

Article

Design General Cam Profiles Based on Finite Element Method

Thi Thanh Nga Nguyen ^{1,*}, Thang Xuan Duong ² and Van-Sy Nguyen ¹

¹ Faculty of Mechanical Engineering, Thai Nguyen University of Technology (TNUT), Thai Nguyen City 250000, Vietnam; vansy@tnut.edu.vn

² Aachen Institute for Study in Computational Engineering Science (AICES), RWTH Aachen University, 52062 Aachen, Germany; duong@aices.rwth-aachen.de

* Correspondence: nguyennnga@tnut.edu.vn

Abstract: This paper presents a general framework to design a cam profile using the finite element method from given displacements of the follower. The arbitrarily complex cam profile is described by Lagrangian finite elements, which are formed by the connectivity of nodes. In order to obtain the desired profile, a penalty-type functional that enforces the prescribed displacement of the follower is proposed. Additionally, in order to ensure convexity of the functional, a numerical stabilization scheme is used. The nodal positions are then obtained by solving a nonlinear system of equations resulting from minimizing the total functional. The geometrical accuracy of the cam profile can be controlled by the number of finite elements. A case study is considered to illustrate the flexibility, accuracy, and robustness of the proposed approach.

Keywords: cam profile; cam design; finite element; curvilinear coordinate



Citation: Nguyen, T.T.N.; Duong, T.X.; Nguyen, V.-S. Design General Cam Profiles Based on Finite Element Method. *Appl. Sci.* **2021**, *11*, 6052. <https://doi.org/10.3390/app11136052>

Academic Editor: José A. F. O. Correia

Received: 22 May 2021

Accepted: 22 June 2021

Published: 29 June 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 mechanisms are widely used in various mechanical machines, such as machine tools, sewing machines, and engines due to their unlimited motions, operation speed, motion accuracy, and structural rigidity. In cam design, the follower motion affects the kinematic and dynamic characteristics of cam systems. Various traditional functions can be used to describe the follower motion such as harmonic, cycloidal, trapezoidal, polynomial, and piecewise polynomial [1,2]. So far, spline functions have also been used to develop the cam curves, for example, the Bezier curve [3], the B-spline curve [4–6], and the NURBS curve [7–10]. Besides, kinematics and dynamics have been considered to optimize the peak values of the displacement, velocity, acceleration, and jerk with referring to several functions, e.g., harmonic, cycloidal, and polynomial [11]. Qin and Chen [12] also proposed the kinematic optimization of cam motions for engine valve trains regarding the characteristics of polynomial cam curves. In order to achieve motion curves with good dynamic characteristics under arbitrary design conditions, the parameters of the motion model were obtained by optimizing the dynamic performance [13,14]. Related to the cam design process, the cam size optimization has been also investigated [15–17]. This plays an important role due to affect the cost of cam systems.

The design of the cam profile is one of the main tasks in designing a cam mechanism. The profile of the cam must be designed correctly to ensure the characteristics of kinematics and dynamics of cam mechanisms. However, when a priori displacement of the follower is given, obtaining the corresponding curve equations for cam profiles is nontrivial in general. Many approaches to design the profile of a cam have been proposed. The graphical method has been established by several studies for disk cams [1,18,19]. In this method, kinematic inversion is used. To increase the accuracy, the analytical method using a Cartesian coordinate system is presented by [20–22]. Since then, the analytical method has been continuously advanced, among others by Shigley and Jr. [1] and Norton [23]. In these studies, the equations of cam profile have been formulated by the loop-closure equation

and using complex polar notation. Wu et al. [24] presented a cam profile of a translating cam driven. In this approach, the follower motion can be arbitrarily chosen for only the forward or the return stroke. The desired cam profile is obtained by using the theory of envelopes [25]. To this end, the envelopes are given by fitting a tangent curve to the successive follower positions, and the cam profile is achieved by the boundary of the family. Biswas et al. [26] presented two approximate analytical methods for determining disk cam profiles of roller followers. In the first method, the average location is defined by two lines tangent of three roller positions. The cam profile is obtained by the interpolation of the roller positions. In the second method of this research, the cam profile is achieved by the intersection of the curvature center and the roller center. A systematic method for the design and analysis of cams with three circular-arc profiles is presented by Jung-Fa Hsieh [27]. This study is based on constructing a generic geometric model of cam profile and deriving the corresponding equations of the constituent circular arcs using a coordinate transformation. Moreover, using coordinate transformation, the cam profile is obtained by [28]. This study proposes a simple yet comprehensive method for the design and analysis of an indexing cam mechanism with parallel axes.

In this paper, we propose a finite element method to design a cam profile from a given displacement of the follower. To the best of our knowledge, this is the first time a finite element description is used directly for obtaining the cam profile. Compared to earlier works, the presented method contains the following merits and novelties:

- A penalty type functional to enforce the prescribed displacement of the follower is proposed;
- A stabilization scheme is added to guarantee convexity of the functional;
- For the Newton–Raphson solution method, the consistent linearization of the weak form is presented;
- The arbitrary desired accuracy of the cam profile can be achieved by increasing the number of finite elements;
- The resulting profile represented by finite elements can be used for cam manufacturing.

The remainder of the paper is structured as follows. In Section 2, the geometrical description of cam profiles is presented. Section 3 introduces the synthetical kinematics. The formulation of the general framework for the synthesis of cam mechanisms is presented in Section 4. The finite element (FE) formulation is discussed in Section 5. Section 6 then presents a solution of the finite element. The case study for this method is presented in Section 7. Finally, conclusions are drawn in Section 8.

2. Geometrical Description of Cam Profiles

In this section, we present the geometrical description of the cam profile. The differential geometry (see [29]) is used for an arbitrarily complex cam surface. The cam surface, denoted by S (see Figure 1), is represented in curvilinear coordinates as

$$\mathbf{x} = \mathbf{x}(\xi), \quad (1)$$

where $\xi = (\xi^1, \xi^2)$ are the curvilinear coordinates. Accordingly, the covariant and contravariant tangent vectors, \mathbf{a}_α and \mathbf{a}^α , at point \mathbf{x} depending on S are calculated by

$$\mathbf{a}_\alpha = \frac{\partial \mathbf{x}}{\partial \xi^\alpha}, \quad (2)$$

and

$$\mathbf{a}^\alpha := a^{\alpha\beta} \mathbf{a}_\beta, \quad (3)$$

respectively, where

$$\begin{aligned} a^{\alpha\beta} &:= [a_{\alpha\beta}]^{-1}, \\ a_{\alpha\beta} &:= \mathbf{a}_\alpha \cdot \mathbf{a}_\beta \end{aligned} \quad (4)$$

are the contravariant and covariant components of the metric tensor. Here and henceforth, the summation convention is implied for repeated Greek indices, e.g., $\alpha = 1, 2$. The unit normal vector \mathbf{n} at \mathbf{x} is defined by [30]

$$\mathbf{n} = \frac{\mathbf{a}_1 \times \mathbf{a}_2}{\|\mathbf{a}_1 \times \mathbf{a}_2\|} . \quad (5)$$

