applied sciences

MDPI

Type of the Paper (Article, Review, Communication, etc.)

Numerical method using homotopic iterative functions based on the via point for the joint-space trajectory generation

Yadira Quiñonez [1], Oscar Zatarain [1,*], Carmen Lizarraga [1,*], Raquel Aguayo[1], Jezreel Mejía [2]

1 Universidad Autónoma de Sinaloa; yadiraqui@uas.edu.mx, oscarzatarain@uas.edu.mx, carmen.lizarraga@uas.edu.mx, raquelaguayog@uas.edu.mx
2 Centro de Investigación en Matemáticas ; jmejia@cimat.mx
* Correspondence: oscarzatarain@uas.edu.mx; carmen.lizarraga@uas.edu.mx

Abstract: In recent years, many methods have been developed to calculate the trajectory of a robotic arm in the joint-space. These methods have many advantages, such as soft motion and infinite jerk avoidance. Nevertheless, these methods present other problems that must be avoided, such as unnatural motion while generating the trajectory and producing unsafe planning. In this sense, this work presents a numerical method named iterative optimal solution trajectory via $(\zeta)_v$-homotopy former (IOSTV $(\zeta)_v$-HF). It is proposed to reduce and avoid oscillation while getting trajectories with different shapes to perform better, reliable, smooth, and long-life robotic systems. The algorithm with the proposed method is described, and examples of the trajectories obtained with different parameters are presented. In addition, these were mapped and a trajectory with continuous velocity and a reduced oscillation and another trajectory with the same restrictions but with continuous acceleration and zero oscillations were shown; the method is versatile since it allows choosing and finding the most optimal solutions according to the application. Finally, the article ends with a critical discussion of the experimental results.

Keywords: Homotopy; joint-space; moment function generation; trajectory; via point; velocity, acceleration, jerk, oscillation avoiding, Matlab.

Citation: Lastname, F.; Lastname, F.; Lastname, F. Title. *Appl. Sci.* **2022**, *12*, x. https://doi.org/10.3390/xxxxx

Academic Editor: Firstname Lastname

Received: date
Accepted: date
Published: date

1. Introduction

In recent decades, robot applications have been extensively studied [1], and numerous improvements have been developed [2]. These advances are becoming more robust today since they mainly focus on work with repetitive tasks to increase productivity, such as industrial applications [3, 4] and applications in the medical area [5]. Therefore, robot arms (manipulators) must be precise to be used by these applications and many others [6]. Usually, these robots can work in dangerous environments, in places where humans cannot access to perform dangerous tasks [7]; at the same time, these robot arms must navigate obstacles because the environment of a robotic arm is often very complicated.

For these reasons, the robotic arm's motion must be precise and fulfill some specific characteristics that are defined depending on the environment and the application. In addition, it is necessary thoroughly study trajectories and kinematics (direct and inverse) to verify that the robotic arm does not show any complications while performing a motion [8]. For example, the most famous methods to calculate the trajectory of a robotic arm [9, 10] are cubic polynomials [11], trapezoidal trajectory [12], and the Euler angles [10], to mention some. Besides, many studies, methods, algorithms, and designs (electronic and mechanical) help to obtain better performance in robotic arm motion [13, 14, 15, 16, 17, 18, 19]. Nevertheless, some of these methods are complicated to implement. Others are feasible in exhibiting unnatural motion, infinite jerk (third time derivative of position), or require other resources, such as an optimal timing solution or specific PID (proportion-al-integral-derivative) control to generate the optimal trajectory.

For example, the LSPB method (Linear Segment with Parabolic Blends) has improved trajectory performance [20]. However, it requires more calculations. This is because the LSPB divides the trajectory into segments and may only sometimes produce the most optimal solution in terms of executed time [21]. Also, using this method can result in undesirable acceleration profiles, limiting the flexibility of the resulting motion [22]. Concerning n-order polynomials also have some disadvantages; according to some authors [11, 23, 24, 25], higher-order polynomials are more complex and may require more computation to evaluate, which can be a drawback in real-time control systems. In addition, it presents local minima or maxima, which can lead to unexpected or undesired behavior if the trajectory is not carefully designed. Finally, this method may not be as flexible as other types of trajectories, such as splines, in accurately following the desired path. In summary, high-order polynomials must be carefully designed using other techniques to optimize this method.

In another example, the authors in [26] present trajectories with a total time of 11 seconds using a radial basis function (RBF) neural network. This method is reliable and an excellent option for performing motion planning. However, the trajectories present many big-long oscillations through time, which means greater energy consumption and unsafe motion planning to avoid collisions [27]. The 3-5-3 interpolation polynomial method presented in [28] has the same problem mentioned before in [26].

Other works [29, 30, 31, 32] use numerical methods with homotopy continuation to generate optimal trajectories in manipulators. The authors mentioned that this approach is a favorable option for generating trajectories because these are versatile and fulfill the characteristics of the mechanical system. Other related works that use homotopic functions for optimal trajectory planning focus mainly on mobile robots [33], humanoid robots [34], dynamics, and control problems [35].

In previous work, a novel algorithm introduced in [36, 37] was presented as a trajectory planning approach with more characteristics that had not been mentioned before and others that had been overlooked. This algorithm generated homotopic trajectories that always start in the specified start position and ends at the final point, generating enough iterations to make trajectories that go closer and closer to the desired via point each time. The algorithm generates the ideal trajectory (the trajectory that passes through the specified via point) with infinite iterations and prelaminar parameters that determine the shape of the trajectory desired. However, a finite quantity of iterations gets an excellent approximation and can be as accurate as desired.

Furthermore, the method proposed in this work makes it possible to obtain better trajectory performance by changing the shape of the velocity, acceleration, and jerk profiles. In this sense, it has coined this algorithm with the name iterative optimal solution trajectory via $(\zeta)_v$-homotopy former (IOSTV $(\zeta)_v$-HF). In addition, with this algorithm, some of the disadvantages presented before are lost. For example, generating many options for getting different trajectory shapes with the same initial, via, and final point makes it possible to obtain the best suitable trajectory for specific applications. Furthermore, the IOSTV-HF method is flexible because it always gets a trajectory that passes through these three points in a defined time, and many options can be generated. Also, oscillations can be reduced or removed by applying the same process and changing initial parameters to generate many different shapes of trajectories. In summary, this method presents versatility as the main characteristic. Many unique advantages introduced through this work have been given to help obtain better reliable, smooth, and long-life robotic systems.

The remainder of this paper is organized as follows: Section 2 introduces the preliminary properties that were taken to construct the trajectory function and define it. Section 3 describes the algorithm and the algorithm's proof and shows examples to generate a trajectory that converges to the via point. Section 4 presents the results by generating trajectories with the same initial via point and final position; and the obtained trajectory with its velocity, acceleration, and jerk. Section 5 mentions a critical discussion of the results obtained with the proposed algorithm (IOSTV $(\zeta)_v$-HF) compared to the 6th-order polynomial method. Finally, section 6 summarizes the conclusion of the work and indicates further work.

2. Preliminary properties

2.1. Nomenclature

n: fixed constant greater to 1 $(n > 1)$.
a: velocity, acceleration, and Jerk modifier parameter (it is fixed).
t_s: final trajectory time.
S_f: final position.
q_0: initial position.
ζ_i: objective parameter outcome in the i^{th} iteration.
n_v: fixed constant greater to 1 $(n_v > 1)$.
a_v: velocity, acceleration, and Jerk modifier parameter (it is fixed).
t_v: via point time.
θ_v: via point.
S_{v_i}: feedback iterative sequence

This method presented in [35] has been tested recently to obtain velocity, acceleration, and jerk with a smaller gap in the via point time (t_v). Although this method presents non-continuous derivatives, these remain finite without showing inconvenience. Moreover, the resulting trajectory can be changed to avoid or reduce unnatural oscillation, no matter how short or long the distance is. This is possible because of the construction of the function below:

$$S_{\zeta a t_s}(t) = \begin{cases} q_0 \; if \; t \le 0 \\ \dfrac{n^{t-\zeta}}{\left|n^{t+\frac{t_s}{t}-1} - n^a\right|} + q_0 \; if \; 0 < t \le t_s \; and \; q_0 \le S_f \\ -\dfrac{n^{t-\zeta}}{\left|n^{t+\frac{t_s}{t}-1} - n^a\right|} + q_0 \; if \; 0 < t \le t_S \; and \; S_f < q_0 \end{cases}. \quad (1)$$

This function has been shown in [26] and it was used to generate point-to-point trajectories, but in this work, (1) is added to generate via point trajectories as well (this is explained deeply in section 4).

