Method for color laser marking process optimization with the use of genetic algorithms

Pavels Narica, Lyubomir Lazov, Artis Teilans, Peteris Grabusts, Edmunds Teirumnieks, Pavels Cacivkins
Rezekne Academy of Technologies, Faculty of Engineering, Rezekne, Latvia

Abstract. Optimization of color laser marking process mostly depends on effective identification of optimal values of laser marking parameters. This is a difficult combinatorial optimization problem, which is still essential for companies that use laser marking systems. The study proposes a new approach to the process optimization through the use of genetic algorithms, carrying out preliminary experimental investigation, analyzing the laser marking results, and presenting possible improvements to the current implementation of genetic algorithms.

Keywords — color laser marking, combinatorial optimization problem, genetic algorithms, machine learning

I. INTRODUCTION

Laser marking systems basically are composed of three main components: control unit, laser source, and scanner. Control unit interprets laser operator's commands and manipulates both laser source and scanner. Laser source generates beam and is controlled by using such input parameters as pulse repetition frequency, pulse energy, pulse duration. Scanner directs laser beam and is controlled using such input parameters as scanning speed, line step, defocus. Input parameters of both laser source and scanner are collectively referred to as laser parameters [1][2].

Color laser marking of stainless steel is complicated physical process. Complexity becomes apparent when one compares and interprets values of laser parameters used during experiment and stainless steel laser marking results [4][5][6]. To make things even more complicated, the producers of laser marking systems usually do not provide pulse energy laser parameter to manipulate directly. The value of pulse energy can be set indirectly using average power and pulse repetition frequency, but then again laser marking systems usually do not allow to set value of average power laser parameter directly. Instead, arbitrary power regulation coefficient is provided, which gives different control for each distinct type of laser marking system. Thus, for some laser marking systems the maximum value of power regulation coefficient always corresponds to the maximum possible value of average laser power, but for others it does not mainly due to the limitations of maximum possible pulse energy that system can generate for specific low pulse repetion frequencies.

Nevertheless, the number of possible combinations of laser parameters values is vast, although for each specific combination there is only one possible physical outcome (laser marking color), given properties of environment and material stay the same. Therefore repeatability of experiments can be ensured [4][5][6]. All things considered, the combinatorial optimization problem still persists, as it is quite difficult to determine correct combination of laser parameters values for desired laser marking result.

We propose using genetic algorithms for solving this problem. Genetic algorithms come from the field of machine learning algorithms and help imitate the iterative process of natural selection of fittest solutions. Basically genetic algorithms are search algorithms that, given the objective function, stochastically and iteratively generate new solutions based on information gathered from previous generations of solutions. Genetic algorithms are commonly used to solve different kinds of combinatorial optimization problems, i.e., in economics, design, scheduling, artificial neural networks, music [3][7].

The resulting solutions are coded using sequences of numeric values (codes) before actual marking, as it so happens that values of laser marking parameters can be represented in such form. Initially, a number of random codes are generated. These are applied to metal surface using laser and the resulting solutions are evaluated. Best laser marking results are selected and used to produce the next generation of solutions by crossing and mutating codes of selected solutions. In each new generation current possible best solutions, which are better than previous ones, may emerge. Solutions that are more fit get to pass their codes to next generations of solutions. This way only best solutions tend to survive.

To sum up, some basic assumptions were made before experimental investigation:

- there is vast amount of possible combinations of laser parameters values;
- two identical combinations of laser parameters values will produce the same marking color;
- small change in laser parameter value will not cause big change in produced marking color;
- the functional relation between power regulation coefficient and average power is not linear;
- there might be some very unique construction specifics or errors in laser marking system.

The aim of this study is to use genetic algorithms in order to produce a set of high quality laser marking colors in shortest possible amount of time.

