Optimising Cam Motion Using Piecewise Polynomials

S P Mermelstein and M Acar

Loughborough University, UK

Address:
Dr M Acar, Senior Lecturer
Dr S P Mermelstein, Research Associate
Wolfson School of Mechanical and Manufacturing Engineering
Loughborough University, Leics. LE11 3TU, UK

Optimising Cam Motion Using Piecewise Polynomials

S P Mermelstein and M Acar

Loughborough University, UK

Abstract

The paper presents method of synthesising cam profiles based on the use of Piecewise Polynomials together with an optimisation technique. Special cases and limitations are discussed and illustrated, making the procedure complete and systematic for any design requirements.

Using Piecewise Polynomials, the complete cam profile can be designed as a combined linear system. The optimisation technique described goes even further, manipulating the variables in the linear system to select the ideal combination. In addition, a means to prove the validity of the results is explained.

Keywords: Cam motion synthesis; piecewise polynomials; optimisation.

1.   Introduction

Cam motion syntheses have been studied extensively as the use of cams has been widespread in many industries. Formulaic methods of synthesis for a number of 'standard' profiles have been well established (Erdman and Sandor [1]; Shigley and Uicker [2]). However recommendations as to how to optimise a cam profile for a certain application are very scarce. Polynomial profiles, which are defined with a number of breakpoints in the profile for which the designer can define boundary conditions, are the most suitable for optimisation because the displacement, velocity, acceleration and jerk can be controlled for any critical point in the cam profile.

Farouki [3] et al describe an approach to cam design focusing on the transferability of the cam profile into CNC machines or CAD systems. Yu and Lee [4] use a standard non-linear optimisation technique to investigate the effect of the features of a motion curve, such as the velocity, acceleration and boundary continuity on the minimum size of the cam.

A single cam motion can be synthesised using a polynomial profile, however, using Piecewise Polynomials (PP), the complete cam profile can be designed as a combined linear system. PP are best suited for implementation of a computer aided design and manufacturing program as it can be generalised and is not case dependant. On the other hand, in order to use PP to design a cam profile, values for the displacement and its derivatives at the profile breakpoints must be known. This can sometimes make the procedure impractical.

Wang and Yang [5] used a quadratic programming (QP) algorithm to optimise the shape of the motion profile created using PP such that the jerk level is reduced to a minimum. The optimisation is carried out using any unspecified breakpoint boundary conditions as variables. That is, the unspecified breakpoint boundary conditions can take any value. However, this method is only applicable to certain combinations of boundary conditions, not all possible combinations. The advantages and limitations of the method were not explained by the

authors. This paper re-defines the method illustrating its restrictions, without a good understanding of which it cannot be used reliably. Furthermore, it suggests an alternative approach to the implementation of the technique which is demonstrated by application examples using Maple Mathematics Software.

2.   Piecewise Polynomials

P*iecewise polynomials* are normally used to define a curve, which is constructed from many "pieces". Advantage of being a polynomial is to have continuous derivatives of all orders. Parametric polynomial fits result in a global fit where one set of fitted coefficients describes the entire data set. As a result, the fit can be erratic. Because piecewise polynomials always produce a smooth fit, they are more flexible than parametric polynomials and can be effectively used for a wider range of data sets.

Cubic spline is a typical example of piecewise polynomial application in the field of approximation and interpolation of data sets where each piece is described by four coefficients, which are calculated using a cubic (third-degree) polynomial. This method fits a different cubic polynomial between each pair of data points. A digital-to-analogue converter is a useful application of spline interpolation for audio and communication systems. Third order piecewise polynomials are also used in image interpolation, known as cubic convolution, in which the ideal sinc function is modelled by a finite extent kernel. Another area of interest is approximation by piecewise polynomials defined over triangulations of polygonal regions in the plane. Applications include interpolation to scattered data and finite-element approximation to the solutions of fourth-order elliptic equations. The approximants are piecewise quadratics or piecewise cubics on each triangle.

