الـجـمــهــوريــة الـجـزائـريــة الـديـمـقـراطـيــة الـشـعـبـيــة

**People's Democratic Republic of Algeria**

وزارة الـتـعـلـيـم الـعـالـي والـبـحـث الـعـلـمــي

**Ministry of Higher Education and Scientific Research**

الـمـــدرســـة الـوطـنـيــة العليا لـلـتـكـنـولـوجـيـات الـمـتـقـدمـة

**National Higher School of Advanced Technologies**

Department of Industrial Engineering and Maintenance


**Final Year Project to Obtain the Diploma of Engineering**

-Field -

**Electromechanics**

-Speciality-

**Mechatronics**

- Subject -

# Comparative Study of Workspace Optimization Algorithms for Serial Robots


Realized by:

SALHI Youcef


**Members of The Jury:**

| | | |
|---|---|---|
| MCB ENSTA | CHABANE Ali | President |
| MCB ENSTA | AMRANE Ahmed | Examiner |
| PROFESSOR ENSTA | HAOUARI Fouad | Supervisor |

**Algiers, the 15/05/2024**

Academic year **2023–2024**

**Abstact:** This article explores the kinematic modeling of serial manipulators, emphasizing the significance of workspace analysis in optimizing their performance. Key factors such as joint forward kinematics and Inverse kinematics, are discussed in relation to workspace characterization. Moreover, the application of optimization algorithms, including Genetic Algorithms (GA), Particle Swarm Optimization (PSO), and Monte Carlo methods (MC), is examined for enhancing workspace utilization and manipulability. By integrating kinematic modeling with advanced optimization techniques, this study aims to provide insights into maximizing the operational capabilities and adaptability of serial manipulators in diverse industrial applications.

**Keywords:** Robots, joints, workspace, forward kinematic, inverse kinematic, optimization Algorithms.

## Contents



# 1. Introduction

*"The reachable workspace of a mechanism is the totality of positions that a particular identified point of the end-effector can reach"*[1]. It is defined by the physical limits of the robot, such as arm length and joints[2], as well as the constraints of the environment in which the robot operates. The workspace can be represented as a three-dimensional volume or a surface in space, depending on the type of movement and the configuration of the robot[3][4]. The value of the workspace lies in its direct relationship with the robot's ability to perform tasks efficiently [5][6].

A well-designed workspace helps optimize robot use by maximizing the area in which it can reach and manipulate objects[7]. This results in greater operational flexibility, increased productivity and better resource management [8].

Applying forward kinematics and inverse kinematics is necessary for serial robots. the forward kinematics allows us to determine the position and the orientation of the end effector, and the inverse determines the angles necessary to achieve the required position and orientation of the end effector as demonstrated in [9] and in [10]. In [11] it is explained how to develop all the surfaces which form the robot workspace starting by the Denavit-Hartenberg representation for serial kinematic chains.

As mentioned in [12] manipulators can be divided in three groups, general manipulators with no singularities, manipulators with one single singularity on the surface S, manipulators with two single singularity on the surface[12] . in [13] a method for identifying the boundary to voids of workspace of serial mechanical manipulators is presented, and in [2] two methods for determining dexterity of serial robotic arms.

in [14] they used a method consisting of isolating a few singularities to determine the workspace boundaries. Workspace optimization has several advantages over other methods. Firstly, it helps maximize the use of available space, which is particularly important in environments where space is limited and valuable, such as factories and production lines, it is possible to avoid collisions between the robot and surrounding obstacles, reducing the risk of property damage and loss of productivity. In addition, it makes it possible to improve the efficiency of the robot's movements, in particular the distances traveled and travel times. This results in shorter work cycles, increased production throughput and more efficient use of resources, such as energy and time. It can help improve the safety of operations, by causing collisions and dangerous situations, it reduces the risk of accidents and injuries to operators and other surrounding equipment. This helps create a safer working environment and ensures compliance with safety standards [15].