II. METHODOLOGY

A. Experimental set-up

Preliminary experimental investigation was carried out on Rofin PoweLine F-20 Varia laser marking system, which has laser source with maximum average power of 20 W, pulse repetition frequency of 2-1000 kHz, and adjustable pulse width of 4-200 ns. The system provides five main laser parameters to manipulate: power regulation coefficient, pulse repetition frequency, scanning speed, pulse duration, and line step. The stainless steel used for color laser marking was of type 4301 18-9E 304. The diagram of experimental set-up is shown in Fig. 1.
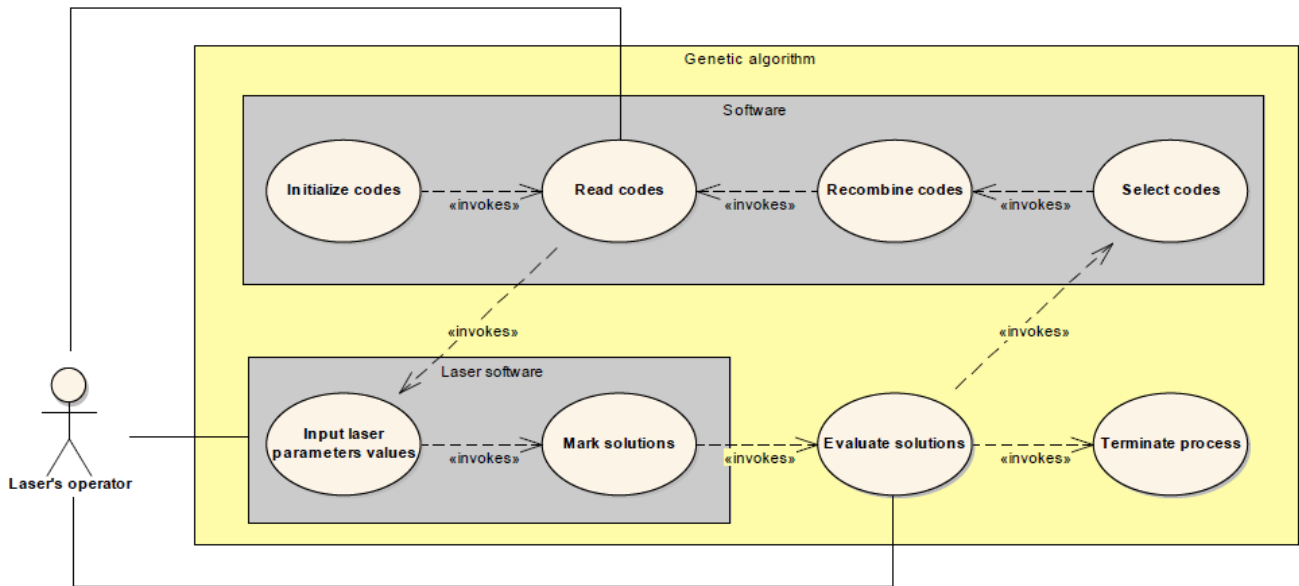
Fig. 1. Experimental set-up diagram – laser operator reads codes from computer program, operates laser system, and evaluates solutions

As shown in Fig. 1, the plan of experimental investigation is to iteratively use genetic algorithms to produce new marking colors until termination criterion is met.

B. Mathematical model

The aim of this study was to be achieved by using the computer program based on our implementation of genetic algorithm. There exist many variations of implementation of genetic algorithms. Basically, they all differ in how each stage of genetic algorithm is implemented. There are five main stages of any genetic algorithm: initialization, evaluation, selection, recombination, and termination (Fig. 2) [3].

Our implementation of genetic algorithm includes a computer program responsible for initialization, selection, and recombination stages, while evaluation and termination stages are done outside the computer program. Each stage is described below.

Initialization - in this stage initial 48 random codes were generated by genetic algorithm and grouped into first generation. As a result each code represented distinct combination of values of five main laser parameters. Genetic algorithm was configured so that for each laser parameter only values within allowed range were generated.

Evaluation - the process of evaluation included laser marking of current generation's codes and then evaluating produced solutions individually and all solutions as a whole. Each solution was ranked as either good looking marking color or not. If current generation's solutions converged (i.e., became similar), the process was terminated, otherwise the information about all the rankings was passed to selection stage of genetic algorithm.