The piecewise polynomials (PP) method is based on describing the displacement of a cam follower with a series of polynomial expressions; one for each segment of the profile. The theory is explained and the method is illustrated by a case study.

The displacement imparted by a cam profile comprising n steps can be described using a polynomial function for each one of the non-dwelling segments in the form,

$$y_i(\theta) = \sum_{j=1}^{k} b_{ij}(\theta - \beta_i)^{j-1} \quad , \qquad \forall \theta \in \{\beta_i \leq \theta \leq \beta_{i+1}\} \quad , \qquad i = 1,...,n-1 \qquad (1)$$

where θ :   Cam angle

b_{ij} :   Displacement coefficients, given by boundary conditions.

k :   Polynomial order

β_i :   The ith breakpoint in the profile in radians.

n :   Number of breakpoints in the profile.

Velocity, acceleration, jerk and any subsequent derivatives of the displacement functions can be expressed similarly as:

$$v_i(\theta) = \dot{y}_i(\theta) = \sum_{j=2}^{k} b_{ij} \cdot (j-1) \cdot (\theta - \beta_i)^{j-2} \quad ,$$

$$\forall \theta \in \{\beta_i \leq \theta \leq \beta_{i+1}\} \quad , \qquad i = 1,...,n-1 \qquad (2)$$

$$\vdots$$

$$s_i(\theta) = y_i^{(4)}(\theta) = \sum_{j=4}^{k} b_{ij} \cdot (j-1) \cdot (j-2) \cdot (j-3) \cdot (j-4) \cdot (\theta - \beta_i)^{j-5} \,,$$

$$\forall \theta \in \{\beta_i \leq \theta \leq \beta_{i+1}\}, \qquad i = 1,\ldots,n-1 \tag{3}$$

where, $v_i(\theta)$: Polynomial function describing velocity for the ith segment.

$a_i(\theta)$: Polynomial function describing acceleration for the ith segment.

$j_i(\theta)$: Polynomial function describing jerk for the ith segment.

$s_i(\theta)$: Polynomial function describing jerk's derivative for the ith segment.

All the above functions are determined by differentiating the displacement with respect to θ; constant angular velocity (ω) is assumed.

$$v(t) = \frac{\partial y}{\partial t} = \frac{\partial y}{\partial \theta} \cdot \frac{\partial \theta}{\partial t} = v(\theta) \cdot \omega \tag{4}$$

Yao at al [6] has carried out research on optimising cam profiles using variable angular velocity. However, it is not always economically sound to implement 'Active Control' of a cam mechanism, as it requires the use of a servo drive and a motion control system in place of a standard drive.

2.1   Boundary Conditions

A designer will generally be able to define the desired displacement values at each breakpoint. As an example lets consider a follower rise and dwell cycle of 50mm amplitude which would be treated as a profile with three breakpoints; $\beta_1 = 0°, \beta_2 = 180°, \beta_3 = 360°$ with the displacement boundary conditions $y(\beta_1) = 0,\ y(\beta_2) = 50mm,\qquad y(\beta_3) = 0$. Other design requirements will determine the velocity, acceleration, jerk or s function (time derivative of jerk) at the boundaries between segments.

In general, these boundary conditions can be expressed as the following sets of $(n-1)$ equations;

$$y_i(\beta_i) = Disp_i, \quad i = 1,\ldots,n-1 \, ; \tag{5}$$

$$v_i(\beta_i) = Vel_i, \quad i = 1,\ldots,n-1 \, ;$$

$$a_i(\beta_i) = Accel_i, \quad i = 1,\ldots,n-1 \, ;$$

$$j_i(\beta_i) = Jerk_i, \quad i = 1,\ldots,n-1 \, ;$$

$$s_i(\beta_i) = S_i, \quad i = 1,\ldots,n-1 \, ; \tag{6}$$

Where $Disp_i, Vel_i, Accel_i, Jerk_i$