 B^2 - 0.000163 C^2 - 1.41556E-06 D^2.

– $eq^{n.}$ 5.1

Material removal rate (MRR) = - 4558.82353 + 309.89583 A + 65.900 B + 16.26667 C – 1.82046 D – 5.52500 AB + 3.10625 AC – 0.45000 BC + 0.050667 BD – 0.002338 D^2.

---$eq^{n.}$5.2

5.2 Multi-objective optimization on laser cutting parameters of A653 galvanised steel

A single optimal solution is generated by a single-response optimization process. However, most multi-response problems principally yield a set of optimal solutions instead of a single optimal solution. Ultimately, the multi-response optimization algorithm functions in maintaining a balance between two or more responses in terms of quality and productivity. The present work considers two responses, namely, KD (quality consideration) and MRR (productivity consideration). Given the production purpose, the optimal combination of parameter levels should produce the maximum MRR and the minimum KD. Particle Swarm Optimisation and Response Surface Methodology is an improved and unique numerical optimization technique used in the industry for the optimization of multiple quality characteristics. The optimization module searches for a combination of process parameter levels that simultaneously satisfy the requirements placed on each of the responses and

process parameters. The developed KD and MRR model equations (5.1 and 5.2) were simultaneously solved to yield the optimal process variables.

5.2.1 Particle swarm optimization (optimal solution) and quality improvement:

The hybrid approach of regression and particle swarm optimization has been used for the optimization of kerf deviation (KD), material removal rate (MRR). The second order regression model developed for the kerf deviation (KD); material removal rate (MRR) has been utilized for the particle swarm optimization. These models have been used as objective functions for the optimization. The PSO optimization coding has been done in MATLAB and after selecting the critical parameters, the program has been run. The optimization result is shown the relationship between the kerf deviation (KD) and MRR (Material Removal Rate) represent with the help of figure 5.1. The selected operator setting is shown in Table 5.11.

Table 5.11: Critical operating parameters

Parameter setting for PSO				
C_1	C_2	W	Population size	Max. Iteration
1	1	0.4	300	300

The optimized values by PSO are 0.164 mm for kerf deviation (KD) and 834.468 mg/min for material removal rate (MRR).

The PSO optimization gives the predicted values for the improvement at optimum control parameter setting for individual quality characteristics. The optimize value found for kerf deviation (KD) and material removal rate (MRR) at optimum setting as Gas Pressure is 9.26 kgf/cm^2, Power is 80 w, Frequency is 89.9716 Hz and Feed is 450 mm/min.

And the graph in figure 5.1 shows the relation between the kerf deviation (KD) and the material removal rate (MRR). The graph shows that the depth of the MRR is maximum when the KD is minimum and thus the relation is inversely proportional, between the KD and the MRR.