The ideas behind the way this function was constructed are simple. First, we know that any function with the form $n^{t-\zeta}$, where $n > 1$, can reach any positive or negative point (when this function is negative $-n^{t-\zeta}$) monotonically increasing or decreasing, respectively, by just calculating ζ at any particular time. Still, it cannot start in any position chosen (when $t > 0$). Nevertheless, this is solved by multiplying this function with the following $\left(\left|n^{t+\frac{t_s}{t}-1} - n^a\right|\right)^{-1}$. Then, note that:

$$\lim_{t \to 0^+} \pm \frac{n^{t-\zeta}}{\left|n^{t+\frac{t_s}{t}-1} - n^a\right|} = 0. \quad (2)$$

Once it is verified that the calculation of the function (1) fulfills the essential requirement to perform a trajectory with any particular start and final position, the parameter ζ is required to generate the trajectory starting and ending at a particular point; the formula is presented in [36, 37], but this time, the case $S_f = q_0$ is added in the definition of ζ. Then:

$$\zeta = \begin{cases} t_s - \log_n\left[\left(S_f - q_0\right)\left(n^{t_s} - n^a\right)\right] \; if \; S_f > q_0 \\ t_s - \log_n\left[\left(q_0 - S_f\right)\left(n^{t_s} - n^a\right)\right] \; if \; q_0 > S_f \\ \infty \; if \; S_f = q_0 \end{cases} \tag{3}$$

2.1. Further properties

Another point to consider using this method is uniqueness, more specifically, getting more properties that can help get a better understanding and use them to get the best solution. So, another remarkable characteristic of this function is that it can also be considered a generating probability function and inherit its features. For example, consider the following probability function where t is any parameter such that $t \in [0, t_s]$ and $x = 1,2,3, \ldots$

$$f(x,t) = \begin{cases} 0 \; if \; t \leq 0, \\ \left(\dfrac{n^{(t-\zeta)}}{\left|t\left(n^{t+\frac{t_s-t}{t}} - n^a + n^{t-\zeta}\right)\right|}\right)^x \; if \; 0 < t \leq t_s \end{cases}. \tag{4}$$

The following is calculated using the definition of probability generating function. The parameter t is considered the same parameter t from $f(x,t)$ and the definition of probability generating function:

$$\sum_{x=0}^{\infty} t^x f(x), \tag{5}$$

Then:

$$\begin{aligned} G(t) &= \sum_{x=0}^{\infty} t^x f(x,t) = \sum_{x=0}^{\infty} t^x \left(\frac{n^{(t-\zeta)}}{\left|t\left(n^{t+\frac{t_s-t}{t}} - n^a + n^{t-\zeta}\right)\right|}\right)^x \\ &= \sum_{k=1}^{\infty} t^x \left(\frac{\left(n^{t-\zeta}\right)^x}{\left(\left|t^x\right|\left|n^{t+\frac{t_s-t}{t}} - n^a + n^{t-\zeta}\right|\right)^x}\right) \\ &= \frac{n^{t-\zeta}}{\left(-\dfrac{n^{t-\zeta}}{\left|n^{t+\frac{t_s-t}{t}} - n^a + n^{t-\zeta}\right|} + 1\right)\left(\left|n^{t+\frac{t_s-t}{t}} - n^a + n^{t-\zeta}\right|\right)} \\ &= \frac{n^{t-\zeta}}{\left(\left|-n^{t-\zeta} + n^{t+\frac{t_s-t}{t}} - n^a + n^{t-\zeta}\right|\right)} = \frac{n^{t-\zeta}}{\left(\left|n^{t+\frac{t_s-t}{t}} - n^a\right|\right)}. \end{aligned} \tag{6}$$

Function (6) is a probability generating function for when $\zeta = 1 - \log_n\left|n^{t_s} - n^a\right|$ and because it must fulfill that $\lim\limits_{t \to 1^-} G(t) = 1$. Taking into account the example of Figure 1, $n = 2, t_s = 3, a = 0, S_f = 4$ and $q_0 = 0$, a trajectory is obtained being $S_{\zeta a t_s}(t)$, function (1), a probability generating function:

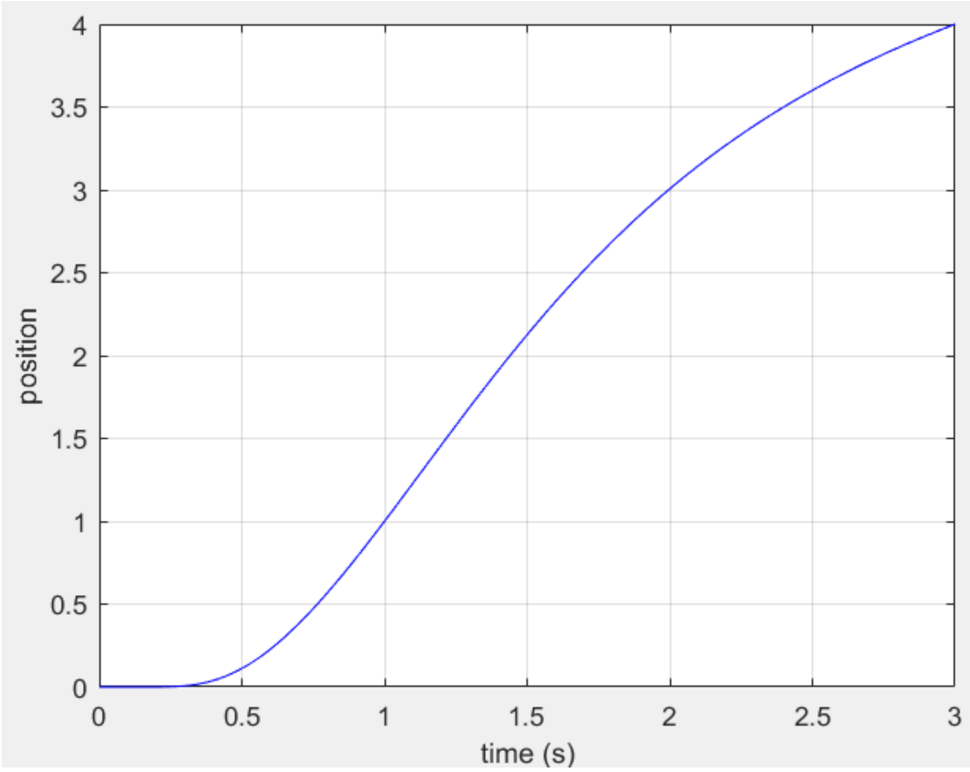

Figure 1. trajectory example where $S_{\zeta a t_s}(t)$ is a probability generating function with parameters $n = 2, t_s = 3, a = 0, S_f = 4$ and $q_0 = 0$.

Making $S_{\zeta a t_s}(t)$ a Probability-Generating Function (PGF) can give us many benefits that will be discussed in future works. For example, one of these remarkable benefits is that: a PGF could potentially be used as a tool to analyze the probability distribution of the joint space trajectory, which could be useful in understanding the characteristics of the trajectory and optimizing the performance and reliability of the robot arm by providing a way to analyze and understand the probability distribution of the joint angles as the arm moves through its range of motion. For example, it can be used to know an average velocity through the trajectory for better performance and then choose a suitable motor that can work with this velocity average, but this must be studied deeply. Also, it could be interesting to note that, without losing generality, if $S_{\zeta a t_s}(t)$ is a PGF, it is easy to see that every derivative starts at zero when $t = 0$ (since $f(x, 0) = 0$ and the preposition that indicates for every generating function to occur that $P(X = x) = \frac{1}{x!} G^x(0)$), which is impossible for some current methods. Any trajectory with a start and final position it can be obtained by calculating ζ, as mentioned before, with its derivatives starting at 0 (when $t = 0$). So, these are some prelaminar ideas that were considered to construct function (1) and to avoid some disadvantages presented in the current works stated in the introduction part of this article.

3. Algorithm description

The trajectory function obtained by applying the method (IOSTV $(\zeta)_v$-HF) is denoted as $S_{i(\zeta a t_s)_v}(t)$, where i represents the number of iterations. While these iterations increase, the trajectory gets closer and closer to the via point, forming a homotopy with a family of functions with different parameters ζ_i and ζ_{v_i} calculated in every iteration until a solution for these two parameters fits to generate the desired via-point trajectory. This happens because ζ_i and ζ_{v_i} converge when i tends to infinity ($i \to \infty$) by calculating the sequence S_{v_i} for every iteration, making $S_{i(\zeta a t_s)_v}(t)$ passes through the specified via point in the desired via point time (t_v). Then, the function is defined as $S_{i(\zeta a t_s)_v}(t)$: $[0, t_s] \to$

$\mathbb{R}$ and with the following conditions (S_{v_i} is defined in (9)), each condition is provided with an example:

$$S_{i(\zeta a t_s)_v}(t) = \begin{cases} q_0 \; if \; t \leq 0 \\ 1st\; cond \begin{cases} \dfrac{n^{t-\zeta_i}}{\left|n^{t+\frac{t_s}{t}-1}-n^{a}\right|}+\dfrac{n_v^{t-\zeta_{v_i}}}{\left|n_v^{t+\frac{t_v}{t}-1}-n_v^{a_v}\right|}+q_0 \; if \; 0<t\leq t_v \; and \; q_0\leq S_{v_i}\leq S_f \\ \dfrac{n^{t-\zeta_i}}{\left|n^{t+\frac{t_s}{t}-1}-n^{a}\right|}+\dfrac{n_v^{t_v-\zeta_{v_i}}}{\left|n_v^{t_v-1}-n_v^{a_v}\right|}+q_0 \; if \; t_v<t\leq t_s \; and \; q_0\leq S_{v_i}\leq S_f \end{cases} \\ 2nd\; cond \begin{cases} -\dfrac{n^{t-\zeta_i}}{\left|n^{t+\frac{t_s}{t}-1}-n^{a}\right|}+\dfrac{n_v^{t-\zeta_{v_i}}}{\left|n_v^{t+\frac{t_v}{t}-1}-n_v^{a_v}\right|}+q_0 \; if \; 0<t\leq t_v \; and \; q_0<S_{v_i}, S_f<S_{v_i} \\ -\dfrac{n^{t-\zeta_i}}{\left|n^{t+\frac{t_s}{t}-1}-n^{a}\right|}+\dfrac{n_v^{t_v-\zeta_{v_i}}}{\left|n_v^{t_v-1}-n_v^{a_v}\right|}+q_0 \; if \; t_v<t\leq t_s \; and \; q_0<S_{v_i}, S_f<S_{v_i} \end{cases} \end{cases}. \tag{7}$$