As shown in [16] and [17] the length of the link and volume of the robot have a proportional relation with the workspace. In [18] using the Monte Carlo and the multi Island genetic algorithm optimizes the workspace of the manipulator In this study, the optimization of the work space allows us to fully exploit the potential of robots by maximizing their work area, improving the efficiency of operations and guaranteeing the safety of activities which allows the creation of systems high-performance, flexible and secure robotics capable of meeting the changing demands of modern industry.[19] in [20] a modified immigration genetic algorithm based on workspace analysis is applied to reduce the unnecessary computation of the immigration genetic algorithm. a modified differential evolutionary technique is used [21] and had significant results in optimizing the workspace in [22] they were able to achieve a kinematic optimization to maximize the workspace of the robot. in[23] a genetic algorithm was used to optimize the workspace.

## 2. Methodology

The analytical approach used to determine the boundary surfaces of a manipulator starts by the geometric and kinematic characteristics of the space in which the robot can operate. This workspace is defined as the volume of space that the robot's end-effector can reach while staying within the limits of its joints.

- The concept of forward kinematics plays a crucial role in the field of robotics as it focuses on calculating the exact position and orientation of a robot's end-effector, such as a gripper or tool. This calculation is based on the joint angles or joint displacements of the robot. By understanding forward kinematics, we can establish a clear relationship between the robot's joint variables and the pose of its end-effector in the workspace, enabling precise control and manipulation of objects.
- Inverse kinematics refers to the computational process of calculating the angles or displacements of a robot's joints in order to attain a specific position and orientation for its end-effector. This involves solving for the joint variables of the robot based on the desired pose of the end-effector in the workspace, encompassing both its position and orientation.
- Singularities in workspace analysis refer to specific points or regions within the workspace where the robot's Jacobian matrix becomes singular or nearly singular. These singularities pose a challenge to the robot's manipulability, making it difficult to achieve certain end-effector poses or execute specific motions.
- to optimize the workspace of the robot we must select suitable optimization algorithms or techniques for seeking optimal or nearly optimal solutions to the workspace optimization issue. Commonly employed optimization algorithms in workspace optimization encompass genetic algorithms, Monte Carlo methods, gradient-based methods, and particle swarm optimization methods.

## 3. Forward Kinematics

As proven in [24] the set of two parallel rectangular beams have the highest structural strength which is used in this paper like in the following figure (1):

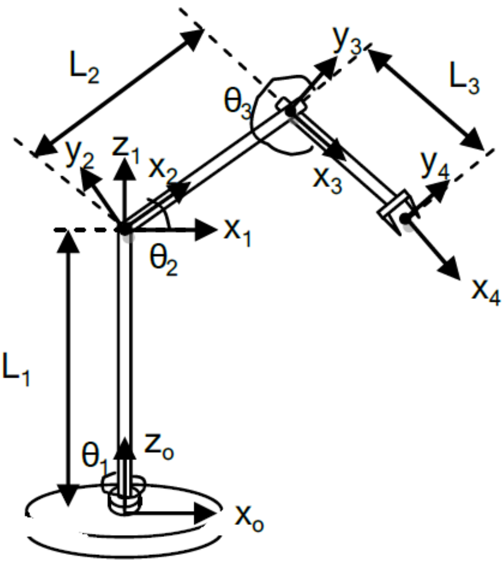


Figure 01: frames of a 3 dof serial robot

Assigning the frames helps to visualize the robot's motion, furthermore to describe the position and orientation of different components of the robot as shown To create the matrix connecting two transformations, we only need a concise set of four parameters to describe how one coordinate system aligns with another. These parameters, known as Denavit-Hartenberg (DH) parameters [25], offer a streamlined approach to depict the connection between neighboring links in a systematic manner.and based on that we find the following parameters :

an each term of the DH- parameters is defined as such:

1. $\alpha$ Link twist is the angle about Xi between $(Z_i, Z_{i+1})$
2. $a_i$ : Link length is the distance along $X_i$ between $(Z_i, Zi+1)$
3. $d_i$ : Link offset is the distance along $Z_i$ between $(X_{i-1}, X_i)$
4. $\theta_i$ : Joint angle is the angle about $Z_i$ between $(X_{i-1,Xi})$

The homogeneous transformation matrix $T_i^0$, which defines the orientation and position of the $i_{th}$ frame relative to the base coordinate system, is obtained by multiplying the successive transformation matrices from $T_{i-1}^{i}$ [11].