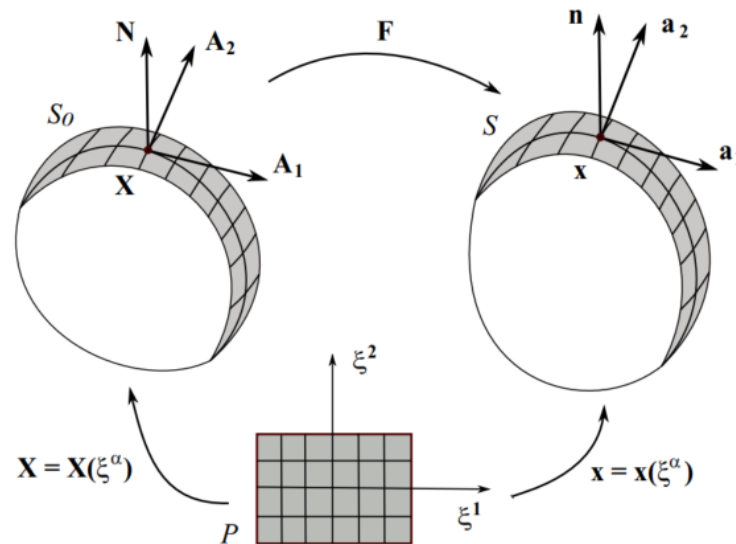


Figure 1. Mapping between parameter domain P , reference configuration S_0 , and current configuration S .

It can be shown that

$$\|\mathbf{a}_1 \times \mathbf{a}_2\| = \sqrt{\det a_{\alpha\beta}} . \quad (6)$$

The curvature tensor can be computed as (see, e.g., [29])

$$\mathbf{b} = b_{\alpha\beta} \mathbf{a}^\alpha \otimes \mathbf{a}^\beta , \quad (7)$$

where

$$b_{\alpha\beta} := \mathbf{n} \cdot \mathbf{a}_{\alpha,\beta} . \quad (8)$$

3. Synthetical Kinematics

In this section, the established approach to describe the deformation of membranes (see, e.g., [29]) is employed in the context of synthetical kinematics for designing cam profiles. To obtain the desired cam profile, a given initial configuration is selected. The initial curve is then “deformed” to the desired configuration. The differences between the initial (reference) configuration S_0 and the current configuration S are shown in Figure 1. Both configurations are described by the geometrical quantities discussed in Section 2. For the reference configuration S_0 we use the upper-case symbols \mathbf{X} , \mathbf{A}_α , $A_{\alpha\beta}$, \mathbf{A}^α , and \mathbf{N} . The corresponding lower-case symbols \mathbf{x} , \mathbf{a}_α , $a_{\alpha\beta}$, \mathbf{a}^α , and \mathbf{n} are used in the current configuration S . In order to characterize the deformation between the reference configuration S_0 and the current configuration S , a line element in S is considered as

$$d\mathbf{x} = \mathbf{a}_\alpha d\xi^\alpha , \quad (9)$$

and its corresponding line $d\mathbf{X} = \mathbf{A}_\alpha d\xi^\alpha$ in S_0 . So that we can write $d\xi^\alpha = \mathbf{A}^\alpha d\mathbf{X}$. With this, $d\mathbf{x}$ in the current configuration can be mapped to $d\mathbf{X}$ in S_0 by

$$d\mathbf{x} = (\mathbf{a}_\alpha \otimes \mathbf{A}^\alpha) d\mathbf{X} . \quad (10)$$

Here the tensor

$$\mathbf{F} = \mathbf{a}_\alpha \otimes \mathbf{A}^\alpha \quad (11)$$

denotes the deformation gradient. Vice versa

$$\mathbf{F}^{-1} = \mathbf{A}_\alpha \otimes \mathbf{a}^\alpha \quad (12)$$

maps the $d\mathbf{X}$ in S_0 to $d\mathbf{x}$ in S . Through \mathbf{F} we thus have the following transformation

$$\begin{aligned} \mathbf{a}_\alpha &= \mathbf{F} \mathbf{A}_\alpha, & \mathbf{A}_\alpha &= \mathbf{F}^{-1} \mathbf{a}_\alpha, \\ \mathbf{a}^\alpha &= \mathbf{F}^{-T} \mathbf{A}^\alpha, & \mathbf{A}^\alpha &= \mathbf{F}^T \mathbf{a}^\alpha. \end{aligned} \quad (13)$$

Finally, the stretch between configurations S_0 and S is defined by

$$\begin{aligned} J &:= \frac{J_a}{J_A}, \\ J_A &:= \sqrt{\det \mathbf{A}_{\alpha\beta}}, \\ J_a &:= \sqrt{\det \mathbf{a}_{\alpha\beta}}. \end{aligned} \quad (14)$$

Here, “det” denotes the surface determinant.

4. Governing Equation for the Synthesis of Cam Profile

In this section, the governing equation for designing cam profile is proposed. The strong form, weak form, and stabilization scheme are presented.

4.1. Strong form Statement

The point-wised governing equation for the synthesis of cam profile is given by

$$g := s - s_0 = 0, \forall \mathbf{x} \in S, \quad (15)$$

subjected to the boundary condition

$$\mathbf{x} = \bar{\mathbf{x}}, \forall \mathbf{x} \in \Gamma_u, \quad (16)$$

where s_0 and s denote the reference and current displacement of the follower, respectively. s_0 and s will be explicitly defined for a particular type of cam. Γ_u is a subset of S . From Equation (15), the variation δg can be expressed as

$$\delta g = \delta s - \delta s_0, \quad (17)$$

where

$$\delta s = \frac{\partial s}{\partial \mathbf{x}} \cdot \delta \mathbf{x}, \quad \delta s_0 = \frac{\partial s_0}{\partial \mathbf{x}} \cdot \delta \mathbf{x}. \quad (18)$$

The strong form equation as shown in Equations (15) and (16) is not a well-posed statement since it is still missing an in-plane constraint equation. In another word, the positions of two arbitrary material points $\mathbf{x}_1 \in S \setminus \Gamma_u$ and $\mathbf{x}_2 \in S \setminus \Gamma_u$ can be exchanged without affecting the solution of Equation (15). The system is thus unstable and needs stabilization, which will be discussed in Section 4.4.

4.2. Penalty-Type Functional

The point-wise condition in Equation (15) can be enforced by a penalty-type functional. The penalty-type functional can be expressed as

$$\Pi_P = \int_{S_0} \frac{\epsilon}{2} g^2 dA_0, \quad (19)$$

where ϵ is the penalty parameter. The variation of the penalty-type functional is computed as

$$\delta \Pi_p = \int_{S_0} \lambda \delta g \, dA_0, \quad (20)$$

where

$$\lambda := \epsilon g \quad (21)$$

represents the pressure of the enforcement.

It should be noted that the potential in Equation (19) exactly enforces the condition at the limit, i.e., $\epsilon \rightarrow \infty$. At $\epsilon \rightarrow 0$, Π_p does not affect the system. It turns out that this property is advantageous for the Newton solution method since it allows for the application of s in an increment manner until a desired accuracy is reached.

