

Article

# Cam Mechanisms Reverse Engineering Based on Evolutionary Algorithms

Monica Tiboni , Cinzia Amici  and Roberto Bussola

Department of Mechanical and Industrial Engineering, University of Brescia, Via Branze, 38, 25123 Brescia, Italy; cinzia.amici@unibs.it (C.A.); roberto.bussola@unibs.it (R.B.)

\* Correspondence: monica.tiboni@unibs.it

**Abstract:** Cam follower mechanisms are widely used in automated manufacturing machinery to transform a rotary stationary motion into a more general required movement. Reverse engineering of cams has been studied, and some solutions based on different approaches have been identified in the literature. This article proposes an innovative method based on the use of an evolutionary algorithm for the identification of a law of motion that allows for approximating in the best way the motion or the sampled profile on the physical device. Starting from the acquired data, through a genetic algorithm, a representation of the movement (and therefore of the cam profile) is identified based on a type of motion law traditionally used for this purpose, i.e., the modified trapezoidal (better known as modified seven segments). With this method it is possible to estimate the coefficients of the parametric motion law, thus allowing the designer to further manipulate them according to the usual motion planning techniques. In a first phase, a study of the method based on simulations is carried out, considering sets of simulated experimental measures, obtained starting from different laws of motion, and verifying whether the developed genetic algorithm allows for identifying the original law or approximating one. For the computation of the objective function, the Euclidean norm and the Dynamic Time Warping (DTW) algorithm are compared. The performed analysis establishes in which situations each of them is more appropriate. Implementation of the method on experimental data validates its effectiveness.

**Keywords:** evolutionary algorithms; reverse engineering; cam mechanisms; law of motion; genetic algorithms



**Citation:** Tiboni, M.; Amici, C.; Bussola, R. Cam Mechanisms Reverse Engineering Based on Evolutionary Algorithms. *Electronics* **2021**, *10*, 3073. <https://doi.org/10.3390/electronics10243073>

Academic Editors: Youye Xie, Kai Liu and Zhihui Zhu

Received: 5 November 2021

Accepted: 7 December 2021

Published: 9 December 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Cam follower mechanisms are the classical mechanical solution to transform, with a high level of precision, a rotary stationary motion into a translating or oscillating or indexing movement. More in general, they transform the cam's motion in a prescribed movement of the follower [1,2]. Although cam devices allow for rigid automation, compared to modern flexible electronic cams, they have some advantages that make them the optimal solution in a wide spectrum of engineering applications. Textile machines, automobile engines, transfer machines, printing presses and bottling machinery are just some examples of systems where cam follower mechanisms are widely adopted. The main characteristics of engineering cam-based solutions, in addition to the aforementioned precision, are simplicity, compactness and high velocities.

Several research topics deal with cam's devices, and many publications treat the design optimization. Flores [3] presents a general and comprehensive procedure for an optimal geometric parameters identification for a cam with translating roller follower, obtaining a constrained optimization multivariate problem, solved with non-linear programming algorithms. Meta-heuristics algorithms are widely used for the solution of the problem arising from the cam design [4,5], and among them the evolution-based category is the most investigated one for this problem. Genetic algorithms (GA) have been used, for example, by Lampinen [6] for cam shape design automation and optimization, or by Tsiafis et al. [7] for the identification of the optimal values for the design parameters, like the

cam base circle radius, the follower offset and the face width. The automatic cam profile synthesis is also a constrained non-linear optimization problem for which many solutions based on GA have been proposed: in 2017 [8], a multi-objective GA was developed by Jana and Bhattacharjee, and applied to cam mechanisms with simple and double harmonic profiles; Hamza et al. in 2018 [9] used a modified adaptive differential evolution (MADE) algorithm for the optimal design of a cam mechanism with offset translating roller follower. The dynamic behavior of high-velocity cams is another important issue, and the cam synthesis plays a very significant role on follower vibratory phenomena. Dynamic modelling and optimization procedures are used to synthesize cam curves to improve the dynamic behavior: in 2020 [10], Yu et al. applied GA to find the optimal solution in a constraint optimization problem based on a single degree of the freedom dynamic model; in 2017, an optimization of the cam profile based on mathematical modelling was applied to an offset press by Ouyang et al. [11]. The follower vibrations control is another main research issue in the field of cam mechanisms [12–15], and it is normally combined with the condition monitoring of these systems [16–18].

In the last decades, computer-aided design (CAD) has become widely consolidated, favored by increasingly higher velocities and computing powers and by the integration of different tools in the same design environment. A need that often arises in industrial practice is to reconstruct or to model a part of a system starting from an already existing physical object, in order to update or improve the design of an existing component. Reverse engineering (RE), which dates back to the 1980s, is the solution for such a problem. RE methods aim at the reconstruction of CAD models from measured data [19] and are applied in different fields including medicine [20], dentistry [21] and mechanical engineering [22].

Reverse engineering applied to cam mechanisms has been investigated by many scholars: Li in 2011 [23] presented the reverse design of a spatial cam based on the measurements achieved through a coordinate measuring machine (CMM) and the mathematical reconstruction of the cam's surface; Zhenghao et al. in 2011 [24] proposed an iterative process consisting of fitting the cam profile using the non-uniform rational basis-splines (NURBS) curves, starting from the measured points, then making a further kinematic analysis and a possible correction of the profile, to obtain a cam profile that is smooth but that still satisfies the boundary conditions in terms of displacement, velocity and acceleration of the follower; Alsoufi et al. [25] presented a cam RE converting the scanned data using coordinate-measuring machine (CMM) technology into a 3D model of the disc CAM and investigated the obtained results in terms of the height variation, dimensional accuracy and surface roughness as well as skewness and kurtosis performance with cams of different materials (including aluminum, hardened steel, stainless steel, PLA+, PLA, ABS+ and photo-polymer resin).

This paper proposes an innovative reverse engineering method for a cam mechanism, based on the identification of a law of motion that better approximates the one generated by the device, having acquired the cam profile through a CMM machine or having measured the follower movement. With this method, the coefficients of a parametric motion law can be estimated, thus allowing the designer to further manipulate them according to the usual motion planning techniques. The coefficients values are identified with an optimization process, by setting a first series of values a priori and by correcting them through the minimization of the error between the acquired data and the resulting curve. The minimization is implemented with an evolutionary algorithm (a genetic algorithm) capable of properly probing the solutions space. However, it is important to adequately define a quantity that represents the abovementioned error, since this quantity can make the effectiveness or the convergence of the method critical. In the current work, two methods were used to quantify the difference between the two curves for comparison purposes: the classical Euclidean norm (which measures the curves distance) and the Dynamic Time Warping (DTW) algorithm, capable of capturing and assessing the similarity between two trends.

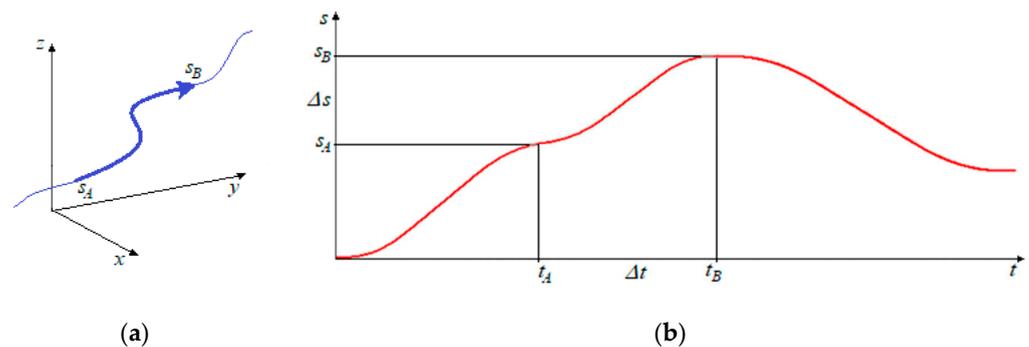
The innovative aspects, with respect to the literature, concern the development of a method whose purpose is to identify the law of motion, instead of directly reconstructing the profile of the cam from the acquired points. The use of evolutionary algorithms within a RE techniques represents a further innovative aspect of the developed research. The law of motion and cam profile are strictly connected. The follower law of motion and stroke are usually selected in accordance with the technological process and are set as the design initial data. The synthesis of the cam, which consists of sizing the cam and of finding the CNC center tool coordinates to profile the cam, follows. The cam sizing is greatly influenced by the law of motion choices, e.g., the maximum velocity influences the dimensioning of the cam's base radius. The choices for the cam manufacturing, such as the material, the production technology, the tolerances chosen in the design phase and the accuracy of the CNC machinery, can instead influence the follower law of motion. Samsonenko in 2020 in [26] researched the influence of errors of cam manufacturing on the law of motion of the output element and concluded that slight deviations of the cam shape from the ideal can lead to significant deviations of the movement's law of the follower. Therefore, a correct choice of the manufacturing method of the cam and of its tolerances is fundamental for the effectiveness of the proposed method.

The paper is organized as follows: Section 2 describes the main concepts related to motion planning and the formulation of a modified trapezoidal motion law in the most general case, the genetic algorithm, the objective function used and the structure of the developed method. Section 3 reports the application of the procedure in some fictitious illustrative cases at the beginning, where the method is tested on data generated by simulation, and afterwards describes the application to a real case. The results are then analyzed in Section 4, and Section 5 summarizes the salient results of the work and the future developments.

## 2. Materials and Methods

### 2.1. Laws of Motion and Motion Planning

The movement of a cam follower (or, more generally, of a machine element) is identified by the trajectory (Figure 1a) and by the law of motion (Figure 1b), which defines how the trajectory is followed with respect to the time.



**Figure 1.** (a) Example of the trajectory and (b) example of the position law of motion.

Often, a law of motion is constituted by a succession of sections characterized by displacement,  $\Delta s$ , and related temporal duration,  $\Delta t$ . Therefore, in the study, we can focus on a single trait and the same considerations can be extended to the others as well. In addition to duration and displacement, the law of motion sometimes respects further constraints, such as the values of velocity or the successive derivatives to the extremes of the interval or the imposed passage for a specific point at a given instant. Some motion laws have a flexibility that makes this possible, unlike others, which are rigid and do not allow for solving these kinds of problems. For now, let us consider the case in which the motion law segment has null values at the extremes for all the quantities except for the

displacement; this represents a very common case in the definition of the requirements of a cam mechanism.

Referring to the first two intervals of the law of motion in Figure 1, the quantities  $t_A$ ,  $s_A$ ,  $t_B$  and  $s_B$  (or similarly  $t_A$ ,  $s_A$ ,  $\Delta t$  and  $\Delta s$ ) are constraints, whereas the definition of the curve shape is based on a set of characteristic coefficients that depend on the motion law type. The displacement is a function of the time, of the parameters of the law of motion ( $c = [c_0, c_1, \dots, c_n]$ ) and of the constraints. According to the classical law of motion definition, the  $c_i$  coefficients are adimensional values. If we denote  $\lambda$  as the vector of constraints, which is assigned, in the most general case the displacement function can be expressed according to Equation (1).

$$s(t) = s(t, c, \lambda) \quad (1)$$

Some of the coefficients  $c$  are determined by imposing the constraints defined by  $\lambda$ , whereas others can be chosen within a certain margin to further characterize the curve. A separation is not introduced between these two subgroups because often the coefficients are dependent on each other as a whole. For example, considering a polynomial law of motion, as in Equation (2), it is possible to choose the degree  $n$  of the polynomial, in order to both satisfy a certain number of constraints at the extremes of the interval, and introduce further parameters that allow for manipulating the shape of the curve (for example by imposing a certain velocity or acceleration value, or jerk in a generic instant).

$$s(t) = c_0 + c_1(t - t_A) + c_2(t - t_A)^2 + \dots + c_n(t - t_A)^n \quad (2)$$

In this case, denoting the velocity with  $v$ , we could impose the following four constraints:

$$s(t_A) = s_A, \quad v(t_A) = v_A, \quad s(t_B) = s_B, \quad v(t_B) = v_B \quad (3)$$

Then, we can define  $\lambda = \{t_A, s_A, v_A, t_B, s_B, v_B\}$ . Given four constraints, the minimum value of  $n$  to satisfy them is three, and consequently, four coefficients are required. If the coefficients' number is equal to the constraints' number, the equations for calculating the coefficients derive from the imposition of the constraints. However, it is also possible to choose a degree  $n$  of the polynomial redundant with respect to the constraints, and to use the additional degrees of freedom to manipulate the shape of the curve. For example, in this case we could choose  $n$  equal to six, obtaining a vector  $c$  made up of six elements, that is, in general, a number greater than the number of constraints. Returning to the more general case of a law of motion, even not a polynomial, the main difficulty consists in determining the value of the coefficients  $c_i$ , which are sometimes not determinable in closed form. In these cases, for the computation of the coefficients, it is reasonable to use numerical methods based on iterative algorithms, which are able to offer approximate solutions for the additional degrees of freedom in compliance with all other constraints.