And for when $S_{v_i} < q_0$, the next conditions are followed:

$$S_{i(\zeta a t_s)_v}(t) = \begin{cases} 3rd\; cond \begin{cases} \dfrac{n^{t-\zeta_i}}{\left|n^{t+\frac{t_s}{t}-1}-n^{a}\right|}-\dfrac{n_v^{t-\zeta_{v_i}}}{\left|n_v^{t+\frac{t_v}{t}-1}-n_v^{a_v}\right|}+q_0 \; if \; 0<t\leq t_v \; and \; S_{v_i}<q_0, S_{v_i}<S_f \\ \dfrac{n^{t-\zeta_i}}{\left|n^{t+\frac{t_s}{t}-1}-n^{a}\right|}-\dfrac{n_v^{t_v-\zeta_{v_i}}}{\left|n_v^{t_v-1}-n_v^{a_v}\right|}+q_0 \; if \; t_v<t\leq t_s \; and \; S_{v_i}<q_0, S_{v_i}<S_f \end{cases} \\ 4th\; cond \begin{cases} -\dfrac{n^{t-\zeta_i}}{\left|n^{t+\frac{t_s}{t}-1}-n^{a}\right|}-\dfrac{n_v^{t-\zeta_{v_i}}}{\left|n_v^{t+\frac{t_v}{t}-1}-n_v^{a_v}\right|}+q_0 \; if \; 0<t\leq t_v \; and \; S_{v_i}<q_0, S_f<S_{v_i} \\ -\dfrac{n^{t-\zeta_i}}{\left|n^{t+\frac{t_s}{t}-1}-n^{a}\right|}-\dfrac{n_v^{t_v-\zeta_{v_i}}}{\left|n_v^{t_v-1}-n_v^{a_v}\right|}+q_0 \; if \; t_v<t\leq t_s \; and \; S_{v_i}<q_0, S_f<S_{v_i} \end{cases} \end{cases}. \tag{8}$$

ζ_i and ζ_{v_i} converge if the feedback iterative sequence $\left(S_{v_i}\right)$ is defined as follows:

$$S_{v_i} = \begin{cases} \theta_v - \dfrac{n^{t_v-\zeta_{i-1}}}{\left|n^{t_v+\frac{t_s}{t_v}-1}-n^{a}\right|} \; for\; the\; 1st\; and\; 3rd\; cond \\ \theta_v + \dfrac{n^{t_v-\zeta_{i-1}}}{\left|n^{t_v+\frac{t_s}{t_v}-1}-n^{a}\right|} \; for\; the\; 2nd\; and\; 4th\; cond \end{cases}, \tag{9}$$

For every $i = 1,2,3,4, \ldots$ and when $i = 0$, then S_{v_0} is any real number that fulfills any of the conditions presented before in (7) and (8).

S_{v_i} is called the iterative feedback sequence and is used to calculate ζ_{v_i} and ζ_i as follows:

$$\zeta_{v_i} = \begin{cases} t_v - log_{n_v}\left[\left(\left|S_{v_i}-q_0\right|\right)\left(\left|n_v^{t_v}-n_v^{a}\right|\right)\right] \; if \; q_0 < S_{v_i} \\ t_v - log_{n_v}\left[\left(\left|q_0-S_{v_i}\right|\right)\left(\left|n_v^{t_v}-n_v^{a}\right|\right)\right] \; if \; S_{v_i} < q_0 \\ \infty \; if \; S_{v_i} = q_0 \end{cases}, \tag{10}$$

$$\zeta_i = \begin{cases} t_s - log_n\left[\left(\left|S_f - S_{v_i}\right|\right)\left(\left|n^{t_s} - n^a\right|\right)\right] \; if \; S_{v_i} < S_f \\ t_s - log_n\left[\left(\left|S_{v_i} - S_f\right|\right)\left(\left|n^{t_s} - n^a\right|\right)\right] \; if \; S_f < S_{v_i} \\ \infty \; if \; S_{v_i} = S_f \end{cases}.$$

Now, with everything mentioned before, the algorithm to obtain a trajectory with an initial point, a via point at t_v, and a final point at t_s is introduced:

Algorithm. Let $n, n_v > 1$, $t_s > t_v > 0$, $a < t_s$, $a_v < t_v$, defining S_{v_i} as in (9) for every $i = 1,2,3,4,5,\ldots$ and taking any S_{v_0} that achieves any of the conditions presented before in (7) and (8), there exist ζ_{v_i} and ζ_i for when i tends to infinity, such that $S_{i(\zeta a t_s)_v}(0) = q_0$, $S_{i(\zeta a t_s)_v}(t_v) = \theta_v$ and $S_{i(\zeta a t_s)_v}(t_s) = S_f$.

Proof: The constraints $S_{i(\zeta a t_s)_v}(0) = q_0$ and $S_{i(\zeta a t_s)_v}(t_s) = S_f$ for every $i = 0,1,2,3,\ldots$ are easily fulfill by definition:

Now, let any S_{v_0} such as $S_f > S_{v_0}$ and $q_0 < S_{v_0}$, then $\zeta_0 = t_s - log_n\left[\left(\left|S_f - S_{v_0}\right|\right)\left(\left|n^{t_s} - n^a\right|\right)\right]$ and $\zeta_{v_0} = t_v - log_{n_v}\left[\left(\left|S_{v_0} - q_0\right|\right)\left(\left|n_v^{t_v} - n_v^{a}\right|\right)\right]$, the trajectory functions (11) at the iteration $i = 0$ is the following:

$$S_{(\zeta a t_s)_{v_0}}(t) = \frac{n^{t-\zeta_0}}{\left|n^{t+\frac{t_s}{t}-1} - n^a\right|} + \frac{n_v^{t-\zeta_{v_0}}}{\left|n_v^{t+\frac{t_v}{t}-1} - n_v^{a_v}\right|} + q_0. \tag{11}$$

Then, making:

$$\frac{n_v^{t_v-\zeta_{v_2}}}{\left|n_v^{t_v} - n_v^{a_v}\right|} + q_0 = S_{v_2} = \theta_v - \frac{n^{t_v-\zeta_0}}{\left|n^{t_v+\frac{t_s}{t_v}-1} - n^a\right|}. \tag{12}$$

Using the new feedback sequence in the second iteration S_{v_2} to calculate $\zeta_2 = t_s - log_n\left[\left(\left|S_f - S_{v_2}\right|\right)\left(\left|n^{t_s} - n^a\right|\right)\right]$ and $\zeta_{v_2} = t_v - log_{n_v}\left[\left(\left|S_{v_2} - q_0\right|\right)\left(\left|n_v^{t_v} - n_v^{a}\right|\right)\right]$.

$$S_{(\zeta a t_s)_{v_2}}(t) = \frac{n^{t-\zeta_2}}{\left|n^{t+\frac{t_s}{t}-1} - n^a\right|} + \frac{n_v^{t-\zeta_{v_2}}}{\left|n_v^{t+\frac{t_v}{t}-1} - n_v^{a_v}\right|} + q_0. \tag{13}$$

Now, taking any value of S_{v_1} such that $S_{v_1} < S_f$ and $q_0 > S_{v_1}$, then $\zeta_1 = t_s - log_n\left[\left(\left|S_f - S_{v_1}\right|\right)\left(\left|n^{t_s} - n^a\right|\right)\right]$ and $\zeta_{v_1} = t_v - log_{n_v}\left[\left(\left|q_0 - S_{v_1}\right|\right)\left(\left|n_v^{t_v} - n_v^{a}\right|\right)\right]$, it gets the following trajectory function (14) at $i = 1$:

$$S_{(\zeta a t_s)_{v_1}}(t) = \frac{n^{t-\zeta_1}}{\left|n^{t+\frac{t_s}{t}-1} - n^a\right|} - \frac{n_v^{t-\zeta_{v_1}}}{\left|n_v^{t+\frac{t_v}{t}-1} - n_v^{a_v}\right|} + q_0. \tag{14}$$

And making:

$$-\frac{n_v^{t-\zeta_{v_3}}}{\left|n_v^{t_v} - n_v^{a_v}\right|} + q_0 = S_{v_3} = \theta_v - \frac{n^{t_v-\zeta_1}}{\left|n^{t_v+\frac{t_s}{t_v}-1} - n^a\right|}. \tag{15}$$

Using S_{v_3} in (15) to calculate ζ_3 and ζ_{v_3} using formula (10) , then, the following trajectory function (16) at $i = 3$ is:

$$S_{(\zeta a t_s)_{v_3}}(t) = \frac{n^{t-\zeta_3}}{\left|n^{t+\frac{t_s}{t}-1} - n^a\right|} - \frac{n_v^{t-\zeta_{v_3}}}{\left|n_v^{t+\frac{t_v}{t}-1} - n_v^{a_v}\right|} + q_0. \tag{16}$$

Then, for the 4th iteration ($i = 4$), the following sequence (17) is obtained:

$$\frac{n_v^{t_v-\zeta_{v_4}}}{\left|n_v^{t_v}-n_v^{a_v}\right|}+q_0=S_{v_4}=\theta_v-\frac{n^{t_v-\zeta_2}}{\left|n^{t_v+\frac{t_s}{t_v}-1}-n^a\right|}. \tag{17}$$

With $\zeta_4=t_s-log_n\left[\left(\left|S_f-S_{v_4}\right|\right)\left(\left|n^{t_s}-n^a\right|\right)\right]$ and $\zeta_{v_4}=t_v-log_{n_v}\left[\left(\left|S_{v_4}-q_0\right|\right)\left(\left|n_v^{t_v}-n_v^a\right|\right)\right]$ by using formula (10).

And for the 5th iteration, then:

$$-\frac{n_v^{t_v-\zeta_{v_5}}}{\left|n_v^{t_v}-n_v^{a_v}\right|}+q_0=S_{v_5}=\theta_v-\frac{n^{t_v-\zeta_3}}{\left|n^{t_v+\frac{t_s}{t_v}-1}-n^a\right|}. \tag{18}$$

With $\zeta_5=t_s-log_n\left[\left(\left|S_f-S_{v_5}\right|\right)\left(\left|n^{t_s}-n^a\right|\right)\right]$ and $\zeta_{v_5}=t_v-log_{n_v}\left[\left(\left|q_0-S_{v_5}\right|\right)\left(\left|n_v^{t_v}-n_v^a\right|\right)\right]$ by using formula (10).