4.3. Augmented Lagrange Functional

In order to enforce the condition in Equation (15) stronger, the Augmented Lagrange method can be used instead of the Penalty method. Accordingly, the pressure λ (see Equation (21)) is further corrected by using extra loops called augmentation steps. For this, we consider the functional

$$\Pi_p = \int_{S_0} \left(\lambda g + \frac{\epsilon}{2} g^2 \right) dA_0, \quad (22)$$

where λ is a non-negative parameter. Thus, its variation reads:

$$\delta \Pi_p = \int_{S_0} (\lambda + \epsilon g) \delta g \, dA_0. \quad (23)$$

From Equation (23), one finds the updated new parameter λ_{new} for Equation (20) as

$$\lambda_{\text{new}} = \epsilon g + \lambda_{\text{old}}. \quad (24)$$

4.4. Stabilization Governing Functional

As noted above, the system is unstable and thus cannot be solved uniquely with FEM. In this paper, a stabilization method is considered as found in [31]. The stabilization functional can be expressed by

$$\Pi_{\text{stab}} := \int_{S_0} \frac{\mu}{2} (I_1 - 2 \ln J) dA_0; \quad I_1 := A^{\alpha\beta} a_{\alpha\beta}, \quad (25)$$

where μ is a given parameter. The variation of Π_{stab} can be found as

$$\delta \Pi_{\text{stab}} := \frac{\mu}{2} \int_{S_0} \tau^{\alpha\beta} \delta a_{\alpha\beta} \, dA_0, \quad (26)$$

where, $\tau^{\alpha\beta} = A^{\alpha\beta} - a^{\alpha\beta}$.

4.5. Stabilized weak Form Statement

The cam surface is assumed to be governed by the weak form

$$\Pi := \Pi_p + \Pi_{\text{stab}}, \quad (27)$$

where Π_p and Π_{stab} are defined in Equations (19) or (22), and (25), respectively. The cam profile is obtained by minimizing potential (27). I.e.,

$$\delta \Pi := \delta \Pi_p + \delta \Pi_{\text{stab}} = 0. \quad (28)$$

Here, $\delta \Pi_p$ and $\delta \Pi_{\text{stab}}$ are defined in Equations (20) or (23), and (26), respectively.

4.6. Linearized Weak Form

The linearization is necessary for the calculation of tangent matrices. The linearization of the potential energy as shown in Equation (20) can be computed by

$$\Delta \delta \Pi_p = \int_{S_0} \epsilon (\Delta g \delta g + g \Delta \delta g) dA_0. \quad (29)$$

Here $\Delta \delta g$ can be expressed in the form

$$\Delta \delta g = \Delta \delta s - \Delta \delta s_0. \quad (30)$$

4.7. Linearized Stabilization Term

From Equation (26), the linearization of the stabilization equation can be computed by

$$\Delta \delta \Pi_{\text{stab}} = \mu \int_{S_0} \left(-\Delta a^{\alpha\beta} \delta a_{\alpha\beta} + \tau^{\alpha\beta} \Delta \delta a_{\alpha\beta} \right) dA_0. \quad (31)$$

5. Finite Element Discretization

Equation (28) is solved by the finite element method. The surface S is therefore discretized into a set of finite elements Ω^e that are defined by nodal points x_i which corresponds to the Lagrangian finite element description (see Figure 2).

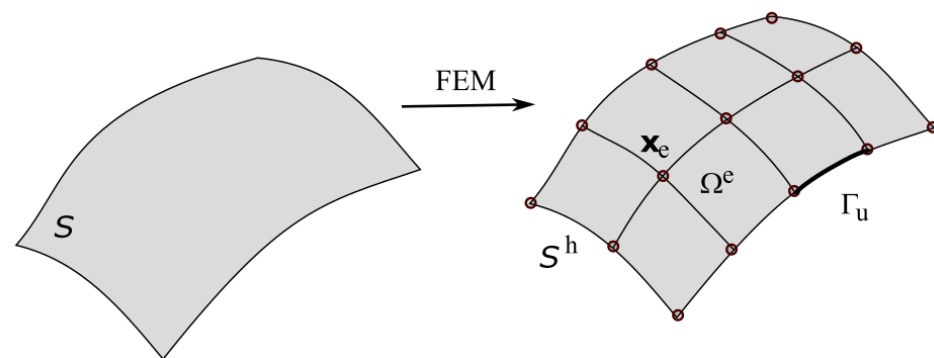


Figure 2. FE discretization of surface S .

5.1. Discretized Synthetical Kinematic Quantities

Within elements Ω^e , the geometry is approximated by the nodal interpolations

$$\mathbf{x} \approx \mathbf{x}^h = \sum_{i=1}^n N_i \mathbf{x}_i, \quad (32)$$

where $N_i = N_i(\xi^\alpha)$ and n denote the shape function and number of nodes, respectively. For shorthand notation, Equations (32) and (2) can be expressed as follows:

$$\mathbf{x} \approx \mathbf{x}^h = \mathbf{N} \mathbf{x}_e, \quad (33)$$

$$\mathbf{a}_\alpha = \mathbf{N}_{,\xi^\alpha} \mathbf{x}_e \quad (34)$$

with $\mathbf{N}_{,\xi^\alpha} = \partial \mathbf{N} / \partial \xi^\alpha$.

To write the discretized weak form of the governing equation, the variation of the tangent vector \mathbf{a}_α , current nodal position vector of the nodes \mathbf{x} , and the unit normal vector \mathbf{n} are required.

The variation of current position vector in Equation (33) and the tangent vector in Equation (34) is written as

$$\delta \mathbf{x} = \mathbf{N} \delta \mathbf{x}_e \quad (35)$$

$$\delta \mathbf{a}_\alpha = \mathbf{N}_{,\xi^\alpha} \delta \mathbf{x}_e \quad (36)$$

The variation of the unit normal vector is given by [30]

$$\delta \mathbf{n} = \mathbf{R}_A^\alpha \delta \mathbf{a}_\alpha, \quad (37)$$

with

$$\mathbf{R}_A^\alpha = -(\mathbf{a}^\alpha \otimes \mathbf{n}). \quad (38)$$

The linearization of vector \mathbf{n} is followed from Equation (37) as

$$\Delta \delta \mathbf{n} = (-\Delta \mathbf{a}^\alpha \otimes \mathbf{n}) \delta \mathbf{a}_\alpha - (\mathbf{a}^\alpha \otimes \Delta \mathbf{n}) \delta \mathbf{a}_\alpha, \quad (39)$$

where $\Delta \mathbf{a}^\alpha$ is can be found in [31]

$$\Delta \mathbf{a}^\alpha = \mathbf{B}^{\alpha\beta} \Delta \mathbf{a}_\beta \quad (40)$$

with $\mathbf{B}^{\alpha\beta} = \mathbf{a}^{\alpha\beta} (\mathbf{n} \otimes \mathbf{n}) - \mathbf{a}^\beta \otimes \mathbf{a}^\alpha$, and $\Delta \mathbf{a}_\alpha$ is the same manner as $\delta \mathbf{a}_\alpha$ from Equation (36). The variation of J is given respectively in [29]