Without treating the formulation of the different types of motion law, for which reference is made to the literature [1], here the case of trapezoidal modified seven traits motion laws is especially depicted as an illustrative application. The modified trapezoidal laws are particularly manageable, suitable for the purposes of the presented work and very widespread in the industrial field. These motion laws fall within the more general class of the *piecewise* motion laws, which consist of the succession of different parts, each defined by a specific mathematical function. In detail, the modified trapezoidal laws are defined by assigning the shape of the acceleration diagram consisting of an alternation of seven sinusoidal and constant strokes. With reference to Figure 2, the total duration  $\Delta t = t_B - t_A$ , in which the displacement  $\Delta s = s_B - s_A$  occurs, is divided into seven sections whose single duration is identified by the vector of coefficients  $c = \{c_0, c_1, \dots, c_6\}$ . In addition to displacement, non-zero values of both velocity and acceleration can be imposed on the extremes of the law of motion, indicated in the figure by  $v_A$ ,  $a_A$ ,  $v_B$  and  $a_B$ . Displacement, velocity and acceleration trend can be consequently determined with a double integration in an analytical form (in parts). The additional parameters  $A_{pos}$ ,  $A_{neg}$  and  $V_{max}$ , introduced for convenience, represent the maximum positive and negative acceleration and maximum

velocity, respectively. Therefore, in the general case, which is characterized by a duration and a displacement compatible with the velocity and acceleration values assigned to the extremes, the profile can be defined in all its parts thanks to the imposition of the constraints (Equation (4)).

$$s(t_A) = s_A, v(t_A) = v_A, a(t_A) = a_A, s(t_B) = s_B, v(t_B) = v_B, a(t_B) = a_B \quad (4)$$

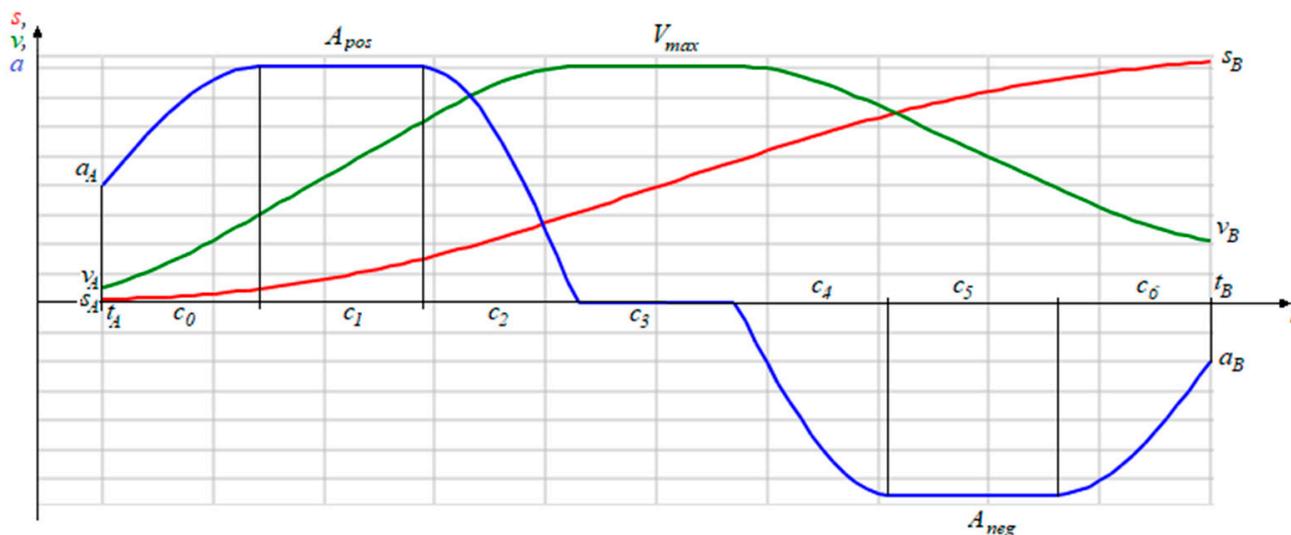


Figure 2. Modified seven traits motion law, with imposed constraint values on displacement and velocity.

At a computational level, it is convenient to first calculate the vectors of cumulative partial displacements  $s_c$  and cumulative partial velocities  $v_c$  at the end of each section, introducing some useful quantities, called  $a_{11}, a_{22}, a_{12}, a_{21}, b_1$  and  $b_2$ , and defined as shown in the formulas in Table A1.

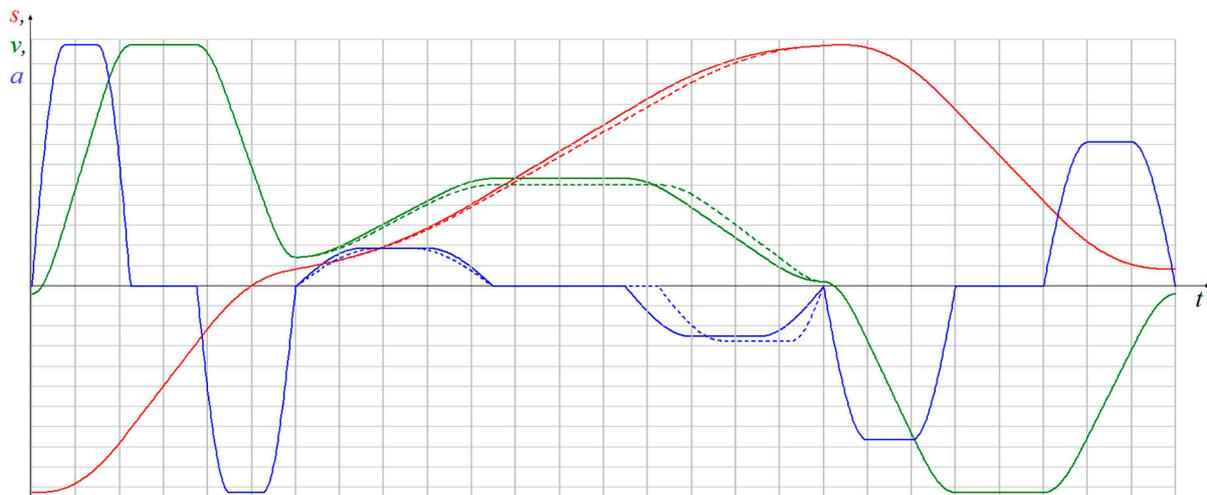
For the  $i$ th segment, let us define a function  $t_i(t)$ , which expresses the instant of start, as shown in Equation (5). It is possible to define the analytic expression of displacement, velocity and acceleration at time  $t$  ( $s(t), v(t)$  and  $a(t)$ ), respectively, obtaining the formulation shown in Table A2, in which the symbol  $t_i$  is used in place of  $t_i(t)$ , omitting the time instant  $t$ , for compactness reasons, and where  $t_{ci}$  indicates the instant of time at the beginning of the  $i^{th}$  section (Equation (6)).

$$t_i(t) = \begin{cases} t - t_A & \text{for } i = 0 \\ t - t_A - \sum_{k=0}^{i-1} c_k & \text{for } i \geq 1 \end{cases} \quad (5)$$

$$t_{ci}(t) = \begin{cases} t_A & \text{for } i = 0 \\ t_A + \sum_{k=0}^{i-1} c_k & \text{for } i \geq 1 \end{cases} \quad (6)$$

As previously stated, this class of motion law is very flexible, since it allows for guaranteeing the constraints  $\lambda = \{t_A, s_A, v_A, a_A, t_B, s_B, v_B, a_B\}$  at the extremes on displacement as well as on velocity and acceleration, but also allows for further manipulations of its shape in order to achieve secondary objectives.

As an example, Figure 3 shows the comparison of two laws of motion consisting of three movements, in which the dashed modified trapezoidal of the central movement was obtained from the continuous curve by slightly modifying the values of the coefficients  $c_4, c_5$  and  $c_6$  of the vector  $c$ . Often, the coefficients  $c$  are in a normalized form with respect to the total motion duration  $\Delta t$ , to make comparisons independent of the cycle time.



**Figure 3.** Modified seven traits acceleration motion law (blue curve), with imposed constraint values on displacement (red curve) and velocity (green curve).

## 2.2. Problem Definition and Genetic Algorithm Implementation

Referring to the laws of motion described in the previous paragraph, the proposed cams reverse engineering method consists of an optimization problem for which the objective function is the distance between two curves. The first curve is obtained starting from the CMM measurements of the profile or direct measurement of the follower's motion; the other one is generated as a sampling of the modified trapezoidal motion law, obtained with the particular vector of seven coefficients  $c$ , which defines the potential solution. Denoting the set of law of motion parameters to be identified with  $c = \{x_1, x_2, \dots, x_n\}$  and the objective function with  $F = F(c)$ , the problem consists in the minimization of  $F$ , therefore determining the particular set of values  $c^*$ , such that  $F(c^*) = \min F(c)$ . Due to the characteristics of this problem, a genetic algorithm represents a valid solution technique.

Genetic algorithms [27] belong to the family of evolutionary algorithms, which adopt the mechanisms of natural selection to solve optimization problems. The use of genetic algorithms is the most suitable method for problems with a large number of parameters or when the function to be optimized is not sufficiently regular or has a domain consisting of several parts that are not connected to each other. In a nutshell, a genetic algorithm, starting from a first set of solutions that are usually randomly generated, carries out the recombination of groups (generations) of solutions (individuals), each consisting of specific characters (variables of the optimization problem).

Each new solution is obtained by recombining, by means of genetic operators, the characters of two solutions selected from the previous group. Selection favors the choice of solutions that best minimize the objective function, so that the properties of the best solutions of a group can propagate in subsequent groups. In addition to the traditional genetic operators (selection, cross-over, mutation, inversion), the algorithm developed in this work uses a further operator, which involves a mutation of the coefficients that is not random, but is bound to a neighborhood of their previous value, thus enabling it to better explore a specific area. As the procedure evolves, the amplitude of this area is narrowed so as to refine the solution in an area that is already optimal in itself. The operators are not always applied, but each one presents a certain probability of occurrence. To ensure the convergence of the method, it was decided to reintroduce in each new generation the best individual obtained in the previous one, thus adopting an elitist algorithm.

As for many optimization algorithms, genetics have intrinsic parameters that affect their effectiveness, such as the number of individuals and the probability of application of the different operators. The chosen parameter values are reported in Section 3.