$$T_i^0 = \prod_{j=1}^{i} T_j^{\,j-1}3 \tag{1}$$

$$T_i^{i-1} = \begin{bmatrix} R_i^0 & P_i^0 \\ 0 & 1 \end{bmatrix} \tag{2}$$

where:

1. $R_i^0$: rotational matrix from the $i_{th}$ coordinate frame to $0^{th}$ coordinate frame.
2. $P_i^0$: the position vector with respect to the $0^{th}$ coordinate frame.

the general formula of $T_i^0$ is given by

$$T_i^{i-1} = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & \alpha_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & -s\alpha_{i-1}d_i \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & c\alpha_{i-1}d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3}$$

where $c = cos$ and $s = sin$

therefore applying to our robot we find:

$$T_4^0 = T_1^0.T_2^1.T_3^2.T_4^3 \tag{4}$$

from 3 and 4 we find:

$$T_4^0 = \begin{bmatrix} c_1c_{23} & -c_1s_{23} & s_1 & c_1c_{23}.L_3 + c_1c_2L_2 \\ s_1c_{23} & -s_1c_{23} & -c_1 & s_1c_{23}.L_3 + s_1c_{23}L_2 \\ s_{23} & c_{23} & 0 & s_{23}.L_3 + s_2L_2 + L_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5}$$

the last column represents the Cartesian coordinates of the position of the origin of the end effector frame 4 with respect to the reference frame 0.

$$\begin{cases} x = c_1c_{23}.L_3 + c_1c_2L_2 \\ y = s_1c_{23}.L_3 + s_1c_{23}L_2 \\ z = s_{23}.L_3 + s_2L_2 + L_1 \end{cases} \tag{6}$$

by finding the expressing of $(x, y, z)$ in 6 we have achieved the purpose of forward kinematics

# 4. Inverse kinematics

Now by using equation 6 we will try to derive the expression of $(\theta_1, \theta_2, \theta_3)$

$$\frac{y}{x} = \frac{c_1c_{23}.L_3 + c_1c_2L_2}{s_1c_{23}.L_3 + s_1c_{23}L_2} = \frac{s_1}{c_1} \tag{7}$$

$$\frac{y}{x} = tan\theta_1 \tag{8}$$

so:

$$\theta_1 = arctan2(\frac{y}{x}) \tag{9}$$

as for $\theta_3$ we start by summing the squares of 6 find :

$$c_3 = \frac{x^2 + y^2 + z^2 - (L_1^2 + L_2^2 + L_3^2)}{2L_2L_3} \tag{10}$$

using the Pythagorean trigonometric identity:

$$s_3 = \pm\sqrt{1 - c_3} \tag{11}$$

$$\theta_3 = 2.atan\frac{s_3}{c_3} \tag{12}$$

and for $\theta_2$ we found:

$$\theta_2 = atan2[\frac{(z - L_1)(c_1 - s_1)}{x - y}] - atan2[\frac{(s_3L_3}{c_3L_3 + L_2}] \tag{13}$$

the explicit procedure of how we got this result is well explained in [24]

# 5. Workspace Identification

The workspace can be identified analytically using the equations obtained from the forward kinematics calculations. Since $-1 \leq cos\theta \leq 1$ which means that:

$$-2L_2L_3 \leq x^2 + y^2 + z^2 - 2L_1z + 2L_1^2 - (L_1^2 + L_2^2 + L_3^2 \leq 2L_2.L_3 \tag{14}$$

simplifying 14 :

$$(L_2 - L3)^2 \leq x^2 + y^2 + (z - L_12)2 \leq (L_2 + L3)^2 \tag{15}$$

we notice that the result is a sphere of a center at $(0, 0, L_1)$, with an inner radius of $(l_2 - L_3)$ and an outer radius of $(l_2 + L_3)$, the workspace is generated in Matlab and shown in the figure where is the red surface represents the boundaries of the workspace as shown in figure (2).