$$\delta J = J \mathbf{a}^\alpha \cdot \delta \mathbf{a}_\alpha. \quad (41)$$

Further, the variation and the linearization of $\mathbf{a}_{\alpha\beta}$ and $\mathbf{a}^{\alpha\beta}$ are:

$$\delta \mathbf{a}_{\alpha\beta} = \mathbf{a}_\alpha \cdot \delta \mathbf{a}_\beta + \delta \mathbf{a}_\alpha \cdot \mathbf{a}_\beta, \quad (42)$$

$$\delta \mathbf{a}^{\alpha\beta} = \mathbf{a}^\alpha \cdot \delta \mathbf{a}^\beta + \delta \mathbf{a}^\alpha \cdot \mathbf{a}^\beta, \quad (43)$$

$$\Delta \delta \mathbf{a}_{\alpha\beta} = \Delta \mathbf{a}_\alpha \cdot \delta \mathbf{a}_\beta + \delta \mathbf{a}_\alpha \cdot \Delta \mathbf{a}_\beta. \quad (44)$$

5.2. Discretized Weak Form of the Governing Equation

With the expressions in Section 5.1, the weak form is now discretized. Accordingly, Equation (20) can be written as follows:

$$\delta \Pi_p \approx \delta \Pi_p^h = \delta \mathbf{x}^T \sum_{e=1}^{n_e} \mathbf{f}_p^e, \quad (45)$$

where

$$\mathbf{f}_p^e := \int_{\Omega_0^e} \mathbf{g}_p \, dA_e. \quad (46)$$

In Equation (46), \mathbf{g}_p is to be defined for a given type of cam.

The linearization can be expressed as

$$\Delta \delta \Pi_p = \delta \mathbf{x}^T \mathbf{k}_p \Delta \mathbf{x}, \quad (47)$$

where \mathbf{k}_p is the tangent matrix. It can be written as

$$\mathbf{k}_p = \sum_{e=1}^{n_e} \mathbf{k}_p^e. \quad (48)$$

5.3. Discretized Stabilization Terms

The weak form of the stabilization equation is discretized. The formula as shown in Equation (26) can be expressed as

$$\delta \Pi_{\text{stab}} \approx \delta \Pi_{\text{stab}}^h = \delta \mathbf{x}^T \sum_{e=1}^{n_e} \mathbf{f}_{\text{stab}}^e, \quad (49)$$

where

$$\mathbf{f}_{\text{stab}}^e := \int_{\Omega_0^e} \mathbf{g}_{\text{stab}} \, dA_e. \quad (50)$$

The linearization of the stabilization functional can be expressed as

$$\Delta \delta \Pi_{\text{stab}} = \delta \mathbf{x}^T \mathbf{k}_{\text{stab}} \Delta \mathbf{x}, \quad (51)$$

where the tangent matrix can be written by

$$\mathbf{k}_{\text{stab}} := \sum_{e=1}^{n_e} \mathbf{k}_{\text{stab}}^e. \quad (52)$$

6. Finite Element Solution

6.1. Solution Procedure

The discretized weak form as shown in Equations (45) and (49) of the governing equation and the stabilization equation can be written as

$$\delta \mathbf{x}^T [\mathbf{f}_p + \mathbf{f}_{\text{stab}}] = 0 \quad \forall \delta \mathbf{x}, \quad (53)$$

where

$$\mathbf{f}_p := \sum_{e=1}^{n_e} \mathbf{f}_p^e; \mathbf{f}_{\text{stab}} := \sum_{e=1}^{n_e} \mathbf{f}_{\text{stab}}^e. \quad (54)$$

The nonlinear in Equation (53) can be solved by numerical methods. The Newton–Raphson method is used to solve this equation. Equation (53) can be described as

$$\mathbf{f}(\mathbf{x}) := \mathbf{f}_p + \mathbf{f}_{\text{stab}} = 0. \quad (55)$$

At iteration i , an approximation solution of cam profile can be sought and denoted by \mathbf{x}^i . The method can be described by Gouri Dhatt, et al. [32]

$$\mathbf{f}(\mathbf{x}^i) := \mathbf{f}(\mathbf{x}^{i-1} + \Delta \mathbf{x}^i) \approx 0, \quad (56)$$

The algorithm is obtained by developing this residual into a Taylor series in the vicinity of \mathbf{x}^{i-1}

$$\mathbf{f}(\mathbf{x}^{i-1} + \Delta \mathbf{x}^i) = \mathbf{f}(\mathbf{x}^{i-1}) + \left[\frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right]_{\mathbf{x}=\mathbf{x}^{i-1}} \Delta \mathbf{x}^i + \dots = 0, \quad (57)$$

and

$$\mathbf{x}^i = \mathbf{x}^{i-1} + \Delta \mathbf{x}^i, \quad (58)$$

By ignoring terms greater than one, we get

$$\mathbf{K} \Delta \mathbf{x}^i = -\mathbf{f}(\mathbf{x}^{i-1}), \quad (59)$$

where

$$\mathbf{K} := \frac{\partial \mathbf{f}}{\partial \mathbf{x}} = \mathbf{k}_p + \mathbf{k}_{\text{stab}} \quad (60)$$

is the tangent matrix.

6.2. Numerical Quadrature

For finite element computations, the integral for the weak form and the tangent matrices must be evaluated. These integrations can be performed on the element level. Since the integration is carried out in the finite element analysis of the reference configuration, the parameter domain coordinate $\xi^\alpha \in [-1, +1]$ must be transformed to this configuration (see Figure 3).

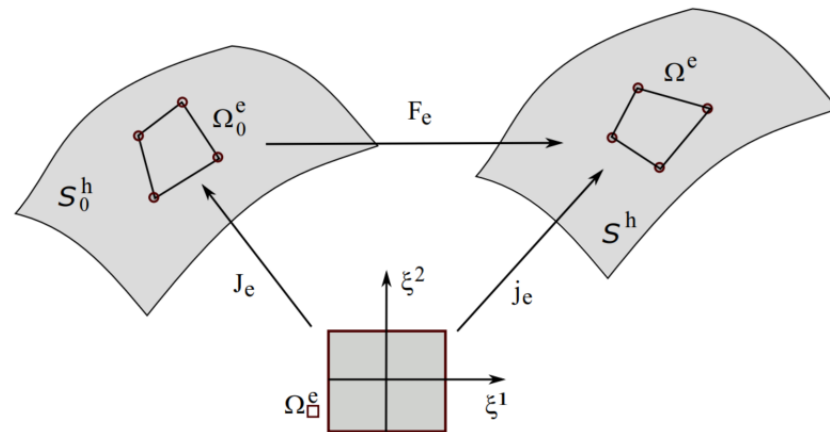


Figure 3. Mapping Gaussian integration between element Ω^e of parameter domain P , element Ω_0^e of discretized reference configuration S_0^h and element Ω^e of the discretized current configuration S^h .