Repeating this process k times, it has:

$$\frac{n_v^{t_v-\zeta_{v_k}}}{\left|n_v^{t_v}-n_v^{a_v}\right|}+q_0=S_{v_k}=\theta_v-\frac{n^{t_v-\zeta_{k-2}}}{\left|n^{t_v+\frac{t_s}{t_v}-1}-n^a\right|}. \tag{19}$$

Using formula (10), $\zeta_k=t_s-log_n\left[\left(\left|S_f-S_{v_k}\right|\right)\left(\left|n^{t_s}-n^a\right|\right)\right]$ and $\zeta_{v_k}=t_v-log_{n_v}\left[\left(\left|S_{v_k}-q_0\right|\right)\left(\left|n_v^{t_v}-n_v^a\right|\right)\right]$ and (20) is obtained representing the trajectory in in the k-iteration:

$$S_{(\zeta at_s)_{v_k}}(t)=\frac{n^{t-\zeta_k}}{\left|n^{t+\frac{t_s}{t}-1}-n^a\right|}+\frac{n_v^{t-\zeta_{v_k}}}{\left|n_v^{t+\frac{t_v}{t}-1}-n_v^{a_v}\right|}+q_0, \tag{20}$$

Such that:

$$\begin{aligned}S_{(\zeta at_s)_{v_k}}(t_v)&=\frac{n^{t_v-\zeta_k}}{\left|n^{t_v+\frac{t_s}{t_v}-1}-n^a\right|}+\frac{n_v^{t_v-\zeta_{v_k}}}{\left|n_v^{t_v}-n_v^{a_v}\right|}+q_0=\frac{n^{t_v-\zeta_k}}{\left|n^{t_v+\frac{t_s}{t_v}-1}-n^a\right|}+\left(S_{v_k}\right)\\&=\frac{n^{t_v-\zeta_k}}{\left|n^{t_v+\frac{t_s}{t_v}-1}-n^a\right|}+\left(\theta_v-\frac{n^{t_v-\zeta_{k-2}}}{\left|n^{t_v+\frac{t_s}{t_v}-1}-n^a\right|}\right),\end{aligned} \tag{21}$$

And for the $k+1$ time, it has:

$$-\frac{n_v^{t_v-\zeta_{v_{k+1}}}}{\left|n_v^{t_v}-n_v^{a_v}\right|}+q_0=S_{v_{k+1}}=\theta_v-\frac{n^{t_v-\zeta_{k-1}}}{\left|n^{t_v+\frac{t_s}{t_v}-1}-n^a\right|}. \tag{22}$$

Then, using formula (10), $\zeta_{k+1}=t_s-log_n\left[\left(\left|S_f-S_{v_{k+1}}\right|\right)\left(\left|n^{t_s}-n^a\right|\right)\right]$ and $\zeta_{v_{k+1}}=t_v-log_{n_v}\left[\left(\left|q_0-S_{v_{k+1}}\right|\right)\left(\left|n_v^{t_v}-n_v^a\right|\right)\right]$ and the trajectory function (23) is obtained:

$$S_{(\zeta at_s)_{v_{k+1}}}(t)=\frac{n^{t-\zeta_{k+1}}}{\left|n^{t+\frac{t_s}{t}-1}-n^a\right|}-\frac{n_v^{t-\zeta_{v_{k+1}}}}{\left|n_v^{t+\frac{t_v}{t}-1}-n_v^{a_v}\right|}+q_0, \tag{23}$$

Such that:

$$\begin{aligned}S_{(\zeta at_s)_{v_{k+1}}}(t_v)&=\frac{n^{t_v-\zeta_{k+1}}}{\left|n^{t_v+\frac{t_s}{t_v}-1}-n^a\right|}-\frac{n_v^{t_v-\zeta_{v_{k+1}}}}{\left|n_v^{t_v}-n_v^{a_v}\right|}+q_0=\frac{n^{t_v-\zeta_{k+1}}}{\left|n^{t_v+\frac{t_s}{t_v}-1}-n^a\right|}+\left(S_{v_{k+1}}\right)\\&=\frac{n^{t_v-\zeta_{k+1}}}{\left|n^{t_v+\frac{t_s}{t_v}-1}-n^a\right|}+\left(\theta_v-\frac{n^{t_v-\zeta_{k-1}}}{\left|n^{t_v+\frac{t_s}{t_v}-1}-n^a\right|}\right).\end{aligned} \tag{24}$$

Then, because of $\frac{n^{t_v-\zeta_{k-2}}}{\left|n^{t_v+\frac{t_s}{t_v}-1}-n^a\right|}$ and $\frac{n^{t_v-\zeta_{k-1}}}{\left|n^{t_v+\frac{t_s}{t_v}-1}-n^a\right|}$ are always finite for every ζ_{k-2} and ζ_{k-1}, and the way ζ_{k-2}, ζ_{k-1} and the subsequent S_{v_k} and $S_{v_{k+1}}$ have been defined, this mean that when k tends to infinity ($k \to \infty$) then $\lim_{k\to\infty} S_{v_k} = \theta_v - \frac{n^{t_v-\zeta_\infty}}{\left|n^{t_v+\frac{t_s}{t_v}-1}-n^a\right|} = \lim_{k\to\infty} S_{v_{k+1}}$, this limit is finite too, making:

$$\begin{aligned} S_{(\zeta a t_s)_{v\infty}}(t_v) &= \frac{n^{t_v-\zeta_\infty}}{\left|n^{t_v+\frac{t_s}{t_v}-1} - n^a\right|} + \frac{n_v^{t_v-\zeta_{v\infty}}}{\left|n_v^{t_v} - n_v^{a_v}\right|} + q_0 = \frac{n^{t_v-\zeta_\infty}}{\left|n^{t_v+\frac{t_s}{t_v}-1} - n^a\right|} + \left(S_{v_\infty}\right) \\ &= \frac{n^{t_v-\zeta_\infty}}{\left|n^{t_v+\frac{t_s}{t_v}-1} - n^a\right|} + \left(\theta_v - \frac{n^{t_v-\zeta_\infty}}{\left|n^{t_v+\frac{t_s}{t_v}-1} - n^a\right|}\right) = \theta_v, \end{aligned} \tag{25}$$

And,

$$\begin{aligned} S_{(\zeta a t_s)_{v\infty}}(t_v) &= \frac{n^{t_v-\zeta_\infty}}{\left|n^{t_v+\frac{t_s}{t_v}-1} - n^a\right|} - \frac{n_v^{t_v-\zeta_{v\infty}}}{\left|n_v^{t_v} - n_v^{a_v}\right|} + q_0 \\ &= \frac{n^{t_v-\zeta_\infty}}{\left|n^{t_v+\frac{t_s}{t_v}-1} - n^a\right|} + \left(-\frac{n_v^{t_v-\zeta_{v\infty}}}{\left|n_v^{t_v} - n_v^{a_v}\right|} + q_0\right) = \frac{n^{t_v-\zeta_\infty}}{\left|n^{t_v+\frac{t_s}{t_v}-1} - n^a\right|} + \left(S_{v_\infty}\right) \\ &= \frac{n^{t_v-\zeta_\infty}}{\left|n^{t_v+\frac{t_s}{t_v}-1} - n^a\right|} + \left(\theta_v - \frac{n^{t_v-\zeta_\infty}}{\left|n^{t_v+\frac{t_s}{t_v}-1} - n^a\right|}\right) = \theta_v. \end{aligned} \tag{26}$$

Therefore, $S_{(\zeta a t_s)_{v_i}}(t_v) = \theta_v$ for when $i \to \infty$. An analog proof can be constructed for the other conditions.

■

3.1. Examples

Now, considering a trajectory with $q_0 = -12$, $\theta_v = 5$ and $S_f = 30$ with the next parameters $n_v = 2, a_v = 0.05, t_v = 1.5$ with a final $\zeta_{v_i} = 5.4327\ldots$ and $n = 2.5$, $a = -1.4$, $t_s = 3$ and a final $\zeta_i = -4.0586\ldots$ at the iteration number 40 and $S_{v_i} = -11.9635$. The following trajectories are presented in Figure 2.

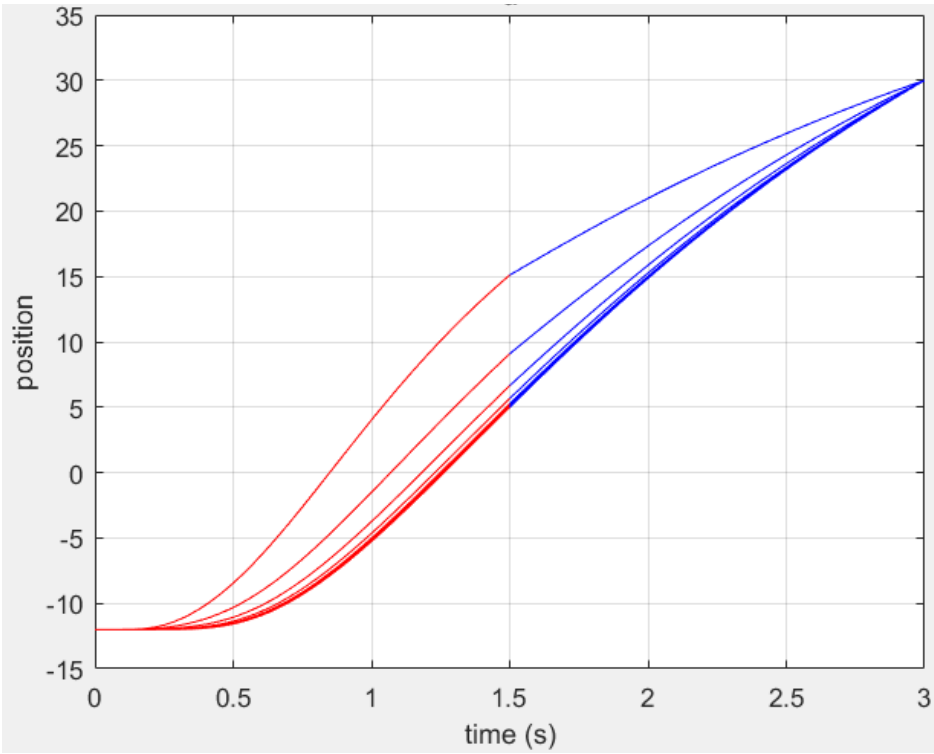

Figure 2. Homotopy trajectories are approaching to the via point $\theta_v = 5$ at $t_v = 1.5$ and initial position $q_0 = -12$ and $S_f = 30$ using the first and second condition.