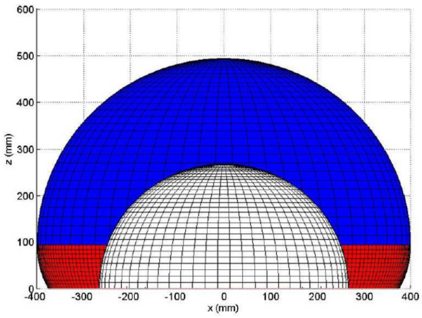


figure 02: workspace simulation

## 5.1. Jacobien

The Jacobian of the position vector is important to understand the robot workspace and its limitations, by identifying singularities in the Jacobian matrix. It's also needed for the workspace optimisation, by identifying optimal joint configurations that maximize reachability and manipulability

The position vector generated by a point on the end-effector of a serial robot is expressed by;

$$G(q) = \begin{bmatrix} x(q) & y(q) & z(q) \end{bmatrix} \tag{16}$$

where we have $G(q) \in R^3$ and $q = \begin{bmatrix} q_1 & q_2 & q_3 \end{bmatrix}$ which G(q) represents all the achievable points of the workspace

The equation of the Jacobian $J$ for a 3-DOF serial robot is given by:

$$J = \begin{bmatrix} \frac{\partial x}{\partial q_1} & \frac{\partial x}{\partial q_2} & \frac{\partial x}{\partial q_3} \\ \frac{\partial y}{\partial q_1} & \frac{\partial y}{\partial q_2} & \frac{\partial y}{\partial q_3} \\ \frac{\partial z}{\partial q_1} & \frac{\partial z}{\partial q_2} & \frac{\partial z}{\partial q_3} \end{bmatrix} \tag{17}$$

where $x$, $y$, and $z$ represent the coordinates of the end-effector of the robot, and $q_1$, $q_2$, and $q_3$ are the joint angles

# 6. optimization methods

## 6.1. the genetic algorithm

Genetic algorithms are designed to solve optimization problems, specially those where the work space is complicated. they are widely used in various fields, including engineering, data science, parameter optimization, complex system design, and many others, due to their ability to find effective solutions in complex and multidimensional search spaces.

In the context of optimizing robot workspace, the optimization process typically begins with defining a set of potential workspace configurations or parameters, which can include factors such as joint angles, link lengths, and end-effector positions. Each configuration represents a potential solution to optimizing the robot's workspace.

The genetic algorithm evolves this initial set of workspace configurations through multiple iterations. At each iteration, the algorithm applies selection, crossover, mutation, and natural selection operations to refine the workspace configurations. to further explain we have the following steps:

1. Selection: Workspace configurations that lead to improved workspace performance metrics, such as increased reachability or reduced interference with obstacles, are selected for further exploration.
2. Crossover: Pairs of selected workspace configurations are combined to create new configurations by exchanging components of their parameters. This exchange allows the algorithm to explore new combinations of parameters that may lead to better workspace optimization.
3. Mutation:Occasionally, random changes are introduced to the parameters of selected workspace configurations to promote diversity in the population of solutions and prevent convergence to local optima.
4. Natural selection: Less effective workspace configurations are discarded, while the most promising configurations are retained for subsequent iterations.

This iterative process continues until a termination condition is met, such as a maximum number of iterations, insufficient improvement in the solution, or convergence to an optimal solution. Ultimately, the genetic algorithm provides a solution or an approximation of a solution to the optimization problem.

in the following figure (03) we visualize the process of a GA:

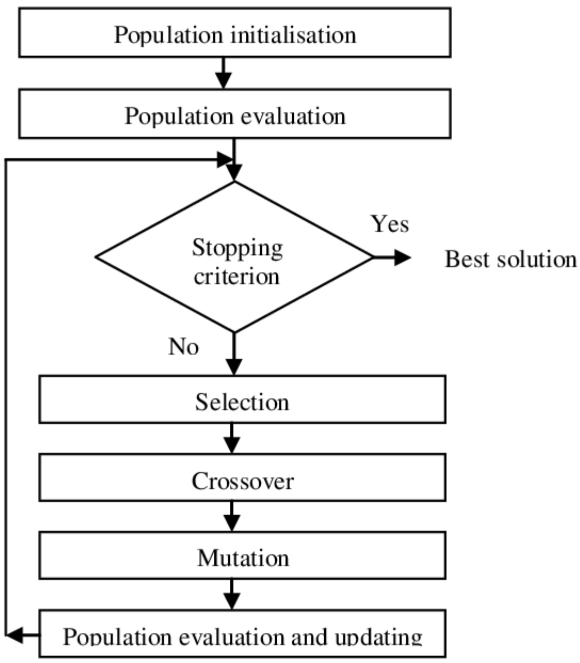


figure 03: Process of the Genetic Algorithm

## 6.2. Proces of Particle Swarm Optimization

Applying the Particle Swarm Optimization (PSO) to optimize the workspace of a robot involves finding configurations of the robot's joints or parameters that maximize the reachable workspace [26]. which usually needs the following steps:

- Termination Criteria:: Define the termination criteria for the PSO algorithm, which determines when the optimization process should stop. This could be based on a maximum number of iterations, a threshold for the improvement in fitness value, or a combination of both.