The Jacobian can be calculated as follows [30]:

$$\begin{aligned} \mathbf{J}_e &= \frac{\partial \mathbf{X}}{\partial \xi} \approx \sum_A^n \mathbf{X}_A \otimes \frac{\partial \mathbf{N}_A}{\partial \xi}, \\ \mathbf{j}_e &= \frac{\partial \mathbf{x}}{\partial \xi} \approx \sum_A^n \mathbf{x}_A \otimes \frac{\partial \mathbf{N}_A}{\partial \xi}, \\ \mathbf{F}_e &= \mathbf{j}_e \mathbf{J}_e^{-1}. \end{aligned} \quad (61)$$

The integration form is then carried out with the standard Gaussian quadrature on the parameter domain. Thus, from Equations (46) and (50), the governing equation of each element can be written as

$$\int_{\Omega_e} \mathbf{g} \, dA_e = \int_{\Omega_e} \mathbf{g}_p \, dA_e + \int_{\Omega_e} \mathbf{g}_{stab} \, dA_e \quad (62)$$

The integration in Equation (62) is computed by

$$\begin{aligned} \int_{\Omega_e} \mathbf{g} \, dA_e &:= \int_{\Omega} \mathbf{g}_p(\xi) \det \mathbf{J}_e \, d\xi + \int_{\Omega} \mathbf{g}_{stab}(\xi) \det \mathbf{J}_e \, d\xi \\ &= \int_{-1}^{+1} \int_{-1}^{+1} \mathbf{g}_p(\xi^\alpha) \det \mathbf{J}_e \, d\xi^1 d\xi^2 + \int_{-1}^{+1} \int_{-1}^{+1} \mathbf{g}_{stab}(\xi^\alpha) \det \mathbf{J}_e \, d\xi^1 d\xi^2. \end{aligned} \quad (63)$$

The integration will be done numerically. n_{gp} is representative for the number of Gauss points. Hence, the integral in Equation (63) will be approximated by

$$\int_{\Omega_e} \mathbf{g} \, dA_e \approx \sum_{gp=1}^{n_{gp}} \mathbf{g}_p(\xi_{gp}) \det \mathbf{J}_e W_p + \sum_{gp=1}^{n_{gp}} \mathbf{g}_{stab}(\xi^\alpha) \det \mathbf{J}_e W_p. \quad (64)$$

Here, W_p is weighting factors and ξ_{gp} denotes the coordinates of the evaluation points. The integration for the whole system can be computed as follows:

$$\sum_{e=1}^{n_e} \int_{\Omega_e} \mathbf{g} \, dA_e \approx \sum_{e=1}^{n_e} \sum_{gp=1}^{n_{gp}} \mathbf{g}_p(\xi_{gp}) \det \mathbf{J}_e W_p + \sum_{e=1}^{n_e} \sum_{gp=1}^{n_{gp}} \mathbf{g}_{stab}(\xi^\alpha) \det \mathbf{J}_e W_p. \quad (65)$$

6.3. Computational Algorithm

This section presents a computational to solve a cam profile. With the above-discussed framework, the computational algorithm of a cam profile that satisfies a prescribed displacement of the follower is given in Table 1.

Table 1. The computational algorithm.

1. Defined the initial configuration
2. Step loop
<ul style="list-style-type: none"> Given parameters: tolerances for the reduced penalty parameter μ, and augmentedLagrangian; penalty parameter ϵ; parameter λ
2.1. Newton iteration:
<ul style="list-style-type: none"> Compute $\mathbf{k}_p, \mathbf{k}_{stab}, \mathbf{f}_p, \mathbf{f}_{stab}$, as shown in Equations (48), (52), and (54) respectively Apply boundary conditions Step 1: Load step for penalty method Step 2: Reduce parameter μ Step 3: Update new parameter λ of Augmented Lagrangian method following as equation $\lambda_{new} = \epsilon g + \lambda_{old}$ Solve $\mathbf{K} \Delta \mathbf{x}^i = -\mathbf{f}(\mathbf{x}^{i-1})$ and update $\mathbf{x}^i = \mathbf{x}^{i-1} + \Delta \mathbf{x}^i$
<ul style="list-style-type: none"> Update reference configuration

7. Case Study

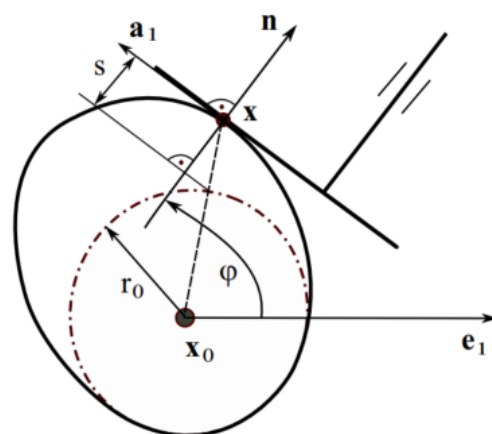
This section presents a case study to demonstrate the procedure of this work. We consider the cam mechanism with a flat-face follower for illustration. The finite element is therefore reduced to line elements. With the general designing framework discussed above, an explicit expression for the follower displacement is required. The expression is, however, generally dependent on the type of cam. The cam curves resulted from the proposed framework are compared with the analytical solution.

7.1. Computation of Cam Flat-Face Follower

With the flat-face follower, as shown in Figure 4, the function of the follower displacement s_0 can be written as follows:

$$s_0 = f(\varphi), \quad (66)$$

where φ is the angle of the camshaft, and the displacement function $f(\varphi)$ is shown in Appendix A.

**Figure 4.** Calculation parameters of cam flat-face follower.

For cam plate, we can describe the tangent vector and others with $\alpha = 1$ and $\beta = 1$. ξ^1 is denoted by ξ . From the variation of the stabilized equation (see Equation (26)), \mathbf{f}_{stab}^e can be computed as follows:

$$\mathbf{f}_{stab}^e = \int_{\Omega_0^e} \mu (A^{11} - a^{11}) \mathbf{N}_{,\xi}^T \mathbf{a}_1 dA_e. \quad (67)$$

Likewise, the tangent matrix is calculated by

$$\mathbf{k}_{\text{stab}}^e = (\mathbf{k}_{\text{stab1}}^e + \mathbf{k}_{\text{stab2}}^e), \quad (68)$$

where

$$\begin{aligned} \mathbf{k}_{\text{stab1}}^e &= \int_{\Omega_0^e} 2\mu \mathbf{a}^{11} \mathbf{N}_{,\xi}^T \mathbf{a}_1 \mathbf{a}_1 \mathbf{N}_{,\xi}^T dA_e, \\ \mathbf{k}_{\text{stab2}}^e &= \int_{\Omega_0^e} \mu \left(\mathbf{A}^{11} - \mathbf{a}^{11} \right) \mathbf{N}_{,\xi}^T \mathbf{N}_{,\xi} dA_e. \end{aligned} \quad (69)$$

As shown in Figure 4, the displacement of the follower can be computed as

$$s := \mathbf{x} \cdot \mathbf{n} - r_0. \quad (70)$$

Taking the variation of the current displacement in Equation (70) is presented by

$$\delta s = \delta \mathbf{x} \cdot \mathbf{n} + \mathbf{x} \cdot \delta \mathbf{n}. \quad (71)$$

The angle of the camshaft can be written as (see Figure 4)

$$\varphi = \arccos(\mathbf{n} \cdot \mathbf{e}_1). \quad (72)$$

The variation of s_0 can be also computed by

$$\delta s_0 = -d_c \mathbf{e}_1 \cdot \delta \mathbf{n}. \quad (73)$$

In Equation (73), d_c is calculated in Appendix A.