Selection - here information about rankings of solutions was used to sort associated codes in such order that codes associated with good looking colors would have higher probabilities of being selected to later take part in recombination stage. Nonetheless, it was ensured that codes associated with bad looking colors still had at least some

probability of being selected. Then 24 combinations of two codes were selected and sent to recombination stage.
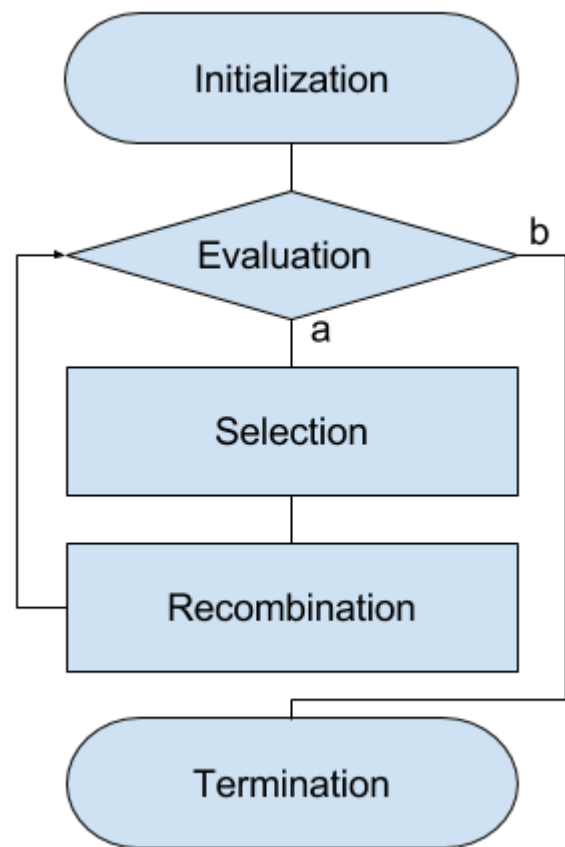

Fig. 2. Schematic view of the stages of the iterative process of genetic algorithms (where *a* path is taken when new generation of solutions is required and *b* path when no more generations of solutions are required)

Recombination - here each retrieved combination of two codes was used to produce new code by the process

of crossover and mutation (Fig. 3). As a result, 24 new codes were produced and grouped into new generation. This generation was then sent back to the evaluation stage of genetic algorithm.

P1	P2	P3	P4	P5
47	187315	461	4	0.048
15	615015	188	8	0.039

↙ Crossover

P1	P2	P3	P4	P5
47	187315	188	8	0.039
15	615015	**498**	4	0.048

Mutation ↗

Fig. 3. Example representation of how two new codes are formed from two previously selected codes through crossover and mutation

Termination - when no more new generations of solutions are required or solutions have converged, the iterative process of genetic algorithm is stopped.

III. RESULTS

It is important to mention that our implementation of genetic algorithm "knew" almost nothing about the laser marking system we used and completely nothing about the physical process of color laser marking. All it "knew" was that codes represented 5 parameters (Fig. 4), values of which had to belong to some allowed sets of possible values, and whether codes produced good solutions or not.
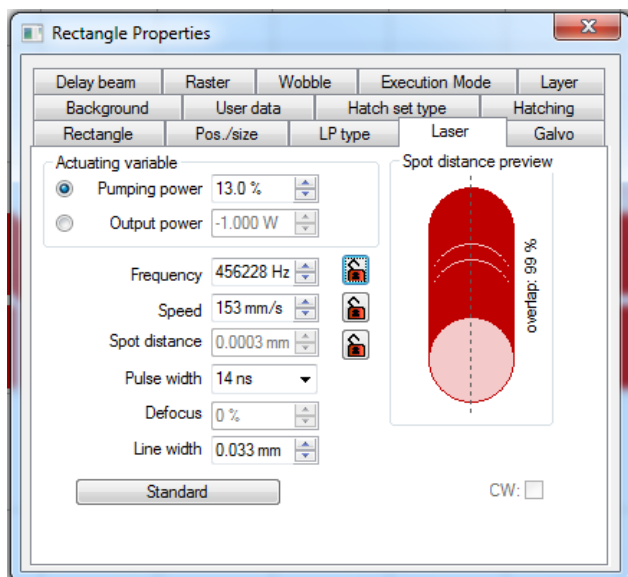

Fig. 4. Five main input laser parameters and their values as enetered by laser operator in Rofin VisualLaserMarker computer program

All marking colors (solutions) were grouped into generations with first containg 48 initial random solutions and all consecutive having 24 solutions each. Each solution represented specific combination of laser parameters values (code) and was laser marked as a filled square of some color on a surface of stainless steel (Fig. 5). The objective function of genetic algorithm was to select only best looking marking colors in each generation of produced solutions.