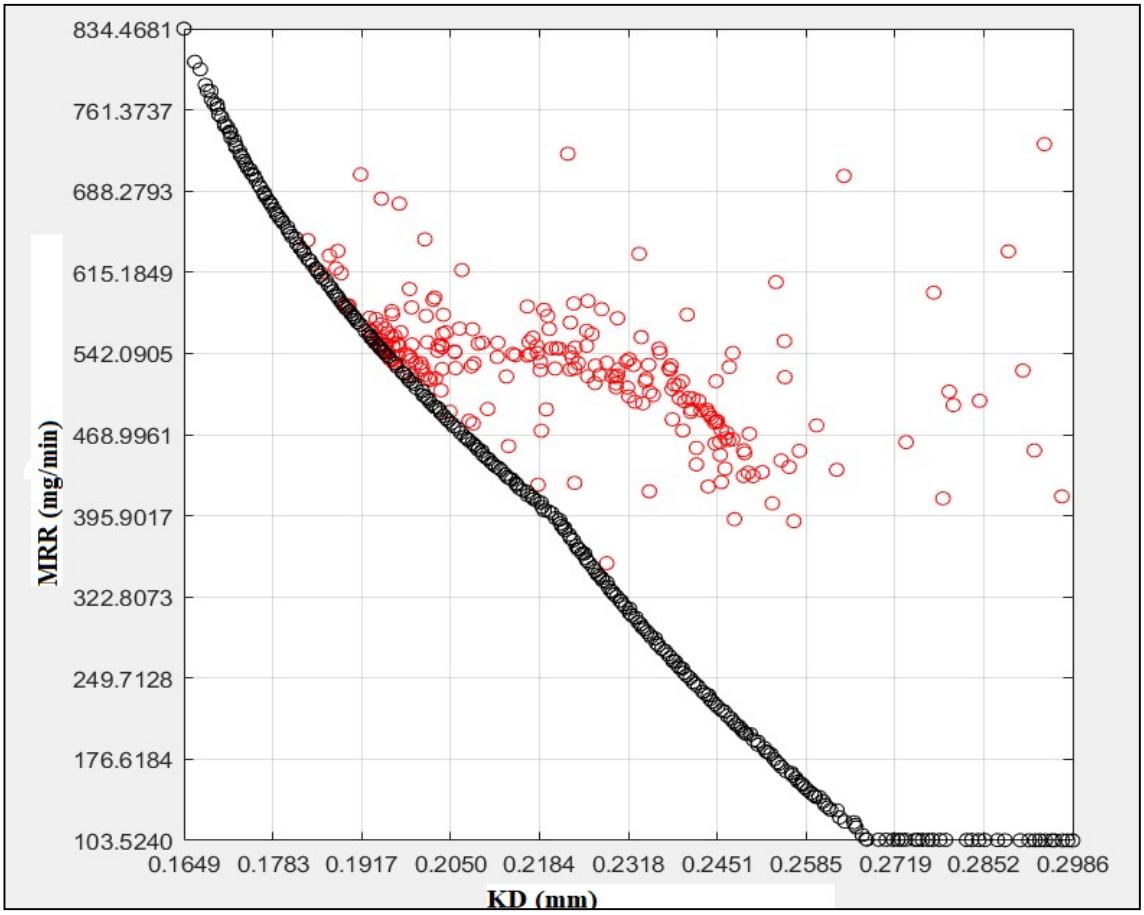

Figure 5.1: PSO optimisation of MRR and KD graph

5.2.2 RSM optimisation

Response Surface Methodology has been used for the optimization of kerf deviation (KD), Material removal rate (MRR). The second order regression model developed for the kerf deviation (KD); material removal rate (MRR) has been utilized for the response surface methodology. These models have been used as objective functions for the optimization. In the study, optimization of KD and MRR has been done during fiber laser beam cutting of A653 galvanised steel sheet using RSM where a response optimizer module of design expert 13 software is used. The goal set, lower limits, upper limits, weights used, and importance of the factors given are shown in Table 5.12.

The minimum and maximum levels were provided for each parameter included. The goals used for kerf deviation and MRR are “minimize” and “maximize”, and the goal used for the

factors or process parameters is “within range”. A weight was assigned to each goal to adjust the shape of its desirability function. The importance of each goal was changed in relation to other goals. Optimization was carried out for a combination of goals.

Figure 5.2 represents the multi objective optimisation analysis for minimizing the kerf deviation (KD) and maximising the MRR based on the developed mathematical model as given by Eq. (1) and Eq. (2). For linear desirability function (d) the value of weight is considered 1 for kerf deviation (KD) and 1 for material removal rate (MRR), from Table 5.12. This figure 5.2 represents the simultaneous optimization of both quality characteristics in one setting.