From the calculation above, \mathbf{f}_p^e can be simply obtained by

$$\mathbf{f}_p^e = \int_{\Omega_0^e} \left(\mathbf{N}^T \mathbf{n} + \mathbf{N}_{,\xi}^T \mathbf{R}_A^T \mathbf{b} \right) dA_e, \quad (74)$$

where, $\mathbf{R}_A := \mathbf{R}_A^1$. The tangent matrix is

$$\mathbf{k}_p^e = \mathbf{k}_{p1}^e + \mathbf{k}_{p2}^e. \quad (75)$$

Here

$$\begin{aligned} \mathbf{k}_{p1}^e &= \int_{\Omega_0^e} \epsilon (\mathbf{n} \otimes \mathbf{n} \mathbf{N} + \mathbf{n} \otimes \mathbf{b} \mathbf{R}_A \mathbf{N}_{,\xi}) dA_e \\ &+ \int_{\Omega_0^e} \epsilon \mathbf{N}_{,\xi}^T \mathbf{R}_A^T (\mathbf{b} \otimes \mathbf{n} \mathbf{N} + \mathbf{b} \otimes \mathbf{b} \mathbf{R}_A \mathbf{N}_{,\xi}) dA_e, \end{aligned} \quad (76)$$

and

$$\mathbf{k}_{p2}^e = \int_{\Omega_0^e} \left(\mathbf{N}^T \mathbf{R}_A \mathbf{N}_{,\xi} + \mathbf{N}_{,\xi}^T \mathbf{R}_A^T \mathbf{G}_1 - \mathbf{N}_{,\xi}^T \mathbf{G}_2 \mathbf{N}_{,\xi} \right) dA_e. \quad (77)$$

Tensor \mathbf{b} in Equation (76) and tensors \mathbf{G}_1 , \mathbf{G}_2 in Equation (77) are defined in Appendix A.

In the example, the cycloid function for the displacement of the follower as shown in Equations (A1) and (A2) in Appendix A is used for the design of the cam profile. The types of motion programs are rise, fall, and dwell. These motions are defined by the angle of segments, i.e., $\beta_1 = \beta_2 = \beta_3 = 120^\circ$ and the lift $L = 5$ mm. By using the computational algorithm in Section 6.3, the cam profile is shown in Figure 5a and the kinematic characteristics of the follower including displacement, velocity, and acceleration are presented in Figure 5b–d. The finite element solution of the cam profile is compared with the analytical solution as shown in Figure 5. From this figure, it can be seen that there are very small differences between the FE solution and the analysis solution.

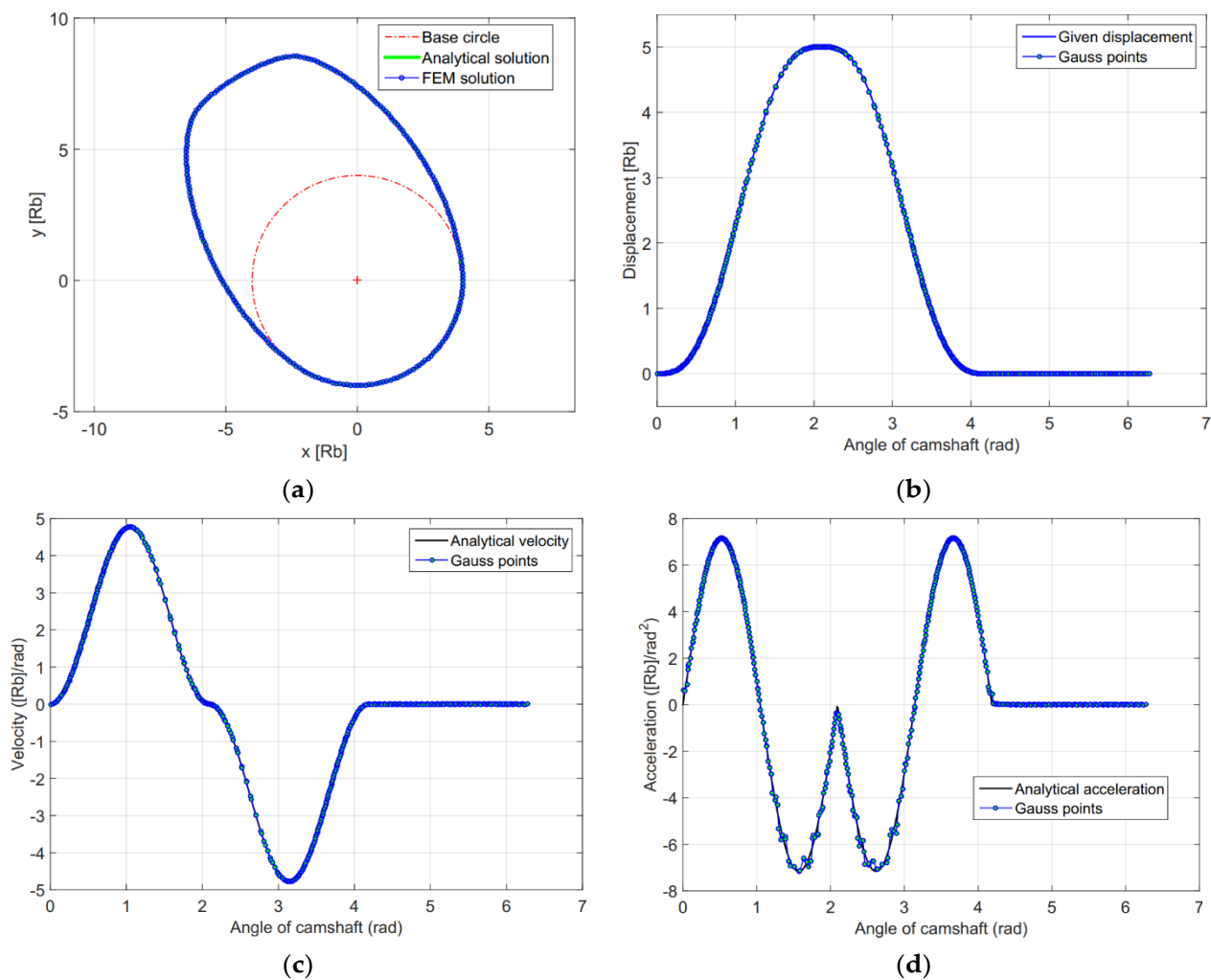


Figure 5. Compare between FE solution with 128 elements and analytical solution (a) comparison of cam profile; (b) comparison of displacement; (c) comparison of velocity; (d) comparison of acceleration.