During preliminary experimental investigation it was observed that values of some laser parameters had a tendency to converge quicker. Such laser parameters included power regulation coefficient, pulse repetition frequency, and pulse duration. Still, convergence was not complete, i.e., multiple distinct values for each of these parameters were present.

The values of other laser parameters, that were less prone to convergence, varied across wider ranges of possible values and made associated codes more diverse. Besides, many of these diverse codes contributed to diferent looking laser marking colors.

In Fig. 5 one can see that color laser marking results on average tend to get better in each new generation – more new colors appear. Although, one can also see that beginning with generation G13 green marking colors start to dominate, and it is a sign that solutions can soon converge.

Interestingly enough, it was also observed that sometimes two good looking solutions can result in a really bad looking solution, as the program was developed in such a way that information about the origins of each solution was always logged. Another observation was that after one seemingly bad generation (G9) there still can emerge a reasonably good one.

IV. DISCUSSION

The results of preliminary experimental investigation helped us draw some very important conclusions, mainly, that diversity among each generation's solutions must be always ensured. Still, current version of the program lets us identify subsets of optimal values of some laser parameters very efficiently due to the effect of convergence described in previous section.

The results certainly look promising, but more research is required. The program developed by our team as well as the overall methodology is to be further improved. Thus, this study is only the first part in the coming series of articles on using machine learning algorithms in order to optimize different kinds of laser materials processing problems.

Ideally, genetic algorithms should "know" very little if nothing at all about the problem they are utilized to find solution(s) to. The purpose of genetic algorithms is to find optimal solutions within the hyperspace of all possible solutions without using domain-specific knowledge during the process. That is why they can be used for optimizing different kinds of problems, to which solutions are coded in a form of a symbolic sequences, and color laser marking is truly compatible with this approach.
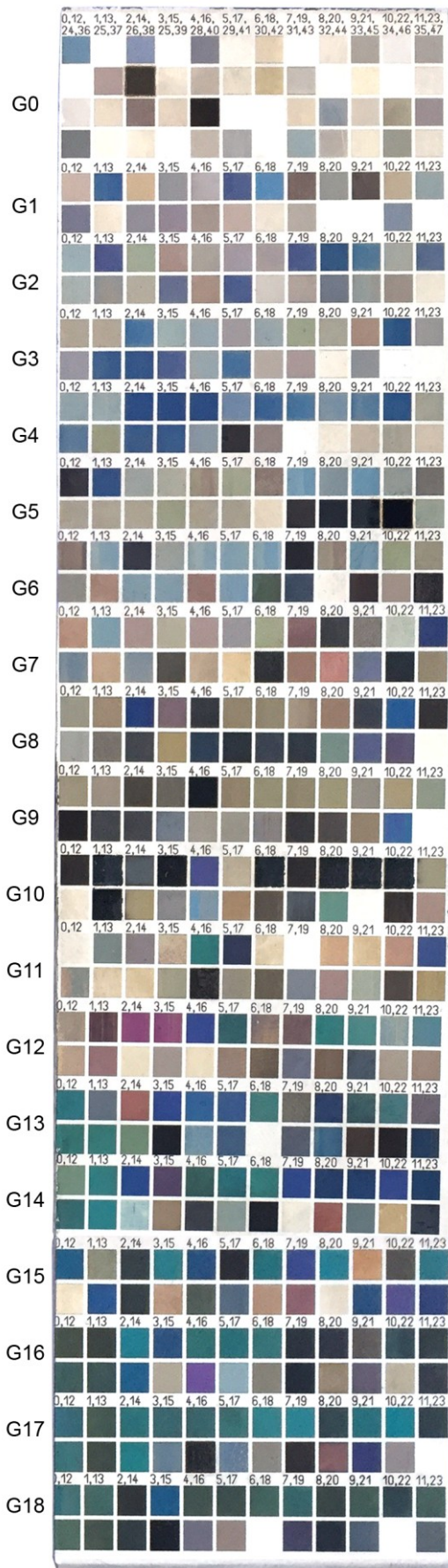
Fig. 5.   Total of 19 generations of produced solutions

There exist many objective functions one can define for genetic algorithms in order to optimize color laser marking process, for example, determining combinations of laser parameters values for one particular color or for marking colors at higher scanning speeds. Furthermore, as genetic algorithms "know" nothing about the construction specifics of laser marking system itself, lack of information about the pulse energy or about the presence of some errors within a system or its software has less significance, as long as for each distinct combination of laser parameters values the color laser marking result is the same.