- Convergence Analysis: Monitor the convergence of the PSO algorithm by tracking the fitness values of the particles over iterations. Analyze the convergence behavior to ensure that the algorithm is progressing towards finding optimal or near-optimal solutions for the robot's workspace.
- Validation and Testing: Validate the optimized robot configurations by conducting simulations or physical experiments to verify their performance in the real-world environment. Test the robot's reachability, manipulability, and other workspace characteristics to ensure that the optimization process has indeed improved the robot's capabilities.
- Fine-tuning and Iteration: If the obtained results are not satisfactory, consider fine-tuning the PSO algorithm parameters or revisiting the objective function definition to better capture the desired workspace characteristics. Iterate the optimization process until the desired level of performance is achieved.
- Application and Deployment: Once the optimized robot configurations have been validated and fine-tuned, deploy them in practical applications where the improved workspace can enhance the robot's performance and efficiency. This could include industrial automation, robotic surgery, or any other field where precise and optimized robot movements are crucial. .

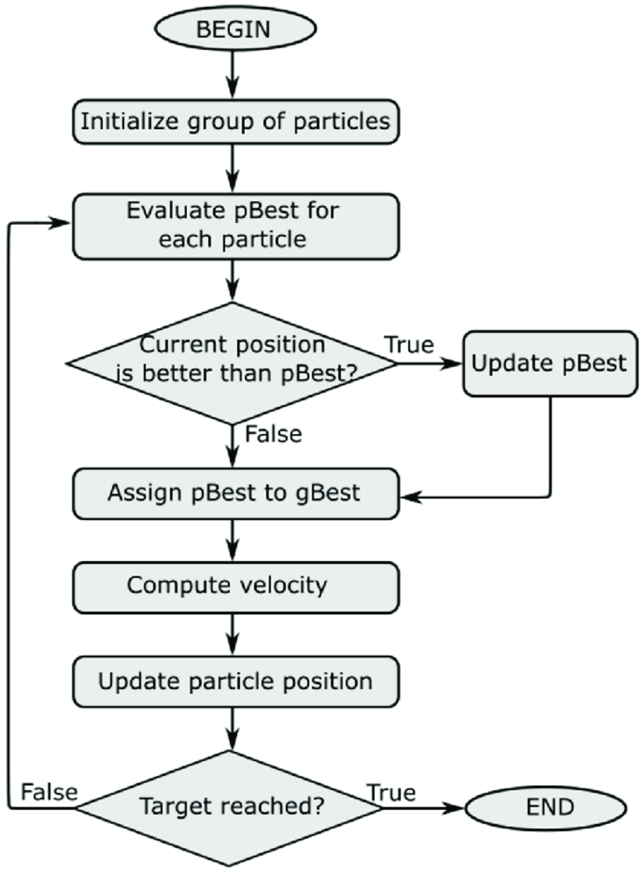


figure 04: Proces of Particle Swarm Optimization algorithm

## 6.3. Monte Carlo Optimisation Algorithm

Monte Carlo optimization utilizes random sampling techniques to approximate solutions for optimization problems, diverging from deterministic algorithms by generating random samples and evaluating their performance iteratively [18].

This method is beneficial for complex or high-dimensional search spaces where traditional methods struggle, efficiently exploring such spaces by sampling numerous points and evaluating their performance to provide a flexible and scalable optimization approach as shown in figure 6.3.