### 2.3. Objective Function

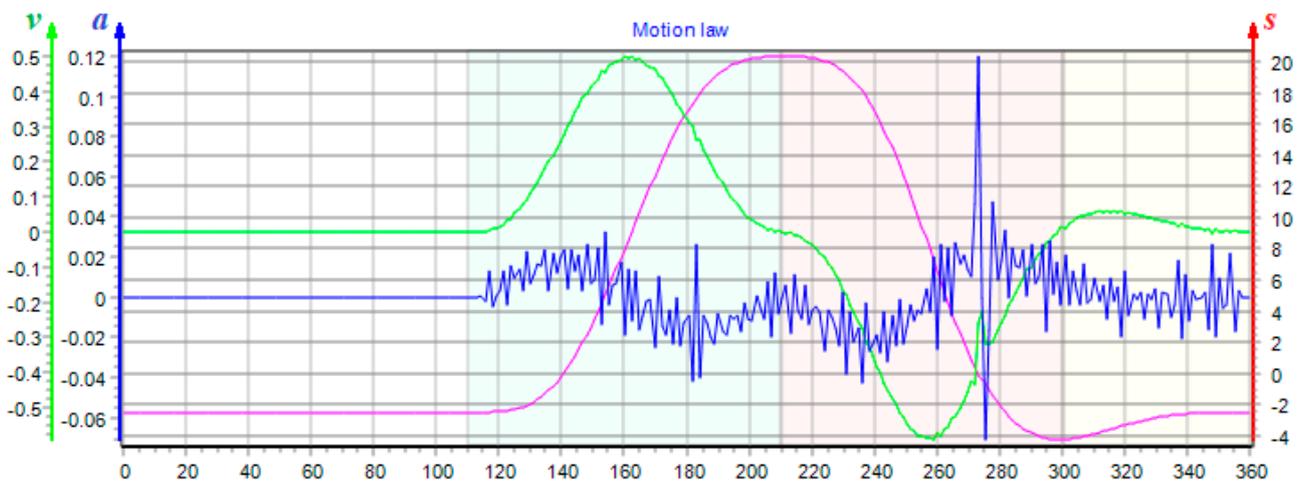
The objective function is an indicator of how much close the identified law of motion and the original one are and could be calculated considering either the position or the velocity or the acceleration. The comparison based on the displacement has an important negative aspect: even large variations of the law of motion, for example in terms of accelerations, are very slightly perceived at displacement level. To overcome this drawback, comparing the curves as time derivatives, therefore at the velocity or acceleration level rather than as displacements, is more effective. In the case of the modified trapezoidal, a further advantage emerges when considering the acceleration, since the law of motion is defined starting exactly from the acceleration diagram itself.

To measure the dissimilarity between two curves, the most commonly used method is the Euclidean norm (EN), i.e., the distance (Equation (7)) between the vectors representing the curves. If we denote  $\hat{x}$  and  $x$  as these vectors, the EN is evaluated as follows:

$$d(\hat{x}, x) = \sqrt{\sum_j (\hat{x}_j - x_j)^2} \quad (7)$$

Besides, the EN is not always the ideal tool for measuring the distance between the curves for the problem under consideration.

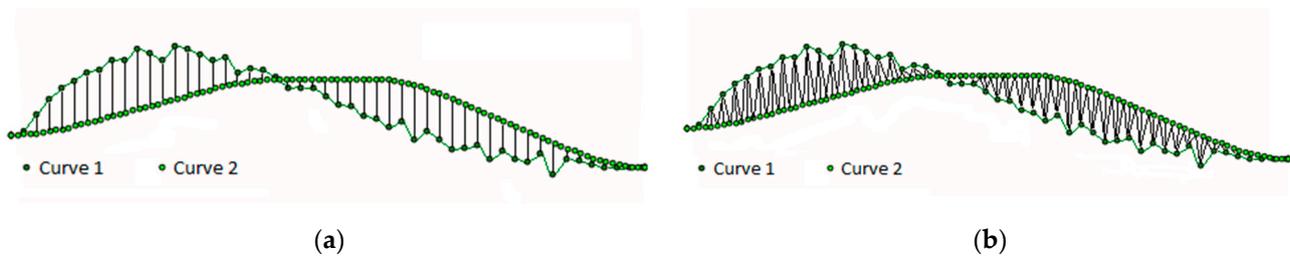
Indeed, the acquired data are frequently subjected to noise and, sometimes, disturbances (due to a surface damage, for example), which are amplified in the derivation of the signal and can affect the results. This usually makes the comparison between the analytic law of motion and the experimental one critical, and even more so if an evolutionary algorithm is used. As an example, Figure 4 shows the acquired displacement (red curve), of the follower movement of a rocker arm cam, damaged near the point with abscissa  $270^\circ$ , and the computed velocity (green curve) and acceleration (blue curve) signals. The data reveal a peak in the acceleration that could greatly affect the Euclidean norm.



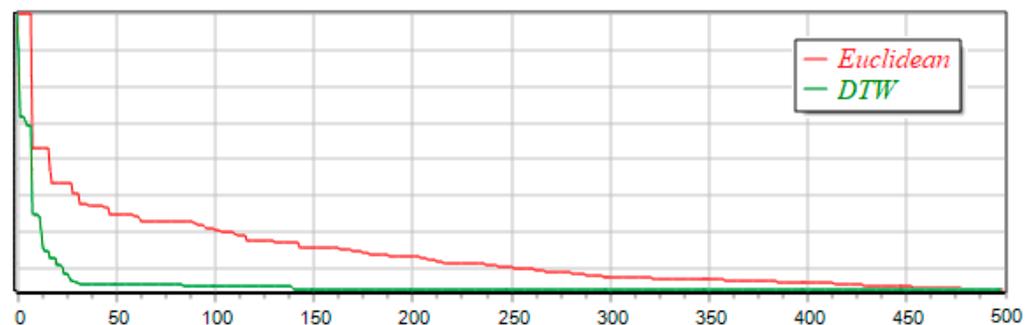
**Figure 4.** Acquired motion law of the follower movement of a damaged rocker arm (blue: acceleration; green: velocity; red: displacement). The law of motion is defined as a function of the rotation angle of the cam, and the displacement referring to the red vertical axis is a rotation of the follower in degrees.

As alternative method for measuring the similarity between the curves, the Dynamic Time Warping (hereinafter DTW) algorithm [28–30] was considered, which in some situations may be more effective than the Euclidean norm. With this technique, each point of a curve is compared with some points of the other curve in order to have a more reliable measurement of *similarity*. Let us consider curves 1 and 2 of Figure 5: with Euclidean distance, each point of curve 1 is compared with the corresponding point of curve 2 (Figure 5a), whereas the DTW algorithm compares different points between the two curves (Figure 5b). The adoption of the DTW technique allows for an optimization that tends to avoid falling

and remaining in relative minimums. Obviously, the use of a complex algorithm for calculating the distance between the curves penalizes the computation time, but, on the other hand, it allows the algorithm to converge faster. As a demonstration of this, Figure 6 shows the trend of the objective function, over the generations of the genetic algorithm, in the case of the Euclidean norm (red curve) and with the DTW algorithm (green curve), for an optimization in 500 generations with 100 individuals, and with a distance calculation on 500 samples.



**Figure 5.** (a) Curves comparison approach with the Euclidean norm; (b) curves comparison approach with the DTW algorithm.



**Figure 6.** Comparison of the objective function vs. generations with the Euclidean norm (red curve) and DTW algorithm (green curve).

In Figure 6, no values are reported on the ordinate axis because the DTW algorithm and Euclidean norm are not directly comparable, but the profiles' trends emphasize that DTW allows for faster convergence. In terms of computation time over 500 generations, the Euclidean norm requires about 4 ms whereas DTW algorithm requires about 6 ms; however, as the figure depicts, DTW has a higher convergence rate. With the DTW algorithm, the goal can be considered achieved at about 150 generations, with a noticeable reduction in convergence time (about 1.8 ms).

#### 2.4. Developed Cam Mechanisms RE Framework

The RE framework generally adopted and commonly referred to in the literature [31–33] is based on the following six main phases: data capture, pre-processing, segmentation, feature classification, surfaces fitting and CAD model reconstruction. Several methodologies are available for each one of these phases and an in-depth analysis of these techniques is discussed in [19]. This approach can also be applied to realize the RE of a cam, as done in [25], which presents the process of reversed disc CAM from an existing one. Alsofi focuses on the strategy for converting the scanned data using coordinate measuring machine (CMM) technology in the form of point cloud data into a 3D model of the disc CAM. Different approaches for the RE of cam mechanisms are also present in the literature, which have the aim of tracing the motion law of the follower of the cam mechanism starting from the sampling of the cam profile. In [34], Tsay considers a planar cam and uses a least squares periodic B-splines approximation to interpolate the digitized cam surfaces and to create their derivatives required in determining the follower motion. In [24] and in [35], Zhenghao presented a similar approach, with the improvement of obtaining a smoother

cam profile, while still exactly satisfying the boundary condition of follower displacement, velocity and acceleration, and used NURBS curves to fit the profile via the measured data.

The proposed cam's RE differs from the RE classical approach. Although the current approach shares with the latest cited works the purpose of tracing the follower's law of motion, it significantly differs from them in the method. In fact, rather than performing an interpolation of the points of the cam profile, the here proposed identification process focuses on the law of motion that generates a cam's profile as close as possible to the measured one, through the optimization process of an objective function. The main advantage of this approach consists in the fact that it enables exploiting at best the flexibility provided by parametric laws. Indeed, if we use this method and a very flexible parametric law, such as the modified trapezoidal, we have the possibility to control, and within certain limits to intervene on, speed and acceleration trends, with all the advantages that influence the dynamic behavior and the dimensioning of the cam. A GA-based optimization procedure allows for identifying the coefficients of a trapezoidal modified motion law, which best approaches the law of motion of the device to be reengineered, and the objective function is evaluated through the DTW algorithm or the Euclidean norm. The flowchart of this method is shown in Figure 7.

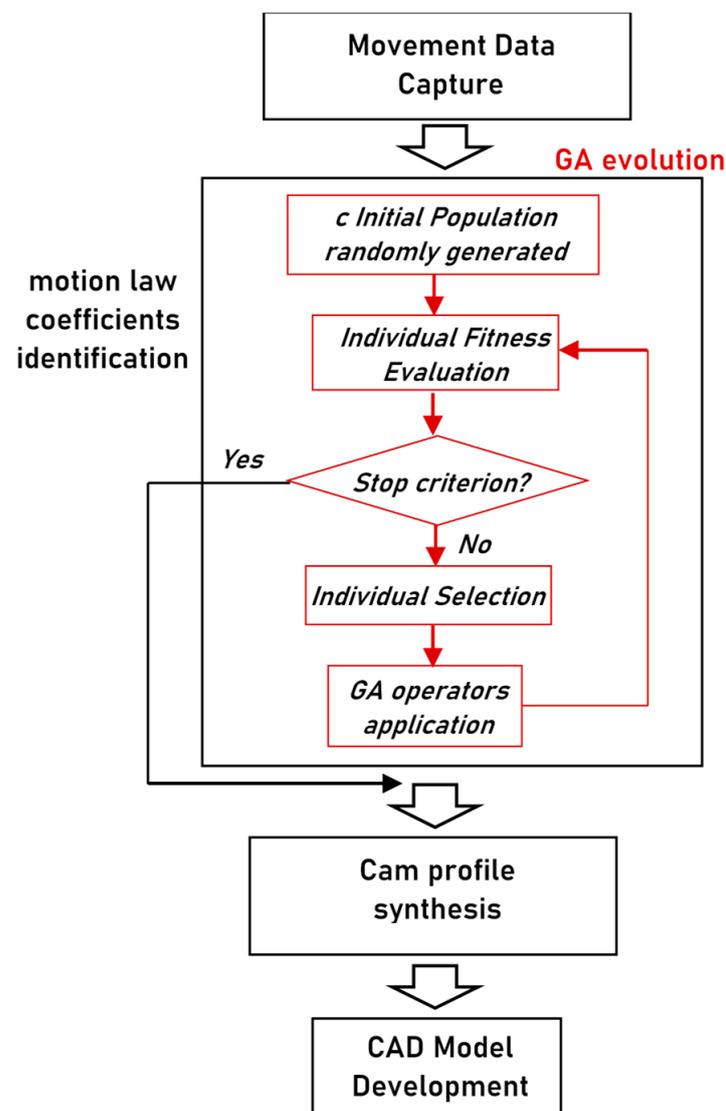


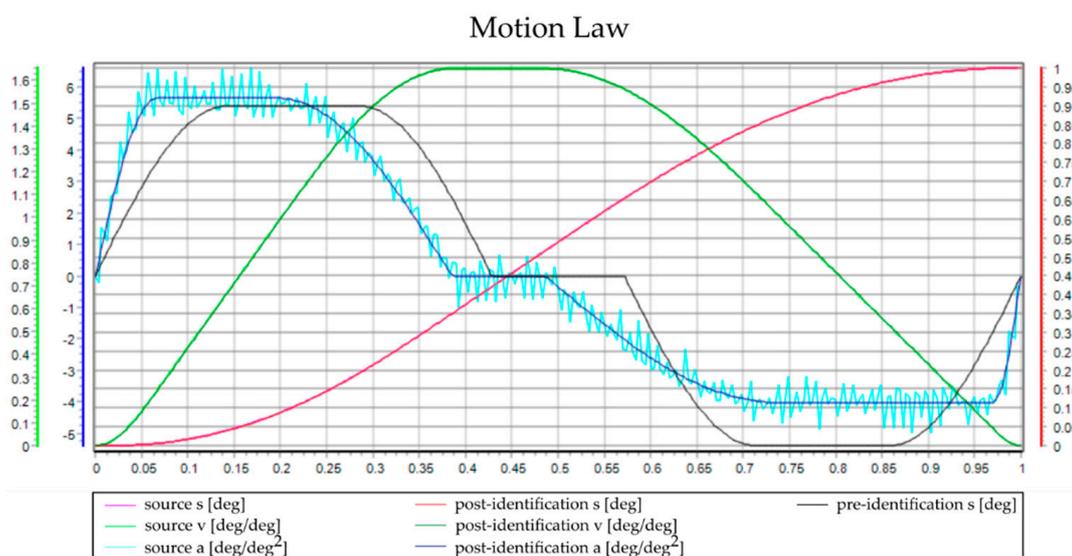
Figure 7. Flowchart of the developed cam mechanism RE.