As noted in Figure 1, the trajectory avoids oscillation with a soft start and reaches the final position with a sharp end (non-zero velocity). Nevertheless, the trajectory can be softer at the end position, changing the parameters a_v and a and switching between the abovementioned conditions. For example, considering the parameters presented before but changing $a_v = 1.2481$ and a $S_{v_i} = -27.2749$. The obtained results are shown in Figure 3.

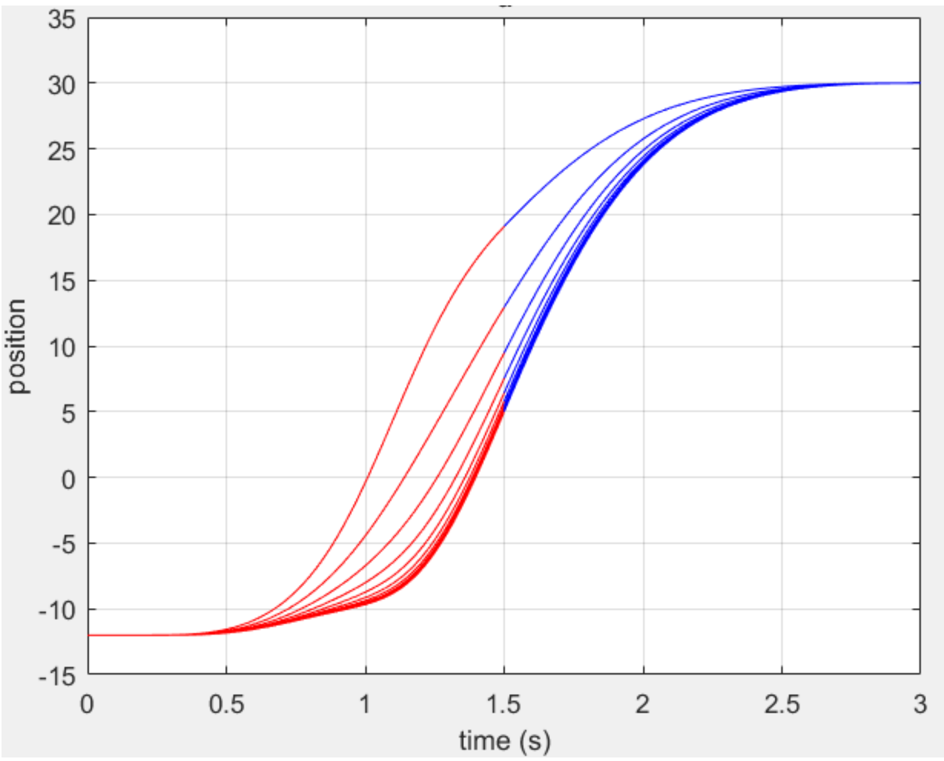

Figure 3. Homotopy trajectories are approaching the via point $\theta_v = 5$ at $t_v = 1.5$ and initial position $q_0 = -12$ and $S_f = 30$ using the first, second, and third condition and changing its shape.

As can be seen in Figure 3, the last trajectory passes through the three desired positions presented in Figure 2 but with a different shape. This is because different parameters have been chosen, and a velocity too close to zero at t_s has been obtained by repeating the process that this algorithm defines, making the trajectory has a S-shape as a result.

Now, taking a much more difficult trajectory to perform, the algorithm can be run for a longer time, and sometimes this is much more difficult to guess the complexity of the trajectory. In other words, it is much more challenging to know exactly where an oscillation is and is not occurring. For example, Figure 4 presents a trajectory with this behavior, considering the following values: $q_0 = 10$, $\theta_v = 25$, and $S_f = 62$. Taking $n_v = 1.7, a_v = 0.05, t_v = 2.6$ with a final $\zeta_{v_i} = -16.6294\ldots$ and $n = 1.8$, $a = 1.2$, $t_s = 3$ and $\zeta_i = -14.8052\ldots$ in the iteration 7067th.

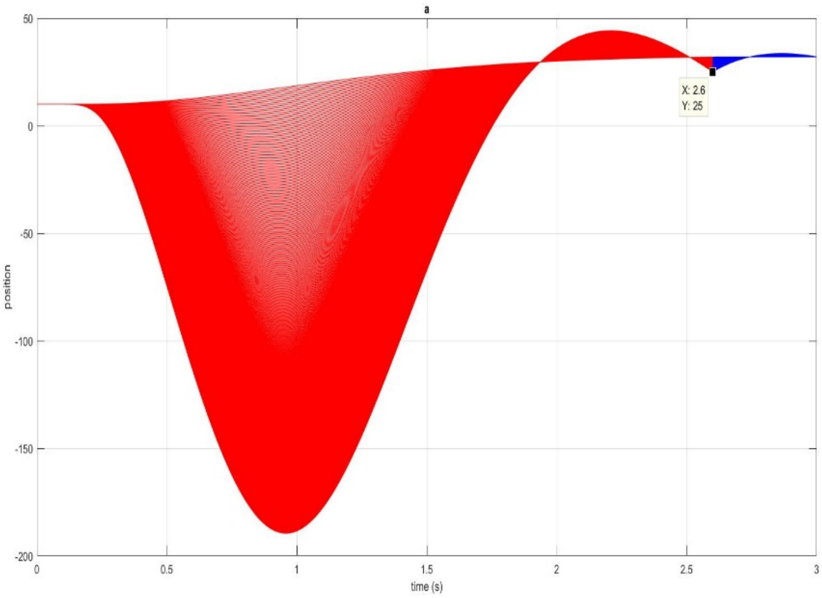

Figure 4. Example of complex trajectories approaching the via point $\theta_v = 25$ at $t_v = 2.6$ and initial position $q_0 = 10$ and $S_f = 62$ at $t_s = 3$.

This trajectory can be modified by changing some preliminary parameters such as n_v, n, a_v, and a. However, these are difficult to guess. Currently, there is not an existing analytic formula or definition to get these parameters for a particular trajectory shape, so these parameters are changed manually to get complex trajectories. For example, Figure 5 shows trajectories that were obtained while changing some of the parameters mentioned before.

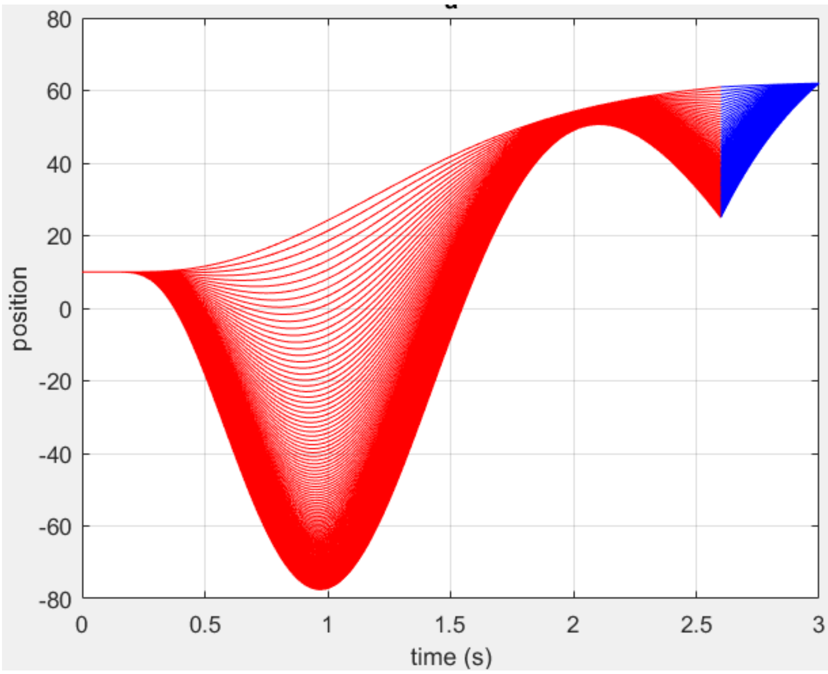

Figure 5. Example of complex trajectories approaching the via point $\theta_v = 25$ at $t_v = 2.6$ and initial position $q_0 = 10$ and $S_f = 62$ at $t_s = 3$, reducing the spikes of the oscillations changing the parameters n_v, a_v, and n, a.

Finally, obtaining a trajectory with the same constraints with no oscillations is possible. For example, Figure 6 shows the trajectories obtained with the following values $n_v = 125$, $a_v = -100$, $n = 70$, and $a = 2.2$ with a final $\zeta_{v_i} = -0.7130 \ldots$ and $\zeta = -1.0328$ and a $S_{v_i} = -21.2603 \ldots$ at the iteration number 58th.

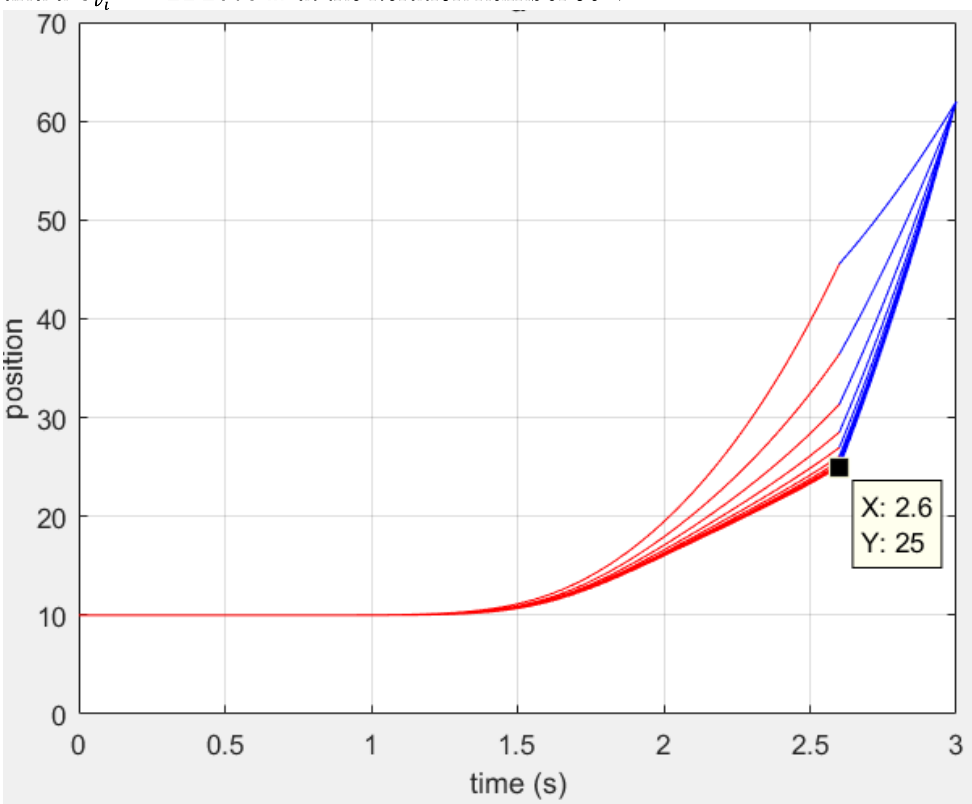

Figure 6. Example of complicated trajectories approaching the via point $\theta_v = 25$ at $t_v = 2.6$ and initial position $q_0 = 10$ and $S_f = 62$ at $t_s = 3$, reducing the spikes of the oscillations changing the parameters n_v, a_v, and n, a.