The great thing about genetic algorithms is that they can process multiple parameters at once as well as produce multiple distinct solutions in parallel. This helps solve very complex problems, the objective function of which can change in time. Bad solutions are quickly discontinued, while mostly good solutions keep getting recombined to produce even more new candidate solutions. Because genetic algorithms do not try to optimize for only one solution, diversity must be ensured at all times [7].

V.   REMARKS

To better ensure diversity among produced solutions, first, the way these solutions are encoded must be reviewed. Thus, the values of laser parameters must be normalized and then converted to binary form so that there emerge more new ways to apply crossover and/or mutation to codes during recombination stage of genetic algorithm (Fig. 6).
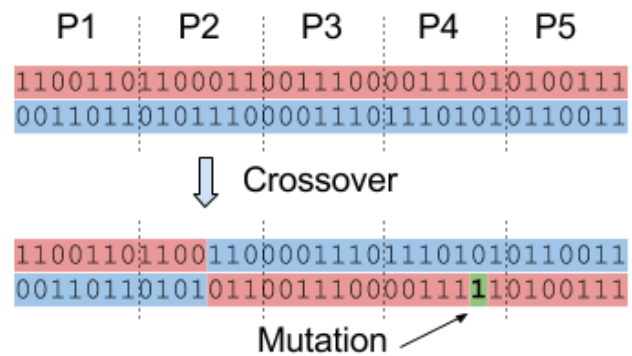

Fig. 6.   Example representation of laser parameters values in a form of normalized binary code

Now each individual bit itself can be interpreted as parameter, and the more there are parameters, the more there are ways to split code for crossover (see Fig. 2 for comparison). Because values of laser parameters are normalized, every laser parameter has equal probability of being split. In addition to that mutations become much simpler. Still, each such simple mutation can have a significant impact on real laser parameter value when converted back do decimal.

Next, it can be allowed for crossover to take place in multiple positions within a code (Fig. 7). Thus, the number of possible combinations of candidate codes formed from two old codes certainly becomes larger and can be calculated as $C(n, m)$, i.e., the number of ways that

m split positions can be chosen from *n* possible positions, where *n + 1* is also a number of bits forming a code. Here, *C(n, m)* approaches the largest possible number, as *m* approaches *n / 2*. In such case new code is formed from two approximately equally long parts of old codes.
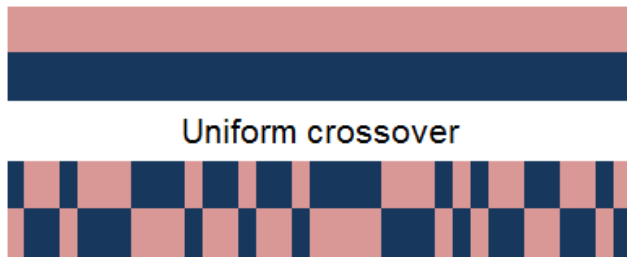

Fig. 7. Example representation of crossover taking place in multiple positions at once uniformly so that each of two new-formed codes consists of approximately equal halves of two previous codes

Then, crossover taking place in multiple positions at once can be applied to multiple codes at once (Fig. 8). Such approach can also increase the number of possible candidate solutions dramatically. This way search within solution hyperspace can be even more complete.
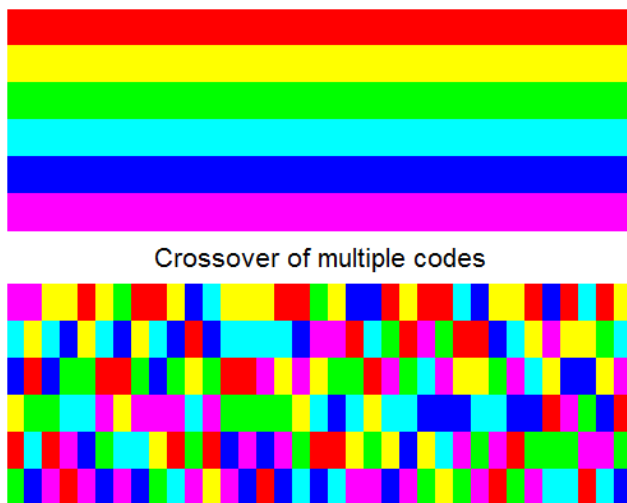