Basically, the idea is to redesign the cam, but respecting the functionality performed by a pre-existing cam, for example, a damaged cam or one of which the original project is not available. The choice of the modified trapezoidal motion law is motivated by the fact that numerous motion laws commonly used in industrial applications can be represented through the mathematical formulation of this law. In the optimization process, it may occur sometimes that some coefficients have a null value, even simultaneously, and this can lead to inconsistencies or calculation errors, such as divisions by zero. Therefore, in the software implementation of the formulas presented in Section 2.1, checking tests are inserted in order to degenerate, if necessary, the seven segments law of motion into a three segments law, or to report a warning of the impossibility of satisfying all conditions. The presented innovative cam mechanism reverse engineering method was implemented in an executable software developed in Delphi 10.4 environment

### 3. Results

#### 3.1. Simulations

Before testing the method on actual data, simulated data were used for preliminary analysis and validation. Figure 8 shows an example of application of the method to a simulated case. Loaded data were generated through a modified trapezoidal motion law (with non-zero initial and final values for velocity and acceleration), in which a noise with zero mean and unitary amplitude was added to the acceleration (light-blue curve). This law of motion realizes a unitary displacement in a unitary time (1 rad in 1 s, as for all the laws of motion in simulation) and is subjected to the constraints  $\lambda$  and the normalized coefficients  $c_{src}$  synthesized in Table 1. These coefficients were chosen randomly, but without null values, so that all the segments are present.

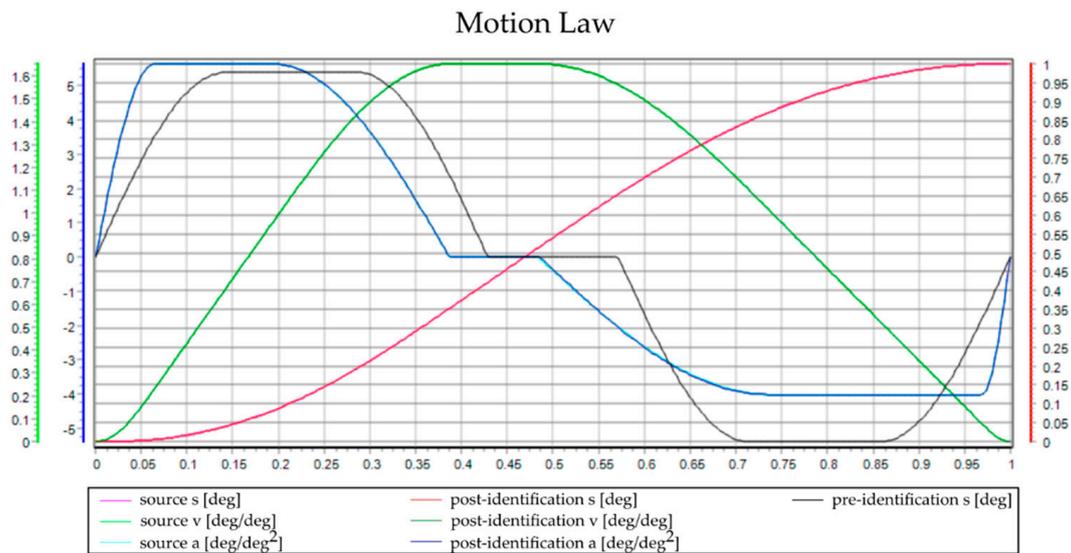


**Figure 8.** Application of the method to a simulated case, with random disturbance added to acceleration. (Black: acceleration of the starting step of the identification process; red: displacement; green: velocity; blue: acceleration, after the identification).

**Table 1.** Values of the main parameters for the law of motion of the simulated scenario.

Parameter	Value
Constraints	$\lambda = \{t_A = 0.0s, s_A = 0.0rad, v_A = 0.2rad/s, a_A = 1.4rad/s^2, \dots, t_B = 1.0s, s_B = 1.0rad, v_B = 0.5rad/s, a_B = 1.2rad/s^2\}$
Normalized coefficients	$c_{src} = \{0.0645, 0.129, 0.1935, 0.0968, 0.2581, 0.2258, 0.0323\}$
Number of sampled points	300

At the beginning of the identification procedure based on GA, an initial solution with all the seven coefficients of  $c$  equal to  $1/7$  was chosen (black curve in Figure 9). The blue curve is the law of motion obtained at the end of the optimization procedure with Euclidean norm evaluated on the acceleration as an objective function. As can be seen, it matches the simulated experimental data very well (light-blue curve). In this case the Euclidean norm proved to be adequate, allowing for convergence after a limited number of generations. The convergence time was of about  $2/5$  s. The software displayed the acceleration, the velocity and the displacement diagrams, which, however, were substantially superimposed. The absolute maximum error on the coefficients was about 0.0076.



**Figure 9.** Simulated case, without simulated disturbances. (Black: acceleration of the starting step of the identification process; red: displacement; green: velocity; blue: acceleration, after the identification).

Figure 9 shows the case with the same shaped law of motion, but without noise. The error on coefficients in this case was 0.00034, so it was reduced by an order of magnitude. This case completely validates the procedure, since it substantially offers the values of the coefficients of the loaded motion law. As visible in the lower part of Figures 8 and 9, the genetic algorithm converges quite quickly, reaching a good reduction of the objective function already after 50 generations. This case, completely general, was used to tune the genetic algorithm, i.e., to establish the values of the algorithm intrinsic parameters that can be used for the developed RE method. After a high number of tests, the values shown in Table 2 were chosen.

**Table 2.** Genetic algorithm intrinsic parameters’ chosen values.

Parameter	Chosen Value
Number of genes for each individual	7
Number of individuals per generation	100
Probability of cross-over application	0.85
Inversion application probability	0.1
Probability of application of mutation around the range	0.1
Percentage of initial range mutation	10%
Mutation range reduction	75% each 50 generations

Keeping the intrinsic parameters of the genetic algorithm fixed (Table 2), tests for nine different cases were carried out, related to the different motion laws used to generate simulated input data. In the following sections, we refer to these laws of motion as “original”, in the sense that they originated the data that replaced the experimental measurements

in simulation. All these motion laws can be represented through the formulas of the modified trapezoidal law; that is, they fall within the class of modified trapezoids. For all the considered cases, a unitary displacement in a unitary time, null constraint vector and 300 extracted sampled points were assumed. The maximum number of generations was set 500, or less in case the algorithm no longer evolves.

Table 3 contains the values of the seven coefficients of the considered laws of motion, whereas Table 4 collects the results of the identification in four cases: identification on acceleration, identification on velocity and for each one adopting either the Euclidean norm or DTW algorithm in the objective function calculation.

**Table 3.** Coefficients’ values of the laws of motion of the modified trapezoidal class used as the original law of motion.

ID	Original Law of Motion	$c_1$	$c_2$	$c_3$	$c_4$	$c_5$	$c_6$	$c_7$
L1	Modified Sinusoidal	0.125	0	0.3750	0	0.3750	0	0.1250
L2	Constant Acceleration	0	0.5000	0	0	0	0.5000	0
L3	Trimmed Constant Acceleration	0	0.2500	0	0.5000	0	0.2500	0
L4	Cubic	0	0	0.500	0	0.500	0	0
L5	Trapezoidal	0.1250	0.2500	0.1250	0	0.1250	0.2500	0.1250
L6	Modified Trapezoidal	0.1250	0.1250	0.1250	0.1250	0.1250	0.1250	0.1250
L7	Trimmed Trapezoidal	0.0625	0.1250	0.0625	0.5000	0.0625	0.1250	0.0625
L8	Cycloidal	0.2500	0	0.2500	0	0.250	0	0.2500
L9	Asymmetric Trapezoidal	0.0500	0.2000	0.0500	0	0.050	0.6000	0.0500

**Table 4.** Identified coefficients’ values (dimensionless) and errors in the simulated cases with laws of motion of the modified trapezoidal class used as the original law of motion. (The method column refers to the method used to calculate the objective function.)

ID	Kinematic Quantity	Method	$c_1$	$c_2$	$c_3$	$c_4$	$c_5$	$c_6$	$c_7$	Max. Error
L1	Acceleration	EN	0.1218	0.0159	0.3598	0.0030	0.3598	0.0169	0.1226	0.0169
		DTW	0.1240	0.0101	0.3617	0.0073	0.3683	0.0033	0.1250	0.0133
	Velocity	EN	0.1246	0.0212	0.3513	0.0091	0.3521	0.0079	0.1335	0.0236
DTW		0.0992	0.1131	0.2646	0.0464	0.2646	0.1118	0.1001	0.1131	
L2	Acceleration	EN	$2.0 \times 10^{-8}$	0.5000	0	0	0	0.5000	$6.7 \times 10^{-8}$	$6.7 \times 10^{-8}$
		DTW	0	0.5000	$1.3 \times 10^{-9}$	$1.5 \times 10^{-9}$	$1.3 \times 10^{-1}$	0.5000	0	$1.5 \times 10^{-9}$
	Velocity	EN	0.0007	0.4922	0.0063	$5.5 \times 10^{-9}$	0.0100	0.4907	0	0.01000
DTW		0.0113	0.4302	0.0536	0	0.0629	0.4302	0.0116	0.0697	
L3	Acceleration	EN	$2.5 \times 10^{-5}$	0.2499	0	0.5000	0.0030	0.2430	0.0039	0.0065
		DTW	$7.3 \times 10^{-5}$	0.2506	$5.8 \times 10^{-5}$	0.4666	0.0889	0.0889	0.0038	0.0889
	Velocity	EN	0.0020	0.2032	0.0682	0.4679	0.02410	0.2345	0.0001	0.0682
DTW		0.0191	0.1000	0.1842	0.4020	0.1440	0.1480	0.0026	0.1841	
L4	Acceleration	EN	0	0.0003	0.4997	0.0002	0.4978	0.0018	$1.4 \times 10^{-8}$	0.0021
		DTW	$2.3 \times 10^{-10}$	$2.9 \times 10^{-9}$	0.5	$1.1 \times 10^{10}$	0.5	0	$1.8 \times 10^{-9}$	$2.9 \times 10^{-9}$
	Velocity	EN	$6.8 \times 10^{-8}$	0	0.5002	0.0018	0.4836	0.0139	0.0003	0.0163
DTW		0.0019	0.0852	0.3910	0.0471	0.3922	0.0825	$1.2 \times 10^5$	0.1090	
L5	Acceleration	EN	0.1253	0.2494	0.1254	0	0.1246	0.2505	0.1248	0.0006
		DTW	0.1250	0.2438	0.1348	0	0.1148	0.2565	0.12	0.0101
	Velocity	EN	0.1244	0.2528	0.1218	$8.6 \times 10^{-5}$	0.1290	0.2465	0.1256	0.0039
DTW		0.1414	0.2159	0.1432	0.0026	0.1303	0.2264	0.1402	0.0341	
L6	Acceleration	EN	0.1254	0.1240	0.1258	0.2498	0.1251	0.1248	0.1251	0.0010
		DTW	0.1250	0.1251	0.1249	0.2451	0.138	0.1169	0.125	0.0130
	Velocity	EN	0.1237	0.1297	0.1205	0.2474	0.1392	0.1113	0.1283	0.0141
DTW		0.1460	0.0709	0.1636	0.2511	0.0871	0.1747	0.1065	0.0540	
L7	Acceleration	EN	0.0628	0.1240	0.0633	0.4998	0.0625	0.1246	0.0628	0.0009
		DTW	0.0625	0.1249	0.0625	0.4952	0.0753	0.1169	0.0625	0.0128
	Velocity	EN	0.0669	0.1008	0.0904	0.4870	0.0783	0.1115	0.0648	0.0278
DTW		0.0619	0.1305	0.0556	0.5127	0.0176	0.1743	0.0478	0.0493	
L8	Acceleration	EN	0.2496	0.0010	0.2492	0.0005	0.2479	0.0029	0.2488	0.0029
		DTW	0.2486	0.0102	0.2364	0.0093	0.2387	0.0076	0.2492	0.0136
	Velocity	EN	0.2409	0.0335	0.2157	0.0194	0.2165	0.0335	0.2404	0.0342
DTW		0.2169	0.0684	0.2125	0.0139	0.1829	0.0893	0.2159	0.0893	
L9	Acceleration	EN	0.0497	0.2008	0.0493	0	0.0508	0.5987	0.0505	0.0012
		DTW	0.0500	0.2136	0.0246	0	0.1099	0.5460	0.0559	0.0598
	Velocity	EN	0.0495	0.2322	0.0002	$1.8 \times 10^{-8}$	0.1075	0.5570	0.0534	0.0575
DTW		0.0498	0.2273	0.0001	$8.8 \times 10^{-8}$	0.1666	0.4930	0.0630	0.1166	

For one of the previous laws of motion (L7), further tests were made with and without noise, carrying out the identification based on displacement, speed and acceleration. Table 5 shows the final errors on displacement, speed and acceleration calculated with the EN and DTW algorithm.