In Figure 6, it can be observed that the trajectory avoids any oscillation. This is possible because the parameters mentioned before have been changed. The trajectory made between the initial point (q_0) and the via-point (θ_v) has a different velocity, acceleration, and Jerk trajectory than the trajectory made between the via-point (θ_v) and the final point S_f. In other words, it has C^0 continuity for the cases presented before. Hence, avoiding any oscillation for any via point time t_v and final time t_s is possible. This is another advantage gained by taking IOSTV $(\zeta)_v$-HF, therefore versatile and complex trajectories can be obtained by applying this method, and this is not always possible with other current methods.

Also, note that in Figures 4, 5, and 6, the trajectory ends with a sharp end position. This is because the velocity has a short period (which is $2.6 \leq t \leq 3$) to be well distributed through that period of time. Nevertheless, the trajectory always presents finite velocity, acceleration, and jerks, and the trajectory at the end position can be softer over a more significant period.

4. Results: Velocity, Acceleration and Jerk function

The velocity, acceleration, and jerk functions have been presented in [37]. This work needs to retake the topic of these functions because it is crucial for a long-life robotic system to exist. As shown in the previous examples, the trajectories presented get no continuous velocity, acceleration, and jerk at the via point time (t_v). Nevertheless, every value is bounded, and the gap between the velocity, acceleration, and jerk in t_v can be reduced as much as desired, making a safe motion in the joint space. Moreover, the IOSTV $(\zeta)_v$-HF is not the first method that presents no-continuous velocity, acceleration, or jerk. Some methods mentioned before in the introduction and many others often used currently present no-continuous velocity, acceleration, or jerk. For example, [38] used trapezoidal velocity profiles to generate trajectories and presents a no-continuous jerk profile, which is bounded and ready for implementation.

Also, [39, 49] present not-zero velocity at the final point, but this method is a perfect tool for obstacle avoidance, as it wanted to show using IOSTV $(\zeta)_v$-HF.

In [38], the via-point trajectory taken from [9] has been used to compare it with the IOSTV $(\zeta)_v$-HF method by presenting a trajectory shape with continuous velocity and another with continuous acceleration. In [9], the trajectory compared was not optimal, and a significant gap in the velocity was presented. In this work, the via-point trajectory taken from [9] is retaken to get a better trajectory than the one that was presented in [37] and compared with the result from [36]. The values are $q_0 = 30$, $\theta_v = 180$ at $t_v = 1.5$ seconds, and $S_f = 120$ at $t_s = 3$ seconds, and the sixth order polynomial from [9] is:

$$\theta(t) = -9.22t^6 + 85.19t^5 - 265.56t^4 + 282.22t^3 + 30. \tag{27}$$

Plotting this trajectory in Matlab, it is shown that the trajectory gets 185.4 degrees as a maximum value, and $t_v = 1.5$ the trajectory gets to $\theta_v = 180$ degrees.

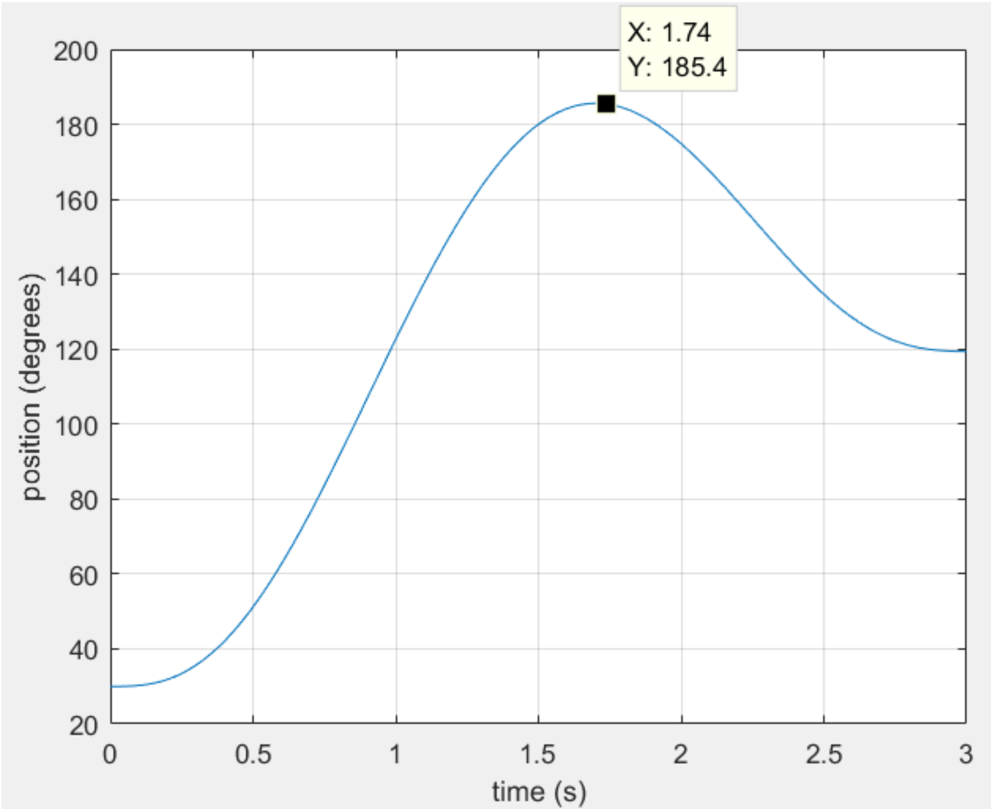

Figure 7. Trajectory obtained from the sixth-order polynomial.

As shown in Figure 7, the trajectory presents a maximum value at $t = 1.74$ sec. The method proposed in this work tried to reduce this oscillation, so this one was would not be greater than 181 degrees. The algorithm IOSTV $(\zeta)_v$-HF was run several times until it got some solutions that fulfilled the requirements. First, it testes the trajectory result by finding values on the parameters to get a continuous velocity, for example, using the parameters $n_v = 1.91$, $a_v = 0.8734$, $\zeta_{v_i} = -6.0657 \ldots$, $n = 81.7467$, $a = 2.4$, and $\zeta_i = -0.9204$ in the iteration number 12, the following profiles are obtained and are shown in Figure 8.

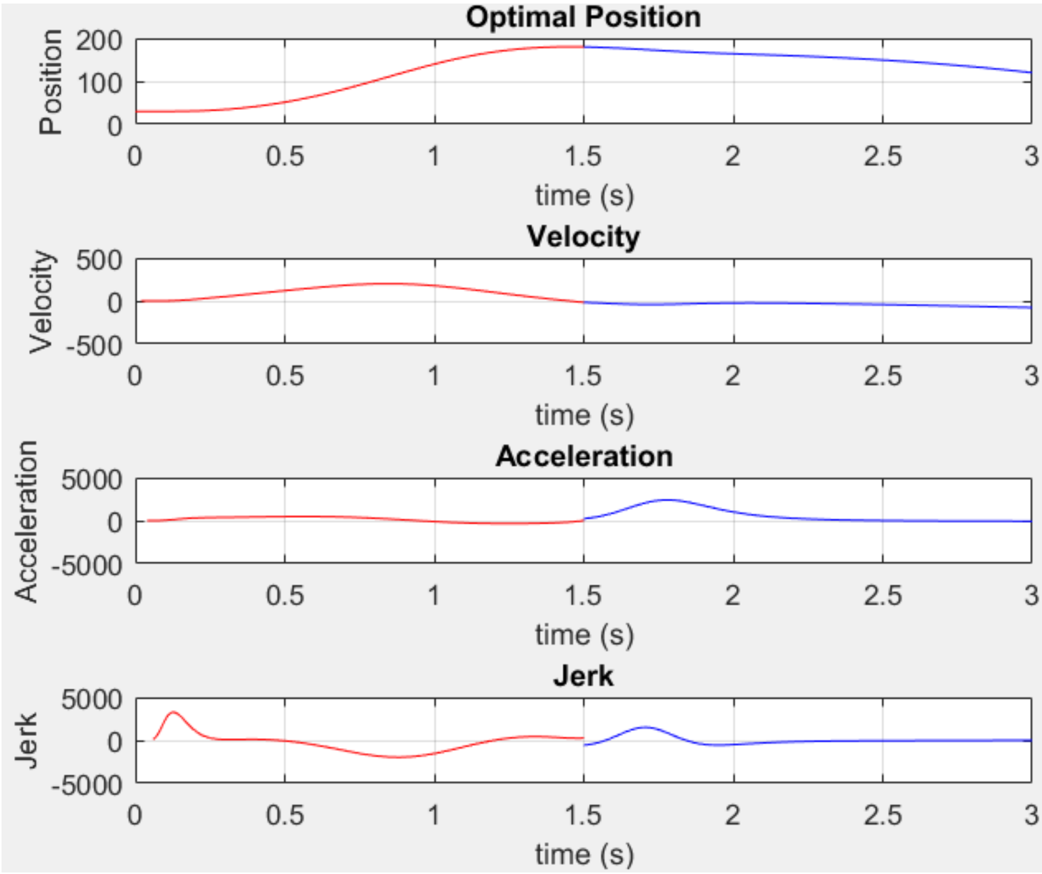

Figure 8. Trajectory solution with continuous velocity (it has C^1 continuity) starting at $q_0 = 30$, via point $\theta_v = 180$ and $S_f = 120$.