Table 5.12: Target and goal settings in RSM optimisation

Name	Goal	Lower Limit	Upper Limit	Lower Weight	Upper Weight	Importance
A: Gas Pressure	is in range	6	10	1	1	3
B: Power	is in range	80	100	1	1	3
C: Frequency	is in range	50	90	1	1	3
D: Feed	is in range	150	450	1	1	3
KD	minimize	0.238	0.416	1	1	3
MRR	maximize	219	802	1	1	3

Table 5.13: Solution of optimal data of RSM optimisation

$Sol^{n.}$	Gas Pressure (kgf/cm^2)	Power (W)	Frequency (Hz)	Feed (mm/min)	MRR (mg/min) Fit	KD (mm) Fit	CD^*
1	9.635	84.939	89.779	386.774	829.309	0.222	1

*CD means composite desirability.

The predicted optimum values (given in Table 5.13) of kerf deviation (KD) and material removal rate (MRR) during multi objective optimisation are 0.222 mm and 829.309 mg/min, respectively at gas pressure 9.635 kgf/cm^2, power 84.939 W, frequency 89.779 Hz and feed 386.774 mm/min.

Table 5.14: Predicted optimise input parameters for KD and MRR

Variable	Setting
Gas Pressure (kgf/cm^2)	9.635
Power (w)	84.939
Frequency (Hz)	89.779
Feed (mm/min)	386.774

The value of composite desirability, (D) is 1.000 and individual desirability, (d) for kerf deviation (KD) and material removal rate (MRR) are 1.0000 and 1.0000.

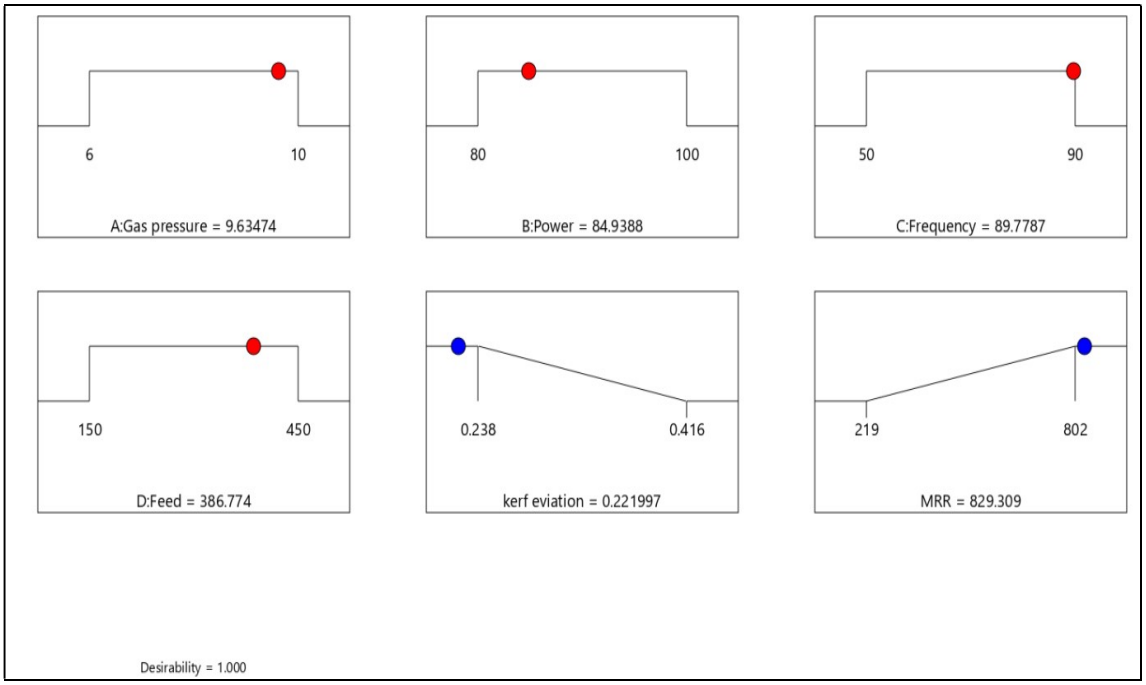

Figure 5.2: RSM optimisation graph of KD and MRR.