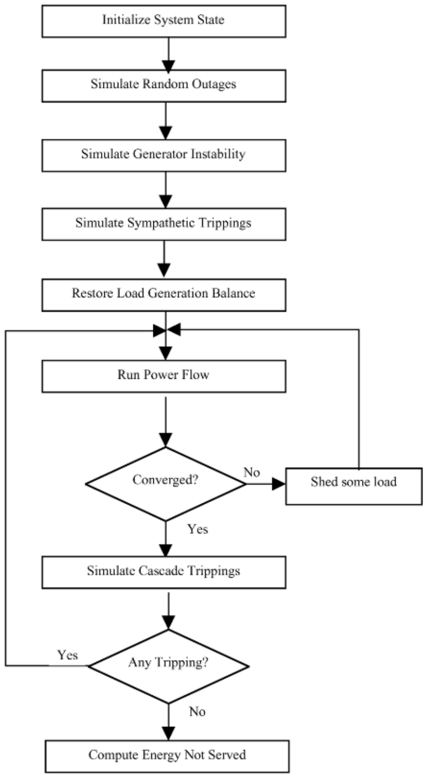


figure 05: Process of the monte carlo Algorithm

# 7. Discussion

When optimizing the workspace for serial manipulators, each algorithm has its strengths. Genetic Algorithms (GA) are highly adaptable to different problems but are complex and moderately fast. Particle Swarm Optimization (PSO) is simpler to implement, converges quickly, and is also highly adaptable. Monte Carlo (MC) methods are the easiest to implement but can be slower and require many samples for accuracy. The best choice depends on the specific needs, balancing complexity, speed, and adaptability.

- Criteria:
  - GA algorithm: natural selection and genetics
  - PSO algorithm: Simulates flocking behavior
  - MC algorithm: random sampling
- Optimization Type:
  - GA algorithm: Global optimization
  - PSO algorithm: Global optimization
  - MC algorithm: Stochastic optimization
- Convergence Speed:
  - GA algorithm: Moderate
  - PSO algorithm: Fast
  - MC algorithm: Variable
- Complexity:
  - GA algorithm: High
  - PSO algorithm: Moderate
  - MC algorithm: Simple
- Adaptability:
  - GA algorithm: Highly adaptable to different problems
  - PSO algorithm: Highly adaptable to different problems
  - MC algorithm: Adaptable but may require many samples for accurate results
- Ease of Implementation:
  - GA algorithm: Moderate to high complexity
  - PSO algorithm: Easy
  - MC algorithm: Easy

## 7.1. kinematic modeling and optimization algorithms

Kinematic modeling and optimization algorithms are closely related in the context of serial manipulators. Kinematic modeling provides a mathematical description of the manipulator's motion, which includes the relationships between joint parameters and the end-effector's position and orientation. This model is crucial for defining the workspace and constraints that the optimization algorithm will work with.

- Genetic Algorithms (GA): Kinematic models help establish the fitness functions used in GA. By simulating various configurations of the manipulator, GAs can explore a vast solution space to find optimal configurations that maximize the workspace or minimize certain kinematic constraints.
- Particle Swarm Optimization (PSO): PSO algorithms benefit from the kinematic models to evaluate the performance of each particle in the swarm. The kinematic model ensures that each potential solution (particle) adheres to the manipulator's physical constraints and achieves the desired optimization goals.
- Monte Carlo (MC) Methods: In MC methods, kinematic models are used to randomly sample configurations within the manipulator's operational space. The kinematic model ensures that the sampled configurations are valid, and the optimization process can then identify regions of the workspace that meet specific criteria.

# 8. Conclusions

these paper has presented an analysis of three degree of freedom (3dof) serial robot in which the forward and inverse kinematic analyses were carried out to established the workspace. furthermore, a set of optimization algorithms of workspace were explained. In summary, kinematic modeling defines the search space and constraints for optimization algorithms, ensuring that the solutions found are physically feasible and optimal for the manipulator's performance. While this paper did not fully covered all the aspects about the workspace analysis and optimization, but it did gave a better understanding about it for future references


## Acknowledgements

I would like to express my sincere gratitude to my advisor for their guidance and support throughout this research. I am also grateful to the staff at GIM Department for their help.


## References


[1] Jingshan Zhao, Zhijing Feng, Fulei Chu, and Ning Ma. *Advanced theory of constraint and motion analysis for robot mechanisms*. Academic Press, 2013.