In this example, 128 elements are used to compute the cam profile. In the penalty step, the maximum difference between the FE displacement and the input displacement is small, i.e., $\varepsilon = 1.9 \times 10^{-3}$. For the second step, the stabilization parameter μ is reduced from 100 to 25, which increases the accuracy in the displacement to $\varepsilon = 7.0282 \times 10^{-5}$. For the augmented step, the accuracy is further improved with $\varepsilon = 2.6133 \times 10^{-5}$.

Next, the size of the cam is considered in the cam design process since it decides the cost of the cam follower. The cam size is influenced by the base circle r_O . Usually, a minimum base circle, denoted by $r_{O_{\min}}$, can be defined by (see Robert L. Norton [23])

$$r_{O_{\min}} > \rho_{\min} - s_{@a_{\min}} - a_{\min}, \quad (78)$$

where ρ_{\min} , a_{\min} , $s_{@a_{\min}}$ are the minimum radius of curvature, minimum acceleration, and the displacement value at a_{\min} , respectively.

In order to determine the minimum base circle, the parameters including ρ_{\min} , $s_{@a_{\min}}$, and a_{\min} need to be computed. The minimum radius of curvature ρ_{\min} is calculated from Equation (7). Values of $s_{@a_{\min}}$ and a_{\min} are determined from the displacement and acceleration functions. Substituting values of the parameters into Equation (78), the minimum base circle is 3.9878 mm. In addition, with the cam flat-face follower, the radius of curvature must be positive to maintain the contact between the cam and the follower.

7.2. Influence of Parameters ϵ and μ

Next, the influence of penalty parameters ϵ and μ is discussed in this section. The accuracy of the cam profile is estimated by parameter ϵ as shown in Figure 6a. It is clearly seen that an accurate solution is obtained for a sufficiently large ϵ . However, it should be noted that for the augmented method, a much smaller penalty parameter is already sufficient for the comparative accuracy due to the extra correction loop.

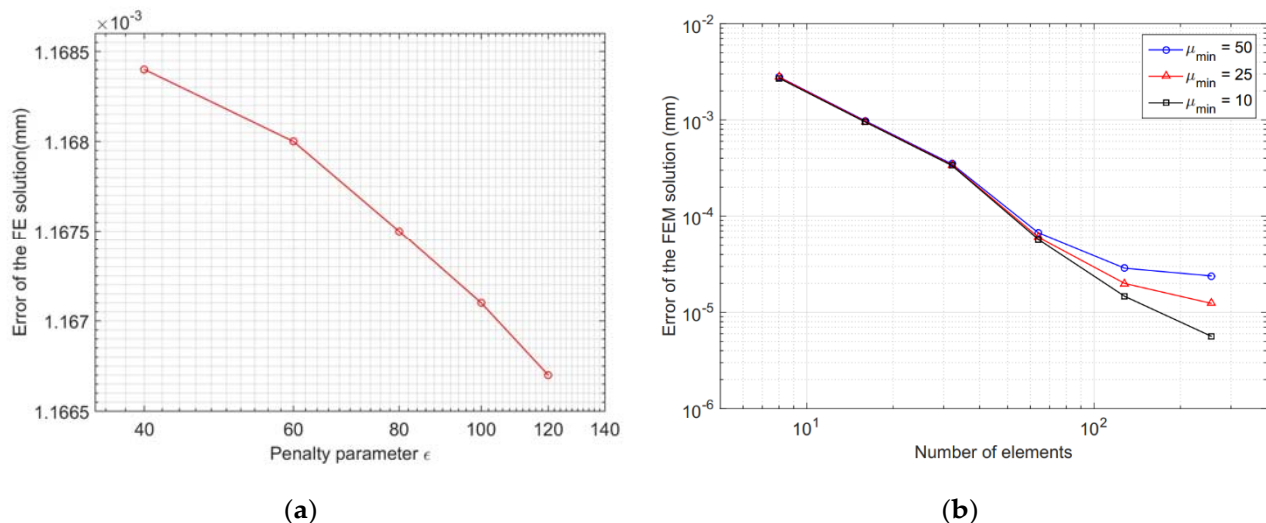


Figure 6. Influence of parameters ϵ and μ : (a) penalty parameter ϵ ; (b) parameter μ .

Figure 6b further shows the convergence of the finite element calculation with mesh refinement for three minimum values of the parameter $\mu = 50, 25$, and 10 . It is observed from the figure that the rate of convergence increases with decreasing the parameter μ . On the other hand, for the fine meshes, parameter μ affects the convergent rate significantly, i.e., the smaller stabilization parameter, the more accuracy is obtained.

8. Conclusions

This paper presents a new and general computational formulation for the synthesis of cam mechanisms using the finite element method. The formulation is based on a general theoretical description of the cam curve in curvilinear coordinates. The governing equation for the synthesis of cam profile is proposed. The penalty-type functional is proposed for the weak form. The stabilization functional is further added for convexity of the total potential. The discretization of the weak form is derived in the framework of the finite element. The Gaussian quadrature is used to integrate the finite element computation. The cam profile is obtained by solving a nonlinear system of equations resulting from minimizing the total functional.

The case study of the flat-face cam mechanism demonstrates that the formulation is able to handle all aspects accurately, robustly, and efficiently. The proposed framework can be applied for more general cam mechanisms, which is a subject of future work.

Author Contributions: Data curation, V.-S.N.; Formal analysis, T.X.D. and V.-S.N.; Investigation, T.T.N.N.; Methodology, T.T.N.N. and T.X.D.; Resources, T.X.D.; Writing—original draft, T.T.N.N.; Writing—review & editing, T.T.N.N. and T.X.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are available from the authors upon request.

Acknowledgments: This work is supported by Thai Nguyen University of Technology (TNUT).

Conflicts of Interest: The authors declare no conflict of interest. They do not know of any competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A

This appendix presents the displacement function of the follower $f(\varphi)$, vector \mathbf{b} and tensors \mathbf{G}_1 , \mathbf{G}_2 in Section 7.