Now, by multi objective technique of response surface methodology (RSM) optimisation technique, the goal is to achieve the target minimum kerf deviation (KD) when input parameter maximizes the cutting speed (mm/min) with all the other parameter like gas pressure, power and frequency remain in between their range value. Table 5.15, shown below shows goal of maximise cutting speed when kerf deviation (KD) targeted at minimised value.

Table 5.15: Constraints of maximise scanning speed when kerf deviation fixed at minimise value

Name	Goal	Lower Limit	Upper Limit	Lower Weight	Upper Weight	Importance
A: Gas Pressure	is in range	6	10	1	1	3
B: Power	is in range	80	100	1	1	3
C: Frequency	maximize	50	90	1	1	3
D: Feed	is in range	150	450	1	1	3
KD	Is target = 0.238	0.238	0.416	1	1	3
MRR	none	219	802	1	1	3

RSM based multi objective optimisation was performed to get the best combination of parameters to get the desired value for targeted minimised kerf deviation (KD) when cutting speed (mm/min) is set to the maximised value. Table 5.16 shows that when kerf deviation is targeted at a minimised value of 0.238 mm value, the optimised value of feed will be 450.000 mm/min, when importance is set to +++ and desirability is 1.000.

Table 5.16: Solution of optimal values when cutting speed is fixed at maximum value

No	Gas Pressure	Power	Frequency	Feed	KD	Desirability	
1	7.099	99.81	50.393	450.0	0.238	1.000	Selected
2	8.203	88.19	87.749	450.0	0.238	1.000	
3	7.832	86.23	87.631	450.0	0.238	1.000	
4	8.495	80.16	82.601	450.0	0.238	1.000	
5	8.475	85.78	86.355	450.0	0.238	1.000	

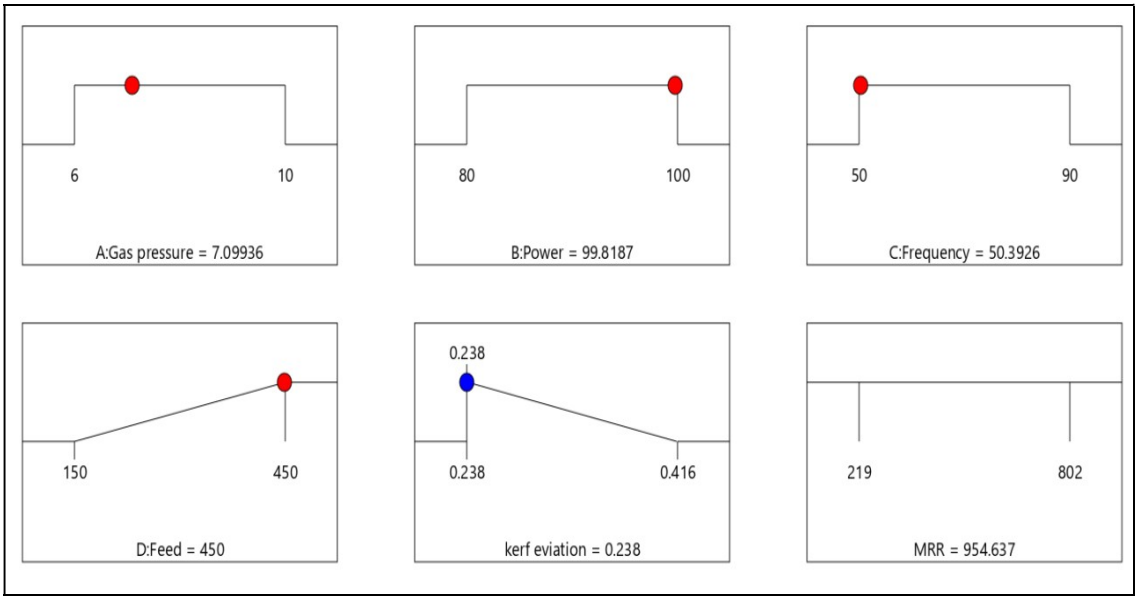

Figure 5.3: Constrains RSM optimisation

5.2.3 Comparison of two optimization

Table 5.17: Comparison of predicted optimal input process conditions of RSM and PSO optimisation