**Table 5.** Final values of the Euclidean norm and DTW algorithm on displacement, velocity and acceleration, in the simulated cases with law L7 as the original law of motion, with and without noise, and identification based on displacement, velocity and acceleration.

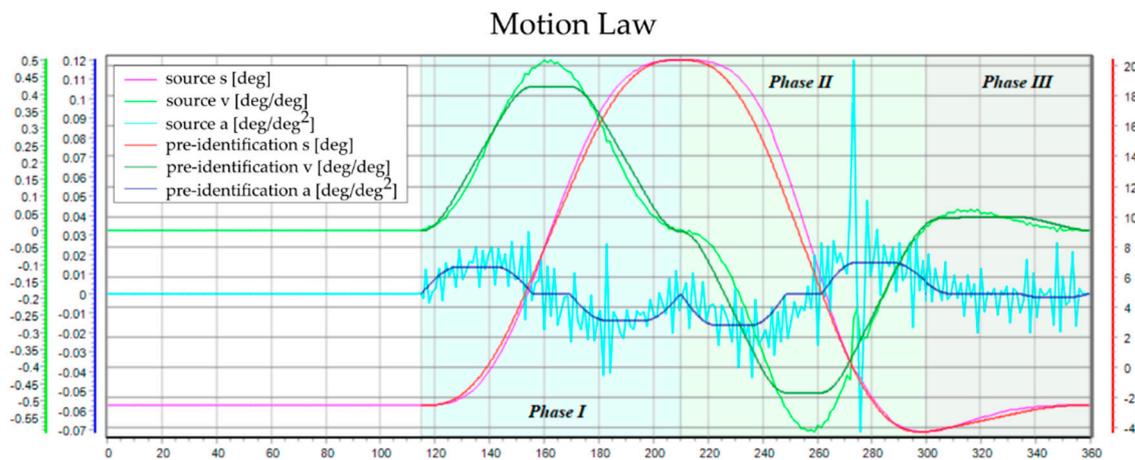
Quantity	Method	Error	Noise	Err_s [Rad]	Err_v [Rad/s]	Err_a [Rad/s <sup>2</sup> ]
Displacement	EN	EN	No	0.0139	0.2542	7.5341
		DTW	No	0.1036	1.2663	63.030
	DTW	EN	No	0.0258	0.4078	12.022
		DTW	No	0.0726	2.4829	106.91
Displacement	EN	EN	Yes	0.0065	0.1816	21.771
		DTW	Yes	0.0710	1.1354	294.62
	DTW	EN	Yes	0.0291	0.4264	23.191
		DTW	Yes	0.0777	2.6604	330.20
Velocity	EN	EN	No	0.0025	0.0054	2.8170
		DTW	No	0.0349	0.0562	16.730
	DTW	EN	No	0.0073	0.1282	5.5620
		DTW	No	0.0548	0.3884	31.030
Velocity	EN	EN	Yes	0.0045	0.0666	20.530
		DTW	Yes	0.0668	0.6859	284.00
	DTW	EN	Yes	0.0119	0.1505	22.270
		DTW	Yes	0.1272	0.7712	292.20
Acceleration	EN	EN	No	0.0004	0.0036	0.1068
		DTW	No	0.0060	0.0477	1.0890
	DTW	EN	No	0.0026	0.0254	1.2838
		DTW	No	0.0320	0.1572	2.2197
Acceleration	EN	EN	Yes	0.0163	0.0853	21.103
		DTW	Yes	0.2278	1.0417	291.66
	DTW	EN	Yes	0.0400	0.2240	20.877
		DTW	Yes	0.2184	2.1890	283.13

Subsequently, tests were carried out with original motion laws that cannot be represented by modified trapezoidal laws.

Three polynomial laws, with degrees of three, four and five, respectively, were considered. Six cases for each law were investigated: identification on acceleration, on velocity and on displacement, and for each one adopting either the EN or DTW algorithm. Table 6 collects the values of the EN and DTW algorithm on displacement, velocity and acceleration evaluated at the starting step of the identification, with a trapezoidal law characterized by seven coefficients equal to 1/7 (black curve in Figure 10). Table 7 contains the results of the identifications, in terms of the final values of the EN and DTW algorithm on displacement, velocity and acceleration.

**Table 6.** Initial values of the Euclidean norm and DTW algorithm on displacement, velocity and acceleration, in the simulated cases with laws of motion of the polynomial class used as the original law of motion.

Error	Polynomial 3-3			Polynomial 4-4			Polynomial 5-5		
	Err_s [Rad]	Err_v [Rad/s]	Err_a [Rad/s <sup>2</sup> ]	Err_s [Rad]	Err_v [Rad/s]	Err_a [Rad/s <sup>2</sup> ]	Err_s [Rad]	Err_v [Rad/s]	Err_a [Rad/s <sup>2</sup> ]
EN	0.0828	0.7645	11.91	0.4886	3.646	35.32	0.784	6.011	58.23
DTW	0.346	6.609	125.7	6.098	48.69	533.026	10.19	82.64	875.06



**Figure 10.** Comparison between the experimental law of motion (light blue) and modified trapezoidal laws for each phase (with equal coefficients values) as the starting point of the identification procedure (red: displacement; green: velocity; blue: acceleration).

**Table 7.** Final values of the Euclidean norm and DTW algorithm on displacement, velocity and acceleration, in the simulated cases with laws of motion of the polynomial class used as the original law of motion.

Kinematic Quantity	Method	Error	Polynomial 3-3			Polynomial 4-4			Polynomial 5-5		
			Err_s [Rad]	Err_v [Rad/s]	Err_a [Rad/s²]	Err_s [Rad]	Err_v [Rad/s]	Err_a [Rad/s²]	Err_s [Rad]	Err_v [Rad/s]	Err_a [Rad/s²]
Disp.	EN	EN	0.010	0.273	7.619	0.0567	0.815	22.040	0.191	2.060	45.940
		DTW	0.142	2.288	84.20	0.522	8.242	269.5	2.020	25.800	560.300
	DTW	EN	0.062	0.632	10.71	0.075	1.015	19.250	0.194	2.077	46.230
		DTW	0.116	5.188	107.9	0.498	12.24	236.600	2.020	25.90	563.900
Velocity	EN	EN	0.004	0.086	2.642	0.747	0.682	13.220	0.225	1.817	29.700
		DTW	0.059	1.057	25.90	0.680	6.645	171.800	2.438	22.590	412.500
	DTW	EN	0.013	0.191	4.020	0.075	0.699	11.330	0.216	1.844	30.520
		DTW	0.132	0.787	45.42	0.649	6.422	149.300	2.290	22.200	329.900
Accel.	EN	EN	0.007	0.064	1.388	0.112	0.808	9.423	0.288	2.137	22.680
		DTW	0.109	0.650	11.81	1.026	8.956	112.900	3.250	28.810	302.500
	DTW	EN	0.015	0.104	1.540	0.114	0.822	9.217	0.295	2.187	29.410
		DTW	0.195	0.770	8.780	1.051	9.217	111.300	3.330	22.750	298.600

### 3.2. Tests on Experimental Data

For the validation of the system, data gathered from the real case study of Figure 4 were used. For this system, composed of a damaged rocker arm follower cam, no construction drawings or characteristics of the laws of motion were available; still, displacement data in terms of follower rotation could be directly acquired on the real mechanism as a function of the cam rotation angle. The goal is to reconstruct the global motion law in order to be able to redesign the cam. The movement starts from the cam angle 110° and is characterized by the three phases—“Phase I”, “Phase II” and “Phase III”—highlighted in Figure 4 with different background colors. The constraints of the movement are reported in Table 8. Note that, in this case, the law of motion is expressed in terms of geometric law of motion, and this explains the measurement units in Table 8 and in the following tables. Since the speed of the cam is constant, the angular displacement of the cam and time are proportional, so the displacement of the follower (in deg) can be expressed as a function of the displacement of the cam (in deg) and not of time; accordingly, the units of follower displacement, speed and acceleration are deg, deg/deg and deg/deg², respectively. The procedure was applied to each one of the three phases independently. A comparison between the Euclidean norm and DTW algorithm to calculate the objective function in the identification based on displacement, velocity and acceleration was performed and the report of the results, in terms of final values of the EN and DTW algorithm, is in Table 9.

**Table 8.** Constraints' values for each phase of the considered actual case.

Phase Name	Constraint Vector	$\alpha_A$ [Deg]	$a_B$ [Deg]	$s_A$ [Deg]	$s_B$ [Deg]	$v_A$ [Deg/Deg]	$v_B$ [Deg/Deg]	$a_A$ [Deg/Deg <sup>2</sup> ]	$a_B$ [Deg/Deg <sup>2</sup> ]
Phase I	$\lambda_I$	115.0	210.0	−2.5000	20.3800	0.0	0.0	0.0	0.0
Phase II	$\lambda_{II}$	210.0	300.0	20.3800	−4.2700	−0.0028	0.0183	0.0	0.0060
Phase III	$\lambda_{III}$	300.0	360.0	−4.2700	−2.500	0.0	0.0	0.0060	0.0

**Table 9.** Final values of the Euclidean norm and DTW algorithm on displacement, velocity and acceleration, in the actual cases after an identification based on velocity.

Quantity	Method	Error	Phase I			Phase II			Phase III		
			Err_s [Deg]	Err_v [Deg/ Deg]	Err_a [Deg/ Deg <sup>2</sup> ]	Err_s [Deg]	Err_v [Deg/ Deg]	Err_a [Deg/ Deg <sup>2</sup> ]	Err_s [Deg]	Err_v [Deg/ Deg]	Err_a [Deg/ Deg <sup>2</sup> ]
Displacement	None	EN	7.083	0.590	0.187	17.600	1.090	1.370	2.868	0.221	0.169
		DTW	77.500	7.393	2.551	193.300	14.020	7.310	36.060	3.186	2.080
	EN	EN	0.364	0.130	0.178	1.977	0.545	1.390	0.104	0.057	0.167
		DTW	4.662	1.181	2.446	18.270	3.962	7.076	0.764	0.683	2.007
	DTW	EN	0.490	0.127	0.178	1.811	0.599	1.392	0.120	0.062	0.172
		DTW	3.250	1.111	2.450	11.060	5.137	7.171	0.760	0.757	2.148
Velocity	EN	EN	1.587	0.080	0.176	4.220	0.512	1.389	0.219	0.052	0.167
		DTW	3.601	0.734	2.420	40.020	3.399	7.017	1.404	0.626	2.085
	DTW	EN	2.132	0.070	0.176	2.870	0.520	1.389	0.210	0.052	0.167
		DTW	7.835	0.592	2.42	24.840	3.130	6.980	1.276	0.621	2.000
Accel.	EN	EN	1.165	0.076	0.176	4.059	0.520	1.389	1.068	0.077	0.166
		DTW	8.189	0.672	2.414	38.030	3.458	6.984	13.410	0.959	2.077
	DTW	EN	2.404	0.122	0.177	2.779	0.592	1.389	1.657	0.147	0.167
		DTW	13.093	1.084	2.406	21.195	4.458	6.950	20.380	2.009	2.039

Figure 10 shows the comparison between experimental law of motion (light blue) and modified trapezoidal laws for each phase (with equal coefficients values) when used as starting point of the identification procedure. Figure 11 reports the comparison with the identified laws based on displacement and EN, whereas Figure 12 reports on displacement and the DTW algorithm. Similarly, Figures 13 and 14 refer to identification based on speed, whereas Figures 15 and 16 refer to acceleration.