The displacement function $f(\varphi)$ for the rise and fall can be expressed as

$$f^{\text{rise}}(\varphi) = L \left(\frac{\varphi}{\beta_1} - \frac{1}{2\pi} \sin \frac{2\pi\varphi}{\beta_1} \right), \quad (\text{A1})$$

$$f^{\text{fall}}(\varphi) = L \left(1 - \frac{\varphi}{\beta_2} + \frac{1}{2\pi} \sin \frac{2\pi\varphi}{\beta_2} \right). \quad (\text{A2})$$

Vector \mathbf{b} is defined by

$$\mathbf{b} := \mathbf{x} + d_c \mathbf{e}_1 \quad (\text{A3})$$

Further, tensors \mathbf{G}_1 and \mathbf{G}_2 are defined by

$$\mathbf{G}_1 = \mathbf{N} + d_{c1} \mathbf{A} \mathbf{e}_1 \otimes \mathbf{e}_1 \mathbf{R}_A \mathbf{N}_{,\xi}, \quad (\text{A4})$$

$$\mathbf{G}_2 = (\mathbf{n} \otimes \mathbf{b}) \mathbf{B}^{11} + (\mathbf{b} \cdot \mathbf{a}^1) \mathbf{R}_A \quad (\text{A5})$$

with

$$d_c = f_{,\varphi} c_s, \quad (\text{A6})$$

and

$$d_{c1} = f_{,\varphi\varphi} c_s^2 + f_{,\varphi} c_s \sin \varphi \cos \varphi. \quad (\text{A7})$$

here

$$c_s = \frac{1}{\sqrt{(1 - (\cos \varphi)^2)}}. \quad (\text{A8})$$

References

1. Uicker, J.J.; Pennock, G.R.; Shigley, J.E.; McCarthy, J.M. *Theory of Machines and Mechanisms*, 5th ed.; Oxford University Press: New York, NY, USA, 2003.
2. Mermelstein, S.P.; Acar, M. Optimising cam motion using piecewise polynomials. *Eng. Comput.* **2003**, *19*, 241–254. [CrossRef]
3. Cardona, S.; Zayas, E.; Jordi, L.; Català, P. Synthesis of displacement functions by Bézier curves in constant-breadth cams with parallel flat-faced double translating and oscillating followers. *Mech. Mach. Theory* **2013**, *62*, 51–62. [CrossRef]
4. Sateesh, N.; Rao, C.S.; Reddy, T.A.J. Optimisation of cam-follower motion using B-splines. *Int. J. Comput. Integr. Manuf.* **2009**, *22*, 515–523. [CrossRef]
5. Jiang, J.K.; Iwai, Y.R. Improving the B-Spline Method of Dynamically-Compensated Cam Design by Minimizing or Restricting Vibrations in High-Speed Cam-Follower Systems. *J. Mech. Des.* **2009**, *131*, 041003. [CrossRef]
6. Nguyen, T.T.N.; Nguyen, V.S.; Nguyen, T.B.N.; Vu, T.L. An Evaluation of B-Spline for Synthesis of Cam Motion with a Large Number of Output Conditions. In *Advances in Engineering Research and Application, Proceedings of the International Conference on Engineering Research and Applications, ICERA 2020, Thai Nguyen, Vietnam, 1–2 December 2020*; Nguyen, D.C., Vu, N.P., Long, B.T., Puta, H., Eds.; Springer: Heidelberg, Germany.
7. Müller, M.; Hüsing, M.; Beckermann, A.; Corves, B. Linkage and Cam Design with MechDev Based on Non-Uniform Rational B-Splines. *Machines* **2020**, *8*, 5. [CrossRef]
8. Nguyen, T.T.N.; Kurtenbach, S.; Hüsing, M.; Corves, B. A general framework for motion design of the follower in cam mechanisms by using non-uniform rational B-spline. *Mech. Mach. Theory* **2019**, *137*, 374–385. [CrossRef]
9. Nguyen, T.T.N.; Kurtenbach, S.; Hüsing, M.; Corves, B. Evaluating the Knot Vector to Synthesize the Cam Motion Using NURBS. *Mech. Mach. Sci.* **2018**, *50*, 209–216. [CrossRef]
10. Nguyen, T.T.N.; Kurtenbach, S.; Hüsing, M.; Corves, B. Improving the Kinematics of Motion Curves for Cam Mechanisms Using NURBS. *Mech. Mach. Sci.* **2018**, *52*, 79–88. [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. Qin, W.; Chen, Y. Study on optimal kinematic synthesis of cam profiles for engine valve trains. *Appl. Math. Model.* **2014**, *38*, 4345–4353. [[CrossRef](#)]
13. Xia, B.-Z.; Liu, X.-C.; Shang, X.; Ren, S.-Y. Improving cam profile design optimization based on classical splines and dynamic model. *J. Cent. South. Univ.* **2017**, *24*, 1817–1825. [[CrossRef](#)]
14. 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](#)]
15. 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](#)]
16. Navarro, O.; Wu, C.-J.; Angeles, J. The size-minimization of planar cam mechanisms. *Mech. Mach. Theory* **2001**, *36*, 371–386. [[CrossRef](#)]
17. Yu, Q.; Lee, H.P. Size optimization of cam mechanisms with translating roller followers. *Proc. Inst. Mech. Eng. Part C* **1998**, *212*, 381–386. [[CrossRef](#)]
18. Rothbart, H.A. *Cam Design Handbook*; McGraw-Hill: New York, NY, USA, 2004.
19. Myszka, D.H. *Machines and Mechanisms: Applied Kinematic Analysis*, 4th ed.; Pren-tice Hall: Upper Saddle River, NJ, USA, 2012.
20. Gupta, B.V.R. *Theory of Machines: Kinematics and Dynamics*; I.K. international Pvt: New Delhi, India, 2010.
21. Li, F.; Feng, X. The design of parallel combination for cam mechanism. *Procedia Environ. Sci.* **2011**, *10*, 1343–1349.
22. Shala, A.; Likaj, R. Analytical Method for Synthesis of Cam Mechanism. *Int. J. Curr. Eng. Technol.* **2013**, *133*, 432–435.
23. Norton, R. *Design of Machinery*; McGraw-Hill: New York, NY, USA, 2003.
24. Wu, L.-I.; Chang, W.-T.; Liu, C.-H. The design of varying-velocity translating cam mechanisms. *Mech. Mach. Theory* **2007**, *42*, 352–364. [[CrossRef](#)]
25. Chen, F.Y. *Mechanics and Design of Cam Mechanisms*; Pergamon Press: New York, NY, USA, 1982.
26. Biswas, A.; Stevens, M.; Kinzel, G.L. A comparison of approximate methods for the analytical determination of profiles for disk cams with roller followers. *Mech. Mach. Theory* **2004**, *39*, 645–656. [[CrossRef](#)]
27. Hsieh, J.-F. Design and analysis of cams with three circular-arc profiles. *Mech. Mach. Theory* **2010**, *45*, 955–965. [[CrossRef](#)]
28. Hsieh, J.-F. Design and analysis of indexing cam mechanism with parallel axes. *Mech. Mach. Theory* **2014**, *81*, 155–165. [[CrossRef](#)]
29. Sauer, R.A.; Duong, T.X.; Corbett, C.J. A computational formulation for constrained solid and liquid membranes considering isogeometric finite elements. *Comput. Methods Appl. Mech. Eng.* **2014**, *271*, 48–68. [[CrossRef](#)]
30. Swinggers, P. *Computational Contact Mechanics*, 2nd ed.; Springer: Heidelberg, Germany, 2006.
31. Sauer, R.A. A contact theory for surface tension driven systems. *Math. Mech. Solids* **2016**, *21*, 305–325. [[CrossRef](#)]
32. Dhatt, G.; Touzot, G.; Lefrançois, E. *Finite Element Method*, 1st ed.; Wiley-ISTE: Hoboken, NJ, USA, 2012.