[2] KA Abdel-Malek and HJ Yeh. Local dexterity analysis for open kinematic chains. *Mechanism and Machine Theory*, 35(1):131–154, 2000.

[3] Taimoor Zahid, Zareena Kausar, Muhammad Faizan Shah, Muhammad Tallal Saeed, and Jianfei Pan. Workspace design and trajectory planning of a five degree of freedom mobile welding manipulator for spherical objects. *IEEE Access*, 9:73843–73856, 2021.

[4] Donghun Lee, TaeWon Seo, and Jongwon Kim. Optimal design and workspace analysis of a mobile welding robot with a 3p3r serial manipulator. *Robotics and Autonomous systems*, 59(10):813–826, 2011.

[5] M Ceccarelli. Feasible workspace regions for a two-revolute manipulator design. In *Recent Advances in Robot Kinematics*, pages 189–198. Springer, 1996.

[6] H Zhang and P Sikka. Characterization of the workspace for planar robot manipulators. In *[1991] IEEE Pacific Rim Conference on Communications, Computers and Signal Processing Conference Proceedings*, pages 158–161. IEEE, 1991.

[7] Chih-Chung Daniel Lin and Ferdinand Freudenstein. Optimization of the workspace of a three-link turning-pair connected robot arm. *The International journal of robotics research*, 5(2):104–110, 1986.

[8] Badreddine Aboulissane, Larbi El Bakkali, and Jalal El Bahaoui. Workspace analysis and optimization of the parallel robots based on computer-aided design approach. *Facta Universitatis, Series: Mechanical Engineering*, 18(1):079–089, 2020.

[9] Dongjing Li, Wei Wang, Qilong Wang, and Daxian Hao. Polishing robot structure optimization based on workspace analysis. In *2016 Asia-Pacific Conference on Intelligent Robot Systems (ACIRS)*, pages 52–56. IEEE, 2016.

[10] Anatoly Pashkevich, Damien Chablat, and Philippe Wenger. The orthoglide: Kinematics and workspace analysis. In *On Advances in Robot Kinematics*, pages 237–246. Springer, 2004.

[11] Khushdeep Goyal and Davinder Sethi. An analytical method to find workspace of a robotic manipulator. *Journal of mechanical engineering*, 41(1):25–30, 2010.

[12] Ottaviano Erika, Husty Manfred, and Ceccarelli Marco. Level-set method for workspace analysis of serial manipulators. 2006.

[13] Karim Abdel-Malek, Harn-Jou Yeh, and Saib Othman. Interior and exterior boundaries to the workspace of mechanical manipulators. *Robotics and Computer-Integrated Manufacturing*, 16(5):365–376, 2000.

[14] Oriol Bohigas, Montserrat Manubens, and Lluís Ros. A complete method for workspace boundary determination on general structure manipulators. *IEEE Transactions on Robotics*, 28(5):993–1006, 2012.

[15] YJ Lou, GF Liu, and ZX Li. A general approach for optimal design of parallel manipulators. In *IEEE International Conference of Robotics and Automation (ICRA)*, volume 26, 2005.

[16] S Kucuk and ZAFER Bingul. Robot workspace optimization based on a novel local and global performance indices. In *Proceedings of the IEEE International Symposium on Industrial Electronics, 2005. ISIE 2005.*, volume 4, pages 1593–1598. IEEE, 2005.

[17] Alexandru Oarcea, Florin Popister, Sergiu-Dan Stan, and Victor Cobilean. Comparative study of cad optimization features for the workspace of 3dof parallel robot. In *2021 9th International Conference on Modern Power Systems (MPS)*, pages 1–6. IEEE, 2021.

[18] ZENG Qingliang, ZHOU Guangyu, SHI Jifei, LI Yongqing, LIU Yuanqiang, CHEN Tianming, and WANG Liang. Robot workspace optimization based on monte carlo method and multi island genetic algorithm. *Mechanics*, 28(4):308–316, 2022.

[19] Yi Cao, Suiping Qi, Ke Lu, Yi Zang, and Guanying Yang. An integrated method for workspace computation of robot manipulator. In *2009 International Joint Conference on Computational Sciences and Optimization*, volume 1, pages 309–312. IEEE, 2009.