Controlling Factor	Unit	PSO Optimal Set	RSM Optimal Set
Gas Pressure	Kgf/cm^2	9.26490	9.63474
Power	W	80	84.9388
Frequency	Hz	89.97	89.7787
Feed	mm/min	450	386.774

Table 5.18: Comparison of predicted optimal output responses of RSM and PSO

Responses	PSO Optimal Responses	RSM Optimal Responses
Kerf Deviation (mm)	0.164949	0.222
Material removal rate (mg/min)	834.468	829.309

CHAPTER -6
CONCLUSION

6.1 General conclusion

In present research study, attempt has been made for optimization of laser cutting of high strength engineering materials i.e., Ti-6Al-4V super alloy and A653 galvanised steel. Experimental data has been taken from the available literature [3, 16] in order to study the modelling and optimisation. The regression equations are developed between the process parameters and the machining characteristics such as, kerf deviation and material removal rate. Finally, the developed regression equations are used for optimisation of the laser cutting process parameters to obtain the minimum kerf deviation and maximum material removal rate using Response Surface Methodology and Particle Swarm Optimisation. The following conclusions can be made from this study based on the range of values of parameters considered.

1. As can be seen from the literature review, response surface methodology based optimisation technique is mostly used to find the feasible optimal solutions for laser cutting process. Moreover, a few researchers also used a hybrid approach based on the genetic algorithm and neural artificial network modeling to improve the performance of the optimisation.
2. For Ti-6Al-4V super alloy, the optimized values found by PSO for minimum kerf deviation (KD) and maximum material removal rate (MRR) are 0.00971 mm and 192.9321 mg/min, respectively, at level of pulse width of 2 ms, pulse energy of 4.64 J, cutting speed of 10 mm/min and gas pressure of 8.17 kg/cm^2.
3. For Ti-6Al-4V super alloy, the optimized values found by RSM for minimum kerf deviation (KD) and maximum material removal rate (MRR) are 0.011 mm and 181.063 mg/min, respectively, at level of pulse width of 1.822 ms, pulse energy of 5.020 J, cutting speed of 10.104 mm/min and gas pressure of 8.176 kg/cm^2.
4. For A653 galvanised steel, the optimized values found by PSO for minimum kerf deviation (KD) and maximum material removal rate (MRR) are 0.164 mm and 834.468 mg/min, respectively, at level of frequency of 89.9716 Hz, laser power of 80 W, cutting feed of 450 mm/min and gas pressure of 9.26 kg-f/cm^2.
5. For A653 galvanised steel, the optimized values found by RSM for minimum kerf deviation (KD) and maximum material removal rate (MRR) are 0.222 mm and

829.309 mg/min, respectively, at level of frequency of 89.779 Hz, laser power of 84.939 W, cutting feed of 386.774 mm/min and gas pressure of 9.635 kg-f/cm^2.

6. The optimal process parameters are found by RSM for targeted kerf deviation (0.238 mm) at relatively low operating energy cost by maximizing scanning speed. Because, maximizing the scan speed helps to minimize the machining time, which leads to increase productivity.
7. Particle Swarm Optimisation algorithm is also capable of predicting the parametric trends of responses.
8. In present recent study, optimisation technique PSO algorithm gives better results in all cases compare to the optimisation technique RSM that has been adopted by the previous researchers. PSO technique should be used for optimisation instead of RSM.