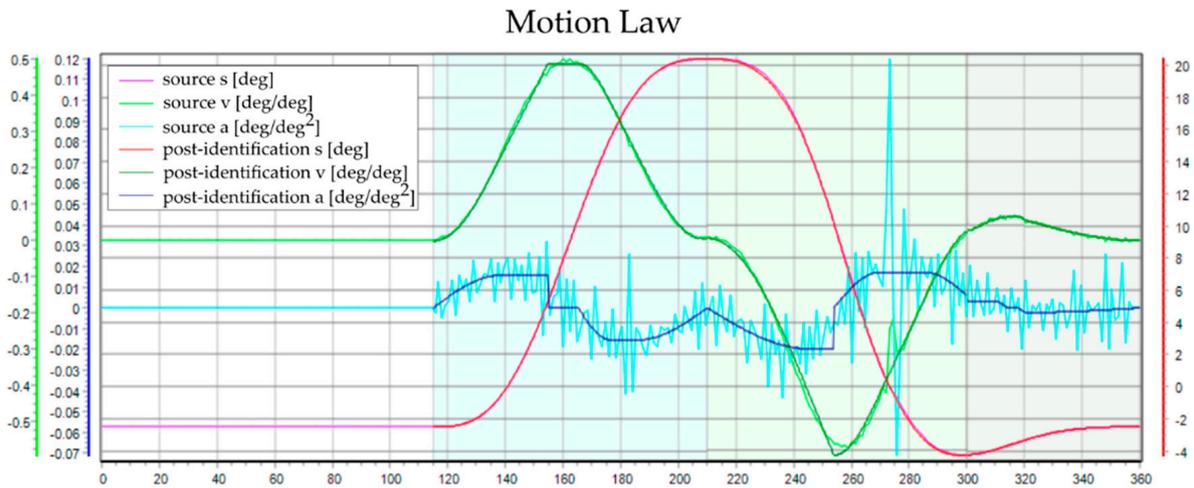


Figure 11. Comparison between the experimental law of motion (light blue) and modified trapezoidal laws for each phase (with equal coefficients values) after identification based on displacement by adopting EN.

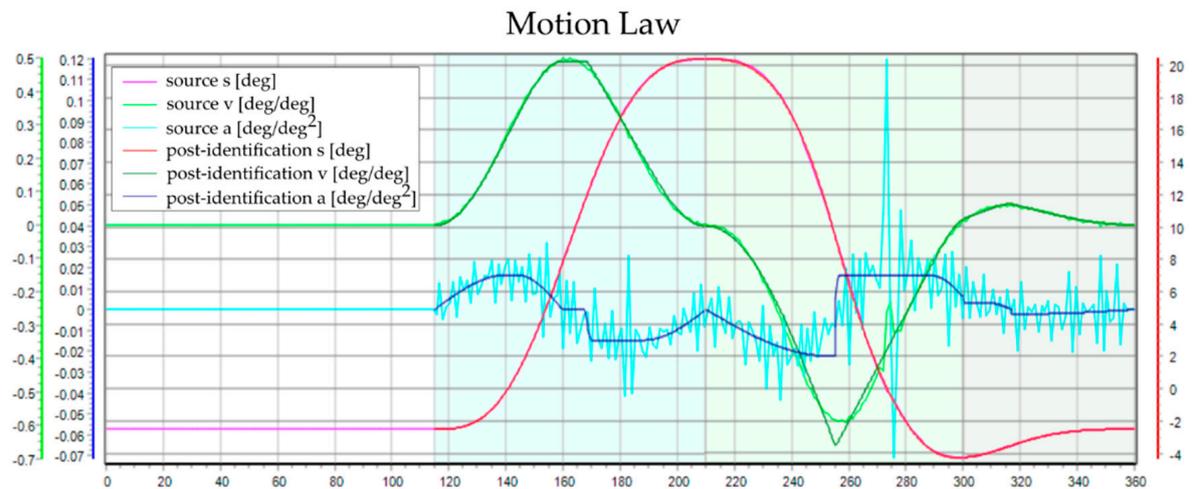


Figure 12. Comparison between the experimental law of motion (light blue) and modified trapezoidal laws for each phase (with equal coefficients values) after identification based on displacement by adopting the DTW algorithm.

The values of the seven coefficients of the best laws identified for the three areas are shown in Table 10.

Table 10. Trapezoidal modified law identified coefficients (adimensional values) for each phase of the considered actual case.

Phase Name	Coefficient Vector	$c_1$	$c_2$	$c_3$	$c_4$	$c_5$	$c_6$	$c_7$
Phase I	$c_I$	0.2386	0.0566	0.1867	0.0126	0.2121	0.0	0.2933
Phase II	$c_{II}$	0.3917	0.0500	0.1195	0.0235	0.1199	0.1389	0.2065
Phase III	$c_{III}$	0.0027	0.0620	0.1691	0.0890	0.0028	0.0002	0.6740

The computation time in all the examined cases keeps within the order of a few seconds.

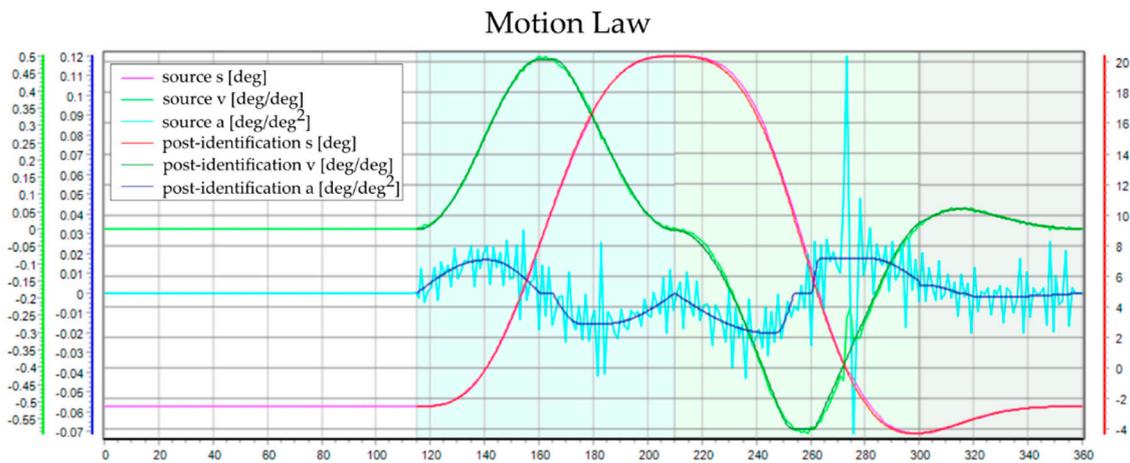


Figure 13. Comparison between the experimental law of motion (light blue) and modified trapezoidal laws for each phase (with equal coefficients values) after identification based on velocity by adopting the EN.

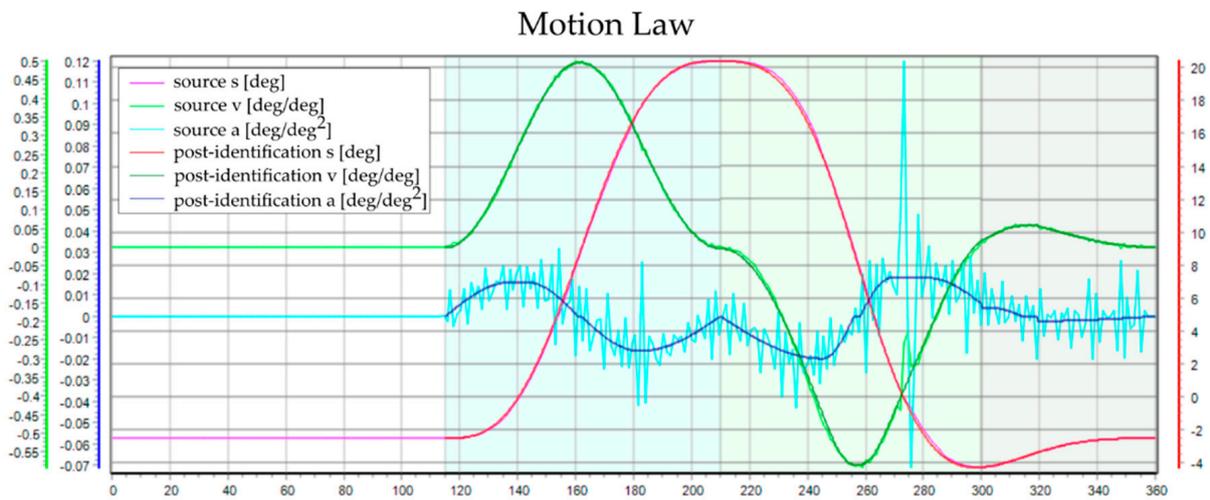


Figure 14. Comparison between the experimental law of motion (light blue) and modified trapezoidal laws for each phase (with equal coefficients values) after identification based on velocity by adopting the DTW algorithm.

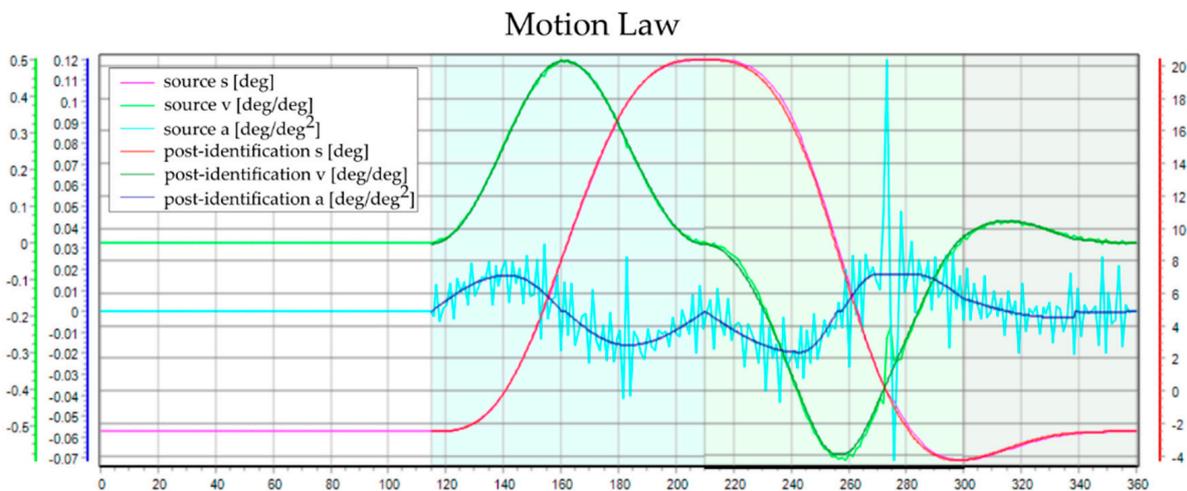
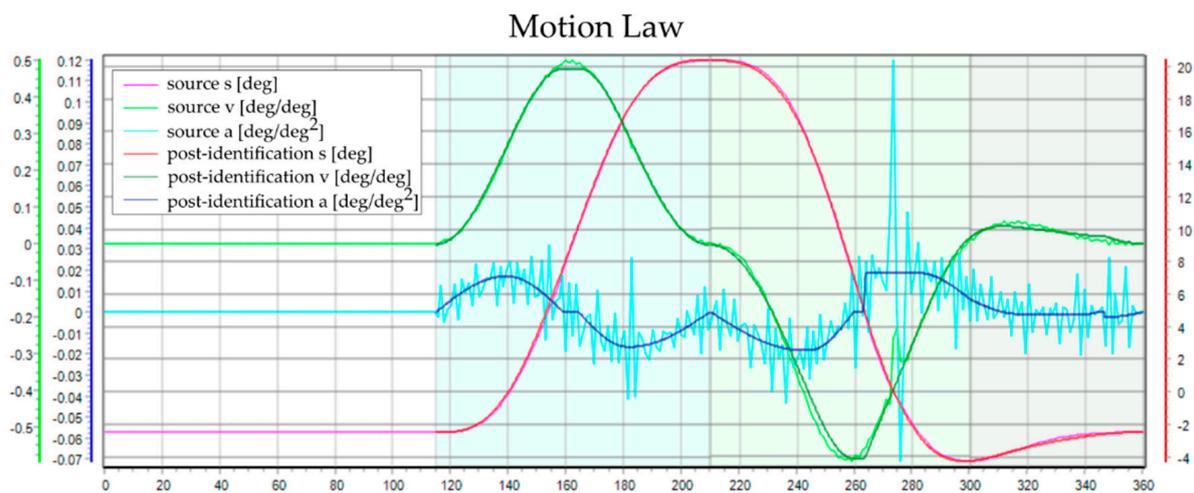


Figure 15. Comparison between the experimental law of motion (light blue) and modified trapezoidal laws for each phase (with equal coefficients values) after identification based on acceleration by adopting the EN.