[20] Chun-Tse Lee and Jen-Yuan Chang. A workspace-analysis-based genetic algorithm for solving inverse kinematics of a multi-fingered anthropomorphic hand. *Applied Sciences*, 11(6):2668, 2021.

[21] S Panda, D Mishra, BB Biswal, and BB Choudhury. Robot workspace optimization using a novel modified differential evolutionary technique. *International Journal of Computational Methods*, 9(03):1250034, 2012.

[22] S Stan. Workspace optimization of a two degree of freedom mini parallel robot. In *2006 IEEE International Conference on Automation, Quality and Testing, Robotics*, volume 2, pages 278–283. IEEE, 2006.

[23] Yuanhai Huang, Zhi Li, Kefan Xing, and Haoqin Gong. A manipulator size optimization method based on dexterous workspace volume. *Cobot*, 1:3, 2022.

[24] Madiha Farman, Muneera Al-Shaibah, Zoha Aoraiath, and Firas Jarrar. Design of a three degrees of freedom robotic arm. *International Journal of Computer Applications*, 179(37):12–17, 2018.

[25] Jacques Denavit and Richard S Hartenberg. A kinematic notation for lower-pair mechanisms based on matrices. 1955.

[26] Vincensius Billy Saputra, Soh-Khim Ong, and Andrew YC Nee. A pso algorithm for mapping the workspace boundary of parallel manipulators. In *2010 IEEE International Conference on Robotics and Automation*, pages 4691–4696. IEEE, 2010.

# A. Annex

The computation of Jacobian singularities has been expanded to include the computation of sets of singular parameters. This new formulation allows for the computation of the boundary of any manipulator with any number of independent variables or degrees of freedom. Additionally, the Jacobian method has been extended to handle non-square Jacobians through a row rank deficiency method. In this general formulation, the resulting Jacobian is not square, and three types of singular behavior can be observed. Singular behavior refers to geometric entities within the accessible set that pose difficulties in satisfying equation5.1. If the Jacobian is a square matrix, the singularities can be computed by setting the determinant of the Jacobian equal to zero. therefore

$$0 = \begin{bmatrix} \frac{\partial x}{\partial q_1} & \frac{\partial x}{\partial q_2} & \frac{\partial x}{\partial q_3} \\ \frac{\partial y}{\partial q_1} & \frac{\partial y}{\partial q_2} & \frac{\partial y}{\partial q_3} \\ \frac{\partial z}{\partial q_1} & \frac{\partial z}{\partial q_2} & \frac{\partial z}{\partial q_3} \end{bmatrix} \tag{18}$$

By solving equation 5.1 with respect to the independent variables we get the following singularites,

- Rank-deficiency singularity set: this singularity occurs when

$$S_1 = p \in R^n; dimNull(\widehat{J}) \geq 1.forsomeq\} \tag{19}$$

  The constant generalized coordinates subset p from q makes the analytic Jacobian singular, while Null represents the dependent rows in the matrix $\hat{\mathbf{J}}$.

- Rank-deficiency of reduced-order accessible set When reaching the boundary of a surface, the edge of a geometric entity is swept. In order to determine these entities, the accessible set needs to be reduced by substituting a value for one of the parameters. Both upper and lower limits are taken into consideration. Once the set is reduced, the null space criteria of equation5.1 is applied. Each variable in the set is constrained by upper and lower limits, By substituting a limit (qi min or qi max) into the wrist point, the position vector for the reduced order set is obtained, resulting in the set losing one degree of freedom by fixing one link.

$$S_2 = p \in R^n; dimNull(\widehat{J}) \geq 1.forsomeq^*\} \tag{20}$$

- **Constraint singularity set: boundary is reached when the number of constant parameters equals the dimensions of $X_w^n$ or $P_n^0$. Entities that are due to active parameter constraints are called constraint singularity sets and are defined by:**

$$S_3 = p \in R^n : \begin{bmatrix} q_i^0 & q_j^0 \end{bmatrix} for i, j = 1 to n; i \neq j \Big\} \tag{21}$$

  **The constraint singularity set is a combination of all constant parameters, i.e. for each combination of the limits of the constraints, there exists a hyperentity.**

- **total singularities set is the combinition of the previous ones and are given by:**

$$S = S_1 \cup S_2 \cup S_3 \tag{22}$$