Regarding the Acceleration and Jerk, it can be observed that they were discontinuous in t_v. However, these do not present discontinuities at the start and end, like the method presented in [9]. Therefore, the discontinuities have been reduced to one by using IOSTV $(\zeta)_v$-HF, and these are also kept finite. Also, the differences between the acceleration at t approaches to the left-approximation and the right-approximation to t_v are insignificant, about 219.824 degrees/s^2 of difference and getting 180.4 degrees as a leading position in the trajectory.

Then, Figure 9 shows a trajectory with a continuous acceleration but discontinuous velocity and jerk using the following values $n_v = 5, a_v = 0.5865$, and $\zeta_{v_i} = -3.0399$, and $n = 5$, $a = 1.95$ and $\zeta_i = -2.6193$ in the iteration number 28.

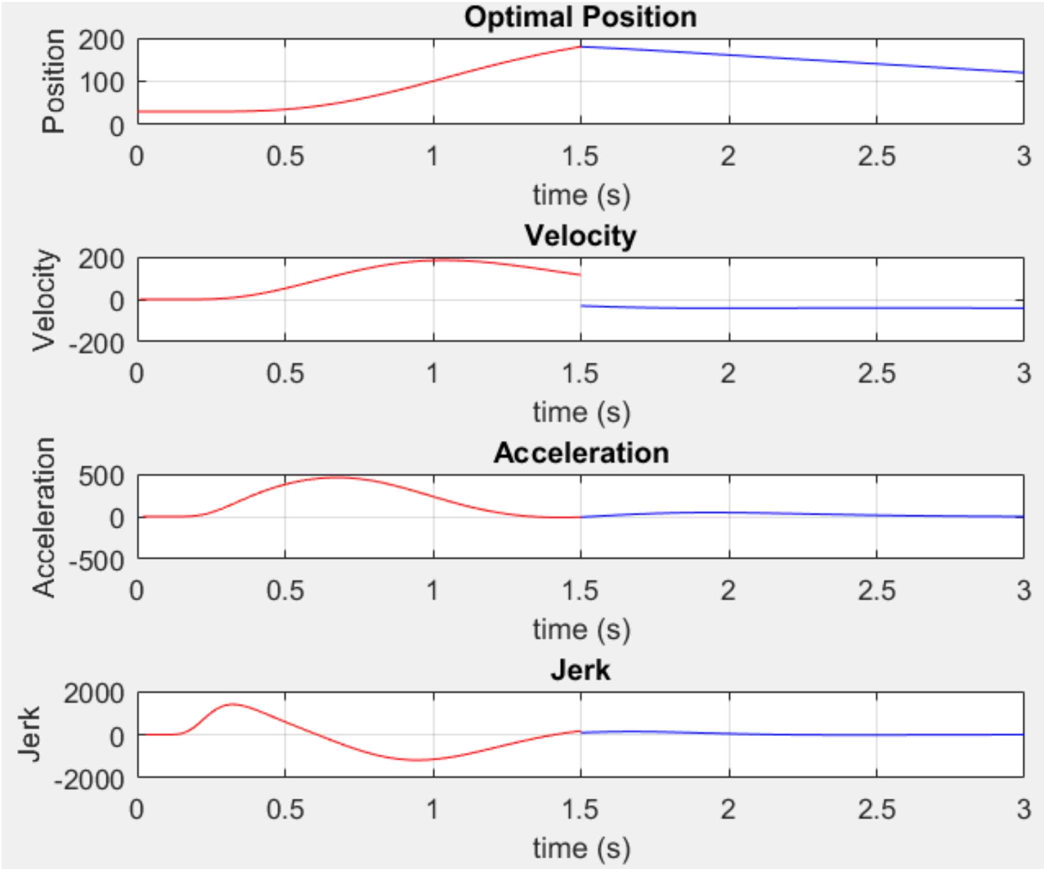

Figure 9. Trajectory solution with continuous acceleration starting at $q_0 = 30$, via point $\theta_v = 180$ and $S_f = 120$.

With the above parameters, the trajectory gets even a smaller jerk than the one that was found while using the 6th-order polynomial function from [9]. The acceleration function using OISTV $(\zeta)_v$-HF starts at 0 degrees/s^3 and ends too close at 0 degrees/s^3, which means that the trajectory shown presents a finite Jerk. The maximum value in the jerk function using OISTV $(\zeta)_v$-HF was 1394 degrees/s^3, and the lowest value was -1177 degrees/s^3 while using the 6th-order polynomial function, the maximum value jerk was 1693 degrees/s^3, and the lowest value was -1294 degrees/ s^3. Also, the trajectory using IOSTV $(\zeta)_v$-HF or the 6th-order polynomial function from [9] presents a finite jerk; this one is not continuous, though, but this characteristic, according to [9], obeys the rule of thumb for mechanical design/motion.

5. Discussion

This paper presents the idea of getting a trajectory to obtain the best performance or a motion in the joint space that fulfills some preliminary conditions that a user can state. This idea is reached by using the method presented and named after this work as OISTV $(\zeta)_v$-HF. Some of these many essential conditions have been tested that make a safe trajectory while controlling a robotic arm. First, the method OISTV $(\zeta)_v$-HF avoids and reduces any undesired oscillation through the trajectory. The trajectory keeps finite velocity, acceleration, and jerk that obey the rule of thumb for mechanical design/motion; the algorithm OISTV $(\zeta)_v$-HF works by iteration. This method creates a trajectory that is as accurate as wanted, generating many trajectories. In contrast, the number of iterations increases until a trajectory passes through the initial position q_0 at $t = 0$, then θ_v at the via point time t_v, and finally gets to the final position S_f at t_s at the end time.

Nevertheless, we have found some disadvantages while using this method. First, a desirable trajectory can be obtained, but some drawbacks must be made; for example, a trajectory can get a no-continuous velocity, more significant spikes in acceleration, or jerk.

At this time, there is not an existing method to get the best parameters that can allow us to generate a trajectory that fulfills all the desirable conditions. Therefore, we had to run this algorithm many times to obtain these parameters and get the desired trajectory, which means much time spent on tests. Moreover, the algorithm can run for an extended period, and the time the convergence occurs can vary depending on the parameters. For example, in these results, the longest-running time was about 17.5 seconds, and the shortest-running time was about 0.052 seconds (the running time was obtained using a function of Matlab called as tic toc function). So, a method to generate trajectories with a faster running time and find the best parameters to generate a trajectory that fulfills all the desired conditions must be developed in future works.

6. Conclusion

According to the results obtained in this work, it can be hypothesized that only one or several sets of parameters can work to obtain the desired trajectory. In addition, one or several sets of parameters can make a trajectory with continuous velocity, acceleration, and jerk.

The method converges to a via point θ_v at any via point time t_v. The method proposed has been tested, and according to the results and all characteristics introduced in this work; the IOSTV $(\zeta)_v$-HF method presents some advantages, such as avoiding unwanted oscillations; in addition, several options are generated to choose the best trajectory or, in its case, the one that meets the desired conditions. It can reduce the complexity of trajectories by setting new parameters, and generating infinity options of generating a trajectory that fulfills basic constraints. Furthermore, the method generates trajectories with a finite jerk and continuous acceleration that avoids infinite jerks. All the derivatives while using the IOSTV $(\zeta)_v$-HF start at 0, which is a uniqueness of this method and is advantageous for reliable, smooth, and long-life robotic systems.

Although this method has to be improved, at this time, it is a confident tool for generating safe and reliable trajectories; meanwhile, a new algorithm has to be designed to get a trajectory that converges to the via point by iterations in a faster way. Also, in future works, finding a solution to get a set of suitable parameters that makes a continuous velocity, acceleration, and jerk without presenting undesirables oscillations could make a complete method to generate safe trajectories in the joint space.

Author Contributions: Conceptualization, Y.Q. and O.Z. Ideas; formulation or evaluation of general research objectives and goals; methodology, Y.Q., O.Z, C.L. and R.A. were developed and designed the methodology; as well as creating models; software, Y.Q. and O.Z. Programming, software development; designing computer programs; implementation of the computer code and supporting algorithms; testing of existing code components; validation, Y.Q. and O.Z. Verification, whether as a part of the activity or separate, of the overall replication/reproducibility of results/experiments and other research outputs; formal analysis, Y.Q. and O.Z. Application of statistical, mathematical, computational, or other formal techniques to analyze or synthesize study data; investigation, Y.Q. and O.Z. Conducting a research and investigation process, specifically performing the experiments, or data/evidence collection; resources, Y.Q., O.Z., C.L. and R.A. Provision of study materials, reagents, materials, patients, laboratory samples, animals, instrumentation, computing resources, or other analysis tools; data curation, Y.Q. and O.Z. Management activities to annotate (produce metadata), scrub data and maintain research data (including software code, where it is necessary for interpreting the data itself) for initial use and later reuse; writing—original draft preparation, Y.Q. and O.Z. Preparation, creation and/or presentation of the published work, specifically writing the initial draft (including substantive translation); writing—review and editing, Y.Q. and O.Z. Preparation, creation and/or presentation of the published work by those from the original research group, specifically critical review, commentary or revision—including pre-or post-publication stages; visualization, Y.Q., O.Z. and R.A. Preparation, creation and/or presentation of the published work, specifically visualization/data presentation; supervision, Y.Q. and O.Z. Oversight and leadership responsibility for the research activity planning and execution, including mentorship external to the core team; project administration, Y.Q., O.Z, C.L. and R.A. Management and

coordination responsibility for the research activity planning and execution; funding acquisition, Y.Q, O.Z. All authors have read and agreed to the published version of the manuscript.