**Figure 16.** Comparison between the experimental law of motion (light blue) and modified trapezoidal laws for each phase (with equal coefficients values) after identification based on acceleration by adopting the DTW algorithm.

#### 4. Discussion

Through the performed tests, the suitability of the method for the current purpose was proven (Figures 8 and 9), even in the presence of noise, and an effective set of GA intrinsic parameters was identified. In the study, for both simulation-based and real-data-based tests, data filtering prior to the application of the presented method was not considered. In this specific problem, filtering the acquired data could cause important components of the signal to be lost. Therefore, it was preferred to investigate the robustness of the method even in presence of disturbances.

Analyzing Table 4, the simulation tests on nine original motion laws of the trapezoidal class revealed that the identification based on acceleration achieved higher performances than with velocity in all the considered cases. For the objective function calculation, the DTW algorithm provided better results for law L1, L2 and L4, whereas the EN provided better results for the others. However, the errors on the coefficients obtained with identification based on acceleration, with both the EN and DTW algorithm, were present in all cases very small maximum values, so it can be said that the identification adequately reached the goal in both cases. The combined use of speed and DTW is characterized, in all the cases, by errors on the significant coefficients and, therefore, is a combination to be discarded.

Considering the errors in terms of displacement, velocity and acceleration for law L7 (Table 5), in the case without noise, a progressive improvement can be detected passing from the identification based on the displacement to that with speed and that with acceleration, with both the EN and DTW algorithm, and in all cases the errors are lower with the EN than with the DTW algorithm. In the case with noise, the error on acceleration substantially does not change identifying the EN on displacement, speed or acceleration, whereas there is a slight improvement with the DTW algorithm, and the obtained error is lower with acceleration. The lowest errors are obtained for displacement and speed when the identification is performed on speed with the EN, and for acceleration when the fitting is on acceleration with the DTW algorithm.

According to the tests with original laws of the polynomial class, the EN performs better than the DTW algorithm, almost in all the tested situations. Analyzing Table 7 by rows, the performance worsens as the degree of the polynomial increases. In particular, in terms of percentage of variation error, the worsening gets higher when passing from degree three to degree four than when passing from degree four to five. Analyzing data by columns, an overall improvement can be seen passing from identification on displacement to identification on speed and subsequently on acceleration, as observed for the law L7.

The percentage of error reduction, also in this case, is more significant for the 3-3 laws and progressively lessens as the degree increases.

Considering all three errors (on displacement, velocity and acceleration), for all the three considered laws of motion, the best case is the one with identification based on acceleration and with the EN.

For the actual case, the comparison between the identification with displacement, velocity and acceleration based on numerical errors (Table 9) shows that for all three phases the errors on the acceleration are substantially equivalent to the variation of the quantity to be identified and both with the EN and DTW algorithm. For the error on the displacement, in Phase I the best case is the identification on the displacement with the EN, in Phase II on the displacement with the DTW algorithm and in Phase 3 on displacement with the EN. In all three cases, however, the difference between the EN and DTW algorithm is low. Phase II is characterized by an anomalous behavior due to the breaking of the cam; therefore, as was hypothesized, the DTW algorithm may be more effective than the Euclidean norm in situations where anomalies are present. It is observed, however, that the numerical results of the errors do not provide information on the shape of the acceleration, which significantly affects the dynamic behavior of the system. A smoother law of motion for the acceleration, i.e., which presents a trend with a slope (Jerk) that varies more gradually, is always preferable, because it improves the dynamic behavior when subjected to high loads and high speeds [14,15]. Therefore, a visual comparison of trends is also useful. From the visual comparison of acceleration trends (Figures 11–16), in Phases I and II, the identifications based on speed with the DTW algorithm and acceleration with the EN are better and substantially equivalent, and in Phase III the identifications based on speed with the EN are the best ones.

The flexibility offered by manipulating the coefficients  $c$ , while keeping fixed the values at the extremes of the range of motion, is a key element of the presented reverse engineering technique. It is based, in fact, on an optimized reconstruction of the cam profile, by manipulating these coefficients.

A further significant advantage of the proposed method is that, based on the identification of the laws of motion, it offers the designer the possibility of verifying the limit values of the kinematic quantities (speed and acceleration), which significantly influence some design aspects (such as the overall dimensions for speed and dynamic loads for acceleration). In this way, when these values are excessive, the designer has the possibility to apply a well-known optimization method to improve the motion law from this point of view.

## 5. Conclusions

The developed procedure, which can be used as a phase of a wider reverse engineering process of cams, proved to be effective and with excellent performance in computational terms. Computational times of the method are well tolerable if the procedure is inserted, as tools, in a mechanical design software.

In all the examined cases, the procedure achieved good results. Using a GA as optimization tool and the EN or DTW algorithm for the evaluation of the objective function, it allows for the analytic reconstruction of a modified seven trapezoidal motion law that best approximates the motion law of the original cam.

The performed tests revealed that in many cases assessing the distance with the Euclidean norm on acceleration diagrams remains more effective, in particular when the signal to be recognized is little or not at all disturbed. Instead, the use of the DTW algorithm is more effective in case of anomalous situations, for instance in the acquired law of motion. In general, the DTW algorithm applied to velocity diagrams gives poor results, and therefore it is a combination that can generally be discarded. In general, there is no configuration that performs better than the others in all cases, but comparing the different conditions, an optimal configuration can be identified for each specific application,

considering, for instance, the numerical results, the visual observation of the trend or the speed of execution of the algorithm.

A limitation of the procedure is that it considers only one type of law of motion, the trapezoidal modified law. While the tests with original motion of the trapezoidal family showed excellent results, performance was much worse with motion laws of the polynomial family, as it was reasonable to expect. Therefore, a future development of the study is the extension of the method by integrating the use of other types of motion laws, such as polynomials, with the aim of identifying, among the different types of motion laws, the one that allows for the best representation of the movement to be recognized. A further development is to investigate the effectiveness of a multi-objective optimization, for which genetic algorithms are particularly suitable. For example, the passage for a precision point could be imposed.

Finally, although this methodology was applied to cam mechanisms in the present work, it has a more general validity and can be applied to any system that generates a movement. It could, therefore, be defined more generally as a reverse engineering methodology of laws of motion.

**Author Contributions:** Conceptualization, M.T., R.B. and C.A.; methodology, M.T., R.B. and C.A.; software, R.B.; validation M.T. and R.B.; formal analysis, M.T., C.A. and R.B.; data curation, R.B., M.T. and C.A.; writing—original draft preparation, M.T., C.A. and R.B.; writing—review and editing, M.T., C.A. and R.B. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

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

### Appendix A. Trapezoidal Modified Law of Motion

**Table A1.** Expressions of the vectors of cumulative partial displacements  $s_c$  and cumulative partial velocities  $v_c$  at the end of each section for a trapezoidal modified law of motion in the most general case.

Title 1	Title 2
$a_{11}$	$\frac{2c_0}{\pi} + c_1 + \frac{2c_2}{\pi}$
$a_{12}$	$-\left(\frac{2c_4}{\pi} + c_5 + \frac{2c_6}{\pi}\right)$
$a_{21}$	$\frac{2c_0}{\pi} \left(c_0 - \frac{2c_0}{\pi}\right) + \frac{c_1\pi^2}{2} + \left(\frac{2c_2}{\pi}\right)^2 + \frac{2c_0c_1}{\pi} + \left(\frac{2c_0}{\pi} + c_1\right)c_2 + \left(\frac{2c_0}{\pi} + c_1 + \frac{2c_2}{\pi}\right)(c_3 + c_4 + c_5 + c_6)$
$a_{22}$	$\frac{2c_6}{\pi} \left(c_6 - \frac{2c_6}{\pi}\right) + \frac{c_5\pi^2}{2} + \left(\frac{2c_4}{\pi}\right)^2 - \frac{2c_4\pi^2}{\pi} - \left(\frac{2c_4}{\pi} + c_5\right)c_5 - \left(\frac{2c_4}{\pi} + c_5 + \frac{2c_6}{\pi}\right)c_6$
$b_1$	$(v_B - v_A) + \left(\frac{2-\pi}{\pi}\right)(c_0a_A + c_6a_B)$
$b_2$	$\Delta s - \left(\Delta t \cdot v_A + \left(-\frac{2}{\pi} + \frac{4}{\pi^2} + \frac{1}{2}\right)c_0^2 + \left(1 - \frac{2}{\pi}\right)(\Delta t - c_0)c_0\right)a_A + \left(\frac{1}{2} - \frac{4}{\pi^2}\right)c_6^2a_B$
$A_{pos}$	$\frac{b_1a_{22} - b_2a_{12}}{a_{11}a_{22} - a_{12}a_{21}}$
$A_{neg}$	$-\frac{a_{11}b_2 - a_{21}b_1}{a_{11}a_{22} - a_{12}a_{21}}$
$v_{c_0}$	$v_A + a_A \cdot c_0 + 2(A_{pos} - a_A) \cdot \frac{c_0}{\pi}$
$v_{c_1}$	$v_{c_0} + A_{pos} \cdot c_1$
$v_{c_2}$	$v_{c_1} + 2A_{pos} \cdot \frac{c_2}{\pi}$
$v_{c_3}$	$v_2$
$v_{c_4}$	$v_{c_3} + 2A_{neg} \cdot \frac{c_4}{\pi}$
$v_{c_5}$	$v_{c_4} + A_{neg} \cdot c_5$
$v_{c_6}$	$v_{c_5} + a_B \cdot c_6 + 2(A_{neg} - a_B) \cdot \frac{c_6}{\pi}$
$s_{c_0}$	$s_A + v_A \cdot c_0 + \left(\left(\frac{2}{\pi} - \frac{4}{\pi^2}\right)A_{pos} + \left(\frac{1}{2} - \frac{2}{\pi} + \frac{4}{\pi^2}\right)a_A\right) \cdot c_0^2$
$s_{c_1}$	$s_{c_0} + v_{c_0} \cdot c_1 + A_{pos} \cdot \frac{c_1^2}{2}$
$s_{c_2}$	$s_{c_1} + v_{c_1} \cdot c_2 + A_{pos} \cdot \frac{2c_2^2}{\pi}$

Table A1. Cont.

Title 1	Title 2
$s_{c3}$	$s_{c2} + v_{c2} \cdot c_3$
$s_{c4}$	$s_{c3} + v_{c3} \cdot c_4 + 2A_{neg} \cdot c_4^2 \cdot \frac{(1-\frac{2}{\pi})}{\pi}$
$s_{c5}$	$s_{c4} + v_{c4} \cdot c_5 + A_{neg} \cdot \frac{c_5^2}{2}$
$s_{c6}$	$s_{c5} + v_{c5} \cdot c_6 + a_f \cdot \frac{c_6^2}{2} + (A_{neg} - a_B) \cdot \frac{2c_6}{\pi^2}$

Table A2. Analytic expression of displacement, velocity and acceleration at time  $t$  of a trapezoidal modified law of motion in the most general case.