Fig. 8. Example recombination of six old codes into six new codes by applying crossover in multiple positions

Though, it may still occur that two selected codes share identical parts (Fig. 9).
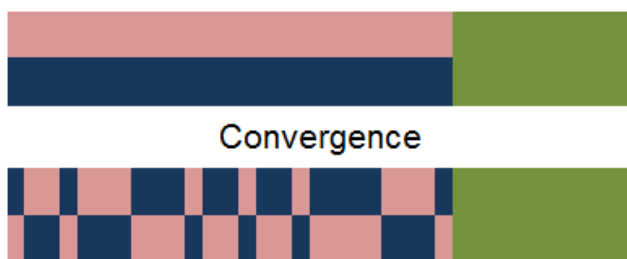

Fig. 9. Solution to convergence problem must involve mutation

The implementation of genetic algorithm can include a mechanism, which would allow introduction of new random codes at each new generation. This could further ensure maintaining diversity among solutions. Although diversity is very important for efficient search throughout hyperspace of all possible solutions, sometimes there might be a need to turn the diversity off. Such mechanism would allow to quickly converge solutions and terminate genetic algorithm.

In the future, it is also planned to study the introduction of optimal solutions in the initialization stage of genetic algorithm. Furthermore, because genetic algorithms help generate many diverse solutions (marking colors), these solutions are to be associated with their encodings by the process of training an artificial neural network. Such artificial neural network would then be able to predict color laser marking outcome for input parameters, and vice versa.

VI. CONCLUSION

The program developed by our team may be useful for laser operators, as it simply helps determine values of laser parameters for producing high quality laser marking colors in shortest possible amount of time without requiring deep understanding of laser parameters and actual physical processes that occur during stainless steel color laser marking. Thus, laser operator does not need to do exhaustive search for optimal combinations of values of laser parameters, but instead he/she needs to define objective function and evaluate marking colors according to it. The process of evaluation for some objective functions can further be automated.

Finally, it is worth mentioning that genetic algorithms do indirectly posess some kind of learning memory, as information from previous good solutions reappears in form of parts of code associated with new solutions. This seems to be really important emergent feature of genetic algorithms.

REFERENCES

[1] J. Bliedtner, H. Muller, A. Barz, "Lasermaterialbearbeitung" Fachbuchverlag Leipzig im Carl Hanser Verlag, pp. 529-551, 2013.

[2] J. Eichler, H. J. Eichler, "Laser", Springer-Verlag Berlin, 1998.

[3] M. Tim Jones, "Al Application Programming", CHARLES RIVER MEDIA, INC., Hingham, Massachusetts, 2005.

[4] P. Laakso, S. Ruotsalainen, H. Leinonen, A. Helle, R. Penttilä, A. Lehmuskero, J. Hiltunen, "Direct color marking of metals with fiber lasers", Research Report VTT-R-02403-09, http://wlsc.com/laser-systems/PDF/DirectColorMarkingFinalReport.pdf

[5] A. J. Antonczak, B. Stepak, P. E. Kozioł, K. M. Abramski, "The influence of process parameters on the laser-induced coloring of titanium", Applied Physics A, 2014.

[6] V. Veiko, G. Odintsova, E. Ageev, Y. Karlagina, A. Loginov, A. Skuratova, et al., "Controlled oxide films formation by nanosecond laser pulses for color marking", 2014, http://dx.doi.org/10.1364/OE.22.024342

[7] Krantikumar K. Bhosale , Vijay D. Jadhav, "A review of genetic algorithm for metal cutting processes and a research agenda", 2015, IJRASET Volume 3 Issue V.