Funding: "This research was funded by Universidad Autonoma de Sinaloa through PROFAPI 2022, the project's key: PRO_A8_052".

Data Availability Statement: "Not applicable" here.

Acknowledgments: "The authors want to thank the Autonomous University of Sinaloa for financing this project".

Conflicts of Interest: "The authors declare no conflict of interest".

References

1. Martín, F.A.; Castillo, J.C.; Malfáz, M.; Castro-González, Á. Applications and Trends in Social Robotics. Electronics 2022, 11, 212.
2. Smids, J.; Nyholm, S.; Berkers, H. Robots in the Workplace: a Threat to—or Opportunity for—Meaningful Work?. Philos. Technol. 2020, 33, 503–522.
3. Grau, A.; Indri, M.; Bello, L.L.; Sauter, T. Industrial robotics in factory automation: From the early stage to the Internet of Things. In Proceedings of the 43rd Annual Conference of the IEEE Industrial Electronics Society, Beijing, China, 29 October–1 November 2017, pp. 6159–6164.
4. Yenorkar, R.; Chaskar, U.M. GUI Based Pick and Place Robotic Arm for Multipurpose Industrial Applications. In Proceedings of the Second International Conference on Intelligent Computing and Control. Systems, Madurai, India, 14–15 June 2018, pp. 200–203.
5. Kyrarini, M.; Lygerakis, F.; Rajavenkatanarayanan, A.; Sevastopoulos, C.; Nambiappan, H.R.; Chaitanya, K.K.; Babu, A.R.; Mathew, J.; Makedon, F. A Survey of Robots in Healthcare. Technologies 2021, 9(1), 8.
6. Benotsmane, R.; Dudás, L.; Kovács, G. Newly Elaborated Hybrid Algorithm for Optimization of Robot Arm's Trajectory in Order to Increase Efficiency and Provide Sustainability in Production. Sustainability 2021, 13, 8193.
7. Singh, G.; Banga, V.K. Robots and its types for industrial applications. Materials Today: Proceedings 2022, 60(3), 1779-1786.
8. Liu, X.; Qiu, C.; Zeng, Q.; Li, A. Kinematics Analysis and Trajectory Planning of collaborative welding robot with multiple manipulators. Procedia CIRP 2019, 81, 1034–1039.
9. Williams, R.L. Simplified Robotics Joint-Space Trajectory Generation with a via Point Using a Single Polynomial. Journal of Robotics, 2013, 1–6.
10. Dong, M.; Yao, G.; Li, J.; Zhang, L. Research on Attitude Interpolation and Tracking Control Based on Improved Orientation Vector SLERP Method. Robotica 2020, 38(4), 719–731.
11. Sidobre, D.; Desormeaux, K. Smooth Cubic Polynomial Trajectories for Human-Robot Interactions. Journal of Intelligent and Robotics Systems 2019, 95, 851–869.
12. Hong-Jun, H.; Yungdeug, S.; Jang-Mok K. A Trapezoidal Velocity Profile Generator for Position Control Using a Feedback Strategy. Energies 2019, 12(7), 1–14.
13. Zhao, R; Shi, Z; Guan, Y.; Shao, Z.; Zhang, Q.; Wang, G. Inverse kinematic solution of 6R robot manipulators based on screw theory and the Paden–Kahan subproblem. International Journal of Advanced Robotic Systems 2018, 15(6), 1–11
14. Wang, Y.; Su, C.; Wang, H.; Zhang, Z.; Sheng, C.; Cui, W.; Liang, X.; Lu, X. A Convenient Kinematic Calibration and Inverse Solution Method for 4-DOF Robot. In Proceedings of Chinese Control and Decision Conference, Nanchang, China, 3-5 June 2019, pp. 5747–5750.
15. Csanádi, B.; Tar, J. K.; Bitó, J. F. Matrix inversion-free quasi-differential approach in solving the inverse kinematic task. In Proceedings of 17th International Symposium on Computational Intelligence and Informatics, Budapest, Hungary, 17-19 November 2016, pp. 000061–000066.
16. Liu, W.; Chen, D.; Steil, J. J. Analytical inverse kinematics solver for anthropomorphic 7-DOF redundant manipulators with human-like configuration constraints. Journal of Intelligent and Robotics Systems 2017, 86(1), 63–79.
17. Kuhlemann, I., Schweikard, A., Ernst, F., Jauer, P. Robust inverse kinematics by configuration control for redundant manipulators with seven DOF. In Proceedings of 2nd International Conference on Control, Automation and Robotics, Hong Kong, China, 28-30 April 2016, pp. 49–55,
18. Gong, M., Li, X., Zhang, L. Analytical Inverse Kinematics and Self-Motion Application for 7-DOF Redundant Manipulator. IEEE Access 2019, 7, 18662–18674.
19. Benotsmane, R.; Dudás, L.; Kovács, G. Newly Elaborated Hybrid Algorithm for Optimization of Robot Arm's Trajectory in Order to Increase Efficiency and Provide Sustainability in Production. Sustainability 2021, 13, 8193.
20. Walch, A.; Eitzinger, C.; Zambal, S.; Palfinger, W. LSPB Trajectory Planning Using Quadratic Splines. In Proceedings of the 3rd International Conference on Mechatronics and Robotics Engineering. ICMRE 2017. Association for Computing Machinery, New York, NY, USA 2017, 81—87.
21. Raji, A. A., Asaolu, O. S., Akano, T. T. Joint Space Robot Arm Trajectory Planning Using Septic Function. ABUAD Journal of Engineering Research and Development 2022, 5(1), 110-123.

22. Zaghlul, S., Al-khayyt, S. Creating Through Points in Linear Function with Parabolic Blends Path by Optimization Method. Al-Khwarizmi Engineering Journal 2018, 14(1), 77-89.
23. Barghi Jond, H., V. Nabiyev, V., Benveniste, R. Trajectory Planning Using High Order Polynomials under Acceleration Constraint. Journal of Optimization in Industrial Engineering 2016, 10(21), 1-6.
24. Sciavicco, L., Siciliano, B. Modelling and Control of Robot Manipulators 2010, Springer: London, UK.
25. Zhang, J., Meng, Q., Feng, X., Shen, H. A 6-DOF robot-time optimal trajectory planning based on an improved genetic algorithm. Robot. Biomim 2018. 5, 1-7.
26. 21 Song, Q.; Li, S.; Bai, Q.; Yang, J.; Zhang, A.; Zhang, X.; Zhe, L. Trajectory Planning of Robot Manipulator Based on RBF Neural Network. Entropy 2021, 23(9), 1207.
27. Carron, A.; Arcari, E.; Wermelinger, M.; Hewing, L.; Hutter, M.; Zeilinger, M. N. Data-Driven Model Predictive Control for Trajectory Tracking with a Robotic Arm. IEEE Robotics and Automation Letters 2019, 4(4), 3758-3765.
28. Wang, W.; Tao, Q.; Cao, Y.; Wang, X.; Zhang, X. Robot Time-Optimal Trajectory Planning Based on Improved Cuckoo Search Algorithm. IEEE Access 2020, 8, 86923-86933.
29. Moradi, M.; Naraghi, M.; Nikoobin, A. Indirect optimal trajectory planning of robotic manipulators with the homotopy continuation technique. In Proceedings of the International Conference on Robotics and Mechatronics, Tehran, Iran, 15-17 October 2014, pp. 286-291.
30. Diaz Arango, G.U. Robotic motion path generation method based on homotopy continuation for multidi-mensional workspaces. PhD, National Institute for Astrophysics, Optics and Electronics, Puebla, 2018.
31. Gallardo-Alvarado, J. An Application of the Newton-Homotopy Continuation Method for Solving the Forward Kinematic Problem of the 3-RRS Parallel Manipulator. Mathematical Problems in Engineering 2019.
32. Rice, J.J.; Schimmels, J.M. Multi-homotopy class optimal path planning for manipulation with one degree of redundancy. Mechanism and Machine Theory 2020, 149, 103834.
33. Li, H.; Savkin, A.V. An algorithm for safe navigation of mobile robots by a sensor network in dynamic cluttered industrial environments. Robotics and Computer-Integrated Manufacturing 2018, 54, 65-82.
34. Ranganeni, V.; Chintalapudi, S.; Salzman, O.; Likhachev, M. Effective footstep planning using homotopy-class guidance. Artificial Intelligence 2020, 286, 103346.
35. Wang, Y.; Topputo, F. A TFC-based homotopy continuation algorithm with application to dynamics and control problems. Journal of Computational and Applied Mathematics 2022, 401, 113777.
36. Quiñonez, Y.; Mejía, J.; Zatarain, O.; Lizarraga, C.; Peraza, J.; Estrada, R. Algorithm to Generate Trajectories in a Robotic Arm Using an LCD Touch Screen to Help Physically Disabled People. Electronics 2021, 10, 104.
37. Quiñonez, Y.; Zatarain, O.; Lizarraga, C.; Aguayo, R.; Mejía, J. A Novel Method Based on Numerical Iterations for Joint-Space Trajectory Generation with a via Point. In New Perspectives in Software Engineering, Mejia, J., Muñoz, M., Rocha, Á., Avila-George, H., Martínez-Aguilar, G.M. (eds). Springer, Cham: Coahuila, México, 2022, vol 1416, pp. 189–204.
38. Jeong, S. Y.; Choi, Y. J.; Park, P.; Choi, S. G. Jerk Limited Velocity Profile Generation for High Speed Industrial Robot Trajectories. IFAC Proceedings Volumes 2005, 38(1), 595-600.
39. Sengupta, A.; Chakraborti, T.; Konar, A.; Nagar, A. Energy efficient trajectory planning by a robot arm using invasive weed optimization technique. In Proceedings of Third World Congress on Nature and Biologically Inspired Computing, Salamanca, Spain, 19-21 October 2011, pp. 311-316.
40. Števo, S; Sekaj, I.; Dekan, M. Optimization of Robotic Arm Trajectory Using Genetic Algorithm. IFAC Proceedings Volumes 2014, 47(3), 1748-1753.