Kinematic Quantity	Interval	Expression
$s(t)$	$t_A \leq t < t_{c1}$	$s_A + v_A t_0 + \frac{a_A}{2} t_0^2 + (A_{pos} - a_A) \frac{2c_0}{\pi} \left( t_0 - \frac{2c_0}{\pi} \sin\left(\frac{t_0 \pi}{2c_0}\right) \right)$
	$t_{c1} \leq t < t_{c2}$	$s_{c0} + v_{c0} t_1 + \frac{A_{pos} t_1^2}{2}$
	$t_{c2} \leq t < t_{c3}$	$s_{c1} + v_{c1} t_2 + A_{pos} \left(\frac{2c_2}{\pi}\right)^2 \left(1 - \cos\left(\frac{t_2 \pi}{2c_2}\right)\right)$
	$t_{c3} \leq t < t_{c4}$	$s_{c2} + v_{c2} t_3$
	$t_{c4} \leq t < t_{c5}$	$s_{c3} + v_{c3} t_4 + A_{neg} \frac{2c_4}{\pi} \left( t_4 - \frac{2c_4}{\pi} \sin\left(\frac{t_4 \pi}{2c_4}\right) \right)$
	$t_{c5} \leq t < t_{c6}$	$s_{c4} + v_{c4} t_5 + \frac{A_{neg} t_5^2}{2}$
	$t_{c6} \leq t \leq t_B$	$s_{c5} + v_{c5} t_6 + \frac{a_B}{2} t_6^2 + (A_{neg} - a_A) \left(\frac{2c_6}{\pi}\right)^2 \left(1 - \cos\left(\frac{t_6 \pi}{2c_6}\right)\right)$
$v(t)$	$t_A \leq t < t_{c1}$	$v_A + a_A t_0 + (A_{pos} - a_A) \frac{2c_0}{\pi} \left(1 - \cos\left(\frac{t_0 \pi}{2c_0}\right)\right)$
	$t_{c1} \leq t < t_{c2}$	$v_{c0} + A_{pos} t_1$
	$t_{c2} \leq t < t_{c3}$	$v_{c1} + A_{pos} \frac{2c_2}{\pi} \sin\left(\frac{t_2 \pi}{2c_2}\right)$
	$t_{c3} \leq t < t_{c4}$	$v_{c2}$
	$t_{c4} \leq t < t_{c5}$	$v_{c3} + A_{neg} \frac{2c_4}{\pi} \left(1 - \cos\left(\frac{t_4 \pi}{2c_4}\right)\right)$
	$t_{c5} \leq t < t_{c6}$	$v_{c4} + A_{neg} t_5$
	$t_{c6} \leq t \leq t_B$	$v_{c5} + a_B t_6 + (A_{neg} - a_B) \frac{2c_6}{\pi} \sin\left(\frac{t_6 \pi}{2c_6}\right)$
$a(t)$	$t_A \leq t < t_{c1}$	$a_A + (A_{pos} - a_A) \sin\left(\frac{t_0 \pi}{2c_0}\right)$
	$t_{c1} \leq t < t_{c2}$	$A_{pos}$
	$t_{c2} \leq t < t_{c3}$	$A_{pos} \cos\left(\frac{t_2 \pi}{2c_2}\right)$
	$t_{c3} \leq t < t_{c4}$	$0$
	$t_{c4} \leq t < t_{c5}$	$A_{neg} \sin\left(\frac{t_4 \pi}{2c_4}\right)$
	$t_{c5} \leq t < t_{c6}$	$A_{neg}$
	$t_{c6} \leq t < t_B$	$a_B + (A_{neg} - a_B) \cos\left(\frac{t_6 \pi}{2c_6}\right)$

References

- Chen, F.Y. *Mechanics and Design of Cam Mechanisms*; Pergamon Press: Oxford, UK, 1982.
- Wilson, C.E.; Sadler, J.P.; Michels, W.J. *Kinematics and Dynamics of Machinery*; Pearson Edition: London, UK, 1983.
- Flores, P. A computational approach for cam size optimization of disc cam-follower mechanisms with translating roller followers. *J. Mech. Robot.* **2013**, *5*, 041010. [CrossRef]
- Abderazek, H.; Yildiz, A.R.; Mirjalili, S. Comparison of recent optimization algorithms for design optimization of a cam-follower mechanism. *Knowl. Based Syst.* **2020**, *191*, 105237. [CrossRef]
- Parouha, R.P.; Verma, P. Design and applications of an advanced hybrid meta-heuristic algorithm for optimization problems. *Artif. Intell. Rev.* **2021**, *54*, 5931–6010. [CrossRef]
- Lampinen, J. Cam shape optimisation by genetic algorithm. *CAD Comput. Aided Des.* **2003**, *35*, 727–737. [CrossRef]
- Tsiafis, I.; Mitsi, S.; Bouzakis, K.D.; Papadimitriou, A. Optimal design of a cam mechanism with translating flat-face follower using genetic algorithm. *Tribol. Ind.* **2013**, *35*, 255–260.

8. Bhattacharjee, P.; Jana, R.K. A multi-objective genetic algorithm for design optimisation of simple and double harmonic motion cams. *Int. J. Des. Eng.* **2017**, *7*, 77–91. [[CrossRef](#)]
9. Hamza, F.; Abderazek, H.; Lakhdar, S.; Ferhat, D.; Yıldız, A.R. Optimum design of cam-roller follower mechanism using a new evolutionary algorithm. *Int. J. Adv. Manuf. Technol.* **2018**, *99*, 1267–1282. [[CrossRef](#)]
10. Yu, J.; Huang, K.; Luo, H.; Wu, Y.; Long, X. Manipulate optimal high-order motion parameters to construct high-speed cam curve with optimized dynamic performance. *Appl. Math. Comput.* **2020**, *371*, 124953. [[CrossRef](#)]
11. Ouyang, T.; Wang, P.; Huang, H.; Zhang, N.; Chen, N. Mathematical modeling and optimization of cam mechanism in delivery system of an offset press. *Mech. Mach. Theory* **2017**, *110*, 100–114. [[CrossRef](#)]
12. Gatti, G.; Mundo, D. On the direct control of follower vibrations in cam-follower mechanisms. *Mech. Mach. Theory* **2010**, *45*, 23–35. [[CrossRef](#)]
13. Carbone, G.; Lanni, C.; Ceccarelli, M.; Incerti, G.; Tiboni, M. A characterization of cam transmissions through an identification of lumped parameters. *Proc. ASME Des. Eng. Tech. Conf.* **2006**, *2006*, 709–718. [[CrossRef](#)]
14. Tiboni, M.; Bussola, R.; Aggogeri, F.; Amici, C. Experimental and model-based study of the vibrations in the load cell response of automatic weight fillers. *Electronics* **2020**, *9*, 995. [[CrossRef](#)]
15. Tiboni, M.; Roberto, B.; Carlo, R.; Faglia, R.; Adamini, R.; Amici, C. Study of the Vibrations in a Rotary Weight Filling Machine. In Proceedings of the 2019 23rd International Conference on Mechatronics Technology, ICMT 2019, Salerno, Italy, 23–26 October 2019; pp. 1–6. [[CrossRef](#)]
16. Dalpiaz, G.; Rivola, A. Condition monitoring and diagnostics in automatic machines: Comparison of vibration analysis techniques. *Mech. Syst. Signal Process.* **1997**, *11*, 53–73. [[CrossRef](#)]
17. Tiboni, M.; Remino, C. Condition monitoring of a mechanical indexing system with artificial neural networks. In Proceedings of the WCCM 2017—1st World Congress on Condition Monitoring 2017, London, UK, 13–16 June 2017; Springer International Publishing: Berlin, Germany, 2017.
18. Tiboni, M.; Incerti, G.; Remino, C.; Lancini, M. Comparison of signal processing techniques for condition monitoring based on artificial neural networks. *Appl. Cond. Monit.* **2019**, *15*, 179–188. [[CrossRef](#)]
19. Buonamici, F.; Carfagni, M.; Furferi, R.; Governi, L.; Lapini, A.; Volpe, Y. Reverse engineering modeling methods and tools: A survey. *Comput. Aided. Des. Appl.* **2018**, *15*, 443–464. [[CrossRef](#)]
20. Fu, P. *Reverse Engineering in the Medical Device Industry*; Springer International Publishing: Berlin, Germany, 2008; pp. 177–194, ISBN 978-1-84628-855-5.
21. Balamurugan, P.; Selvakumar, N. Development of patient specific dental implant using 3D printing. *J. Ambient Intell. Humaniz. Comput.* **2021**, *12*, 3549–3558. [[CrossRef](#)]
22. Chaudhary, K.; Govil, A. *Application of 3D Scanning for Reverse Manufacturing and Inspection of Mechanical Components BT—Proceedings of the International Conference on Industrial and Manufacturing Systems (CIMS-2020)*; Pratap Singh, R., Tyagi, D.M., Panchal, D., Davim, J.P., Eds.; Springer International Publishing: Cham, Switzerland, 2020; pp. 61–76.
23. Li, D. Application of Coordinate Measuring Machine in Reverse Engineering. *Adv. Mater. Res.* **2011**, *301–303*, 269–274. [[CrossRef](#)]
24. Zhenghao, G.; Kaikai, Z.; Meng, J.; Fulian, Y. An Exact Reverse Design Approach for Disk Cam Mechanisms. *Adv. Syst. Sci. Appl.* **2011**, *11*, 301–308.
25. Alsoufi, M.S.; Alhazmi, M.W.; Azam, S.; Modeling, C.A.D. Mechanical reverse engineering approach for precise measurements of reproducing disc CAM. *Int. J. Eng. Technol.* **2018**, *7*, 5741–5757. [[CrossRef](#)]
26. Samsonenko, D.M.; Khudoyarov, V.A.; Barbashov, N.N.; Abdullina, L.R.; Potapov, D.M. Research of influence of errors of cam manufacturing on the law of motion of output element. *IOP Conf. Ser. Mater. Sci. Eng.* **2020**, *862*, 032053. [[CrossRef](#)]
27. Goldberg, D.E. *Genetic Algorithms in Search, Optimization and Machine Learning*, 1st ed.; Addison-Wesley Longman Publishing Co., Inc.: Boston, MA, USA, 1989; ISBN 0201157675.
28. Ratanamahatana, C.A.; Keogh, E. Making time-series classification more accurate using learned constraints. In Proceedings of the 2004 SIAM International Conference on Data Mining, Lake Buena Vista, FL, USA, 22–24 April 2004; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA; pp. 11–22. [[CrossRef](#)]
29. Yadav, M.; Alam, M.A. Dynamic Time Warping (DTW) Algorithm In Speech: A Review. *Int. J. Res. Electron. Comput. Eng.* **2018**, *6*, 524–528.
30. Senin, P. Dynamic Time Warping Algorithm Review. *Science* **2008**, *2007*, 1–23.
31. Gameros, A.; De Chiffre, L.; Siller, H.R.; Hiller, J.; Genta, G. A reverse engineering methodology for nickel alloy turbine blades with internal features. *CIRP J. Manuf. Sci. Technol.* **2015**, *9*, 116–124. [[CrossRef](#)]
32. Wang, J.; Gu, D.; Yu, Z.; Tan, C.; Zhou, L. A framework for 3D model reconstruction in reverse engineering. *Comput. Ind. Eng.* **2012**, *63*, 1189–1200. [[CrossRef](#)]
33. Wang, J.; Gu, D.; Gao, Z.; Yu, Z.; Tan, C.; Zhou, L. Feature-based solid model reconstruction. *J. Comput. Inf. Sci. Eng.* **2013**, *13*, 1–13. [[CrossRef](#)]
34. Tsay, D.-M.; Huang, M.-H.; Ho, H.-C. Producing Follower Motions Through Their Digitized Cam Contours. *J. Comput. Inf. Sci. Eng.* **2002**, *2*, 98–105. [[CrossRef](#)]
35. Ge, Z.; Su, P.; Yang, F.; Jiang, M. The Reverse Design Method for Cam Contours and Motion Specification. In Proceedings of the International Technology and Innovation Conference 2009 (ITIC 2009), Xi'an, China, 12–14 October 2009; pp. 1–4. [[CrossRef](#)]