References

1. “Multi – objective Optimisation of Laser beam cutting Process”, Kumar Dubey A. and Vinod Yadava (2008)
2. “Selection of optimal machining parameters in pulsed CO_2 laser cutting of Al6061/Al_2O3 composite using Taguchi based response surface methodology”, Adalarasan R., M. SanthanaKumar and Thileepan S.(2016)
3. “ Multi objective optimisation of Nd: YAG laser cutting parameters of Ti-6Al-4V superalloy sheet”, Tamilarasan A. and Rajamani D. (2016)
4. “Multi objective optimisation of Pulsed Nd: YAG laser cutting process using entropy based ANN-PSO Model” Chaki Sudipto, Bose Digbalay and Bathe N. Ravi.(2020)
5. “Geometrical quality evaluation in laser cutting of Incole 718 sheet by using Taguchi based regression analysis and Particle swarm optimisation”, Prasant Kumar Shrivastava, Arun Kumar Pandey (2018)
6. “Particle Swarm optimisation prediction model for surface roughness”, M.M.Noor, K. Kadirgama and M.M. Rahman. (2011)
7. “Machining Parameters Optimization of Titanium Alloy using Response Surface Methodology and Particle Swarm Optimization under Minimum-Quantity Lubrication Environment “, Manish Kumar Gupta, P.K.Sood and Vishal S.Sharma (2015)
8. “Application of particle swarm optimization and response surface methodology for machining parameters optimization of aluminium matrix composites in milling operation” Rashmi Laxmikant Malghan, Karthik M. C. Rao , Arun Kumar Shettigar, Shrikantha S. Rao, R. J, D’Souza (2017)
9. “Investigation of optimal process parameters for laser cutting of Inconel-718 sheet”, Prasant Kumar Shrivastava, Bhagat Singh and Yogesh Shrivastava (2019)
10. “Prediction and optimisation of dross formation in laser cutting of electrical steel sheet in different environments”, Muhamad Nur Rohman, Jeng-Rong Ho, Pi-Cheng Tung, Chin-Te Lin, Chih-Kuang Lin (2022)
11. “Multi-objective optimization and analysis for laser beam cutting of stainless steel (SS304) using hybrid statistical tools GA-RSM”, A D Tura, H B Mamo, and D G Desisa (2021)
12. “Hybrid neural network–particle swarm optimization algorithm and neural network–genetic algorithm for the optimization of quality characteristics during CO2 laser

cutting of aluminium alloy", Senthilkumar Vagheesan and Jayaprakash Govindarajalu (2019)

13. "Research Article Nd: YAG laser cutting of Hastelloy C276: ANFIS modeling and optimization through WOA", D. Rajamani, M. Siva Kumar, E. Balasubramanian and A. Tamilarasan (2021)
14. "Neural Network Modeling and Particle Swarm Optimization (PSO) of Process Parameters in Pulsed Laser Micromachining of Hardened AISI H13 Steel", J. Ciurana,
    G. Arias and T. Ozel (2008)
15. "Prediction of optimal cut quality characteristic of Inconel 718 sheet by genetic algorithm and particle swarm optimization", Prasant Kumar Shrivastava, Bhagat Singh and Yogesh Shrivastava.
16. "Study on material transfer and surface properties during fiber laser cutting of A653 galvanized steel sheet", Vipin Kumar Sharma and Vinod Kumar (2019)
17. "Multi-objective Optimization of Kerf-taper and Surface-roughness Quality Characteristics for Cutting-operation On Coir and Carbon Fibre Reinforced Epoxy Hybrid Polymeric Composites During CO2-Pulsed Laser-cutting Using RSM", Yadvinder Singh (2021)
18. " optimisation of Process Parameters for Laser Cutting Process of Stainless steel 304: A comparative analysis and estimation with Taguchi Method and Response Surface Methedology", Vi Nguyen, Faisal Altarazi and Thanh Tran (2022)
19. "The Impact of Process Parameters on Surface Roughness and Dimensional Accuracy during CO2 Laser Cutting of PMMA Thin Sheets", Konstantinos Ninikas, John kechagias, and Konstantinos Salonitis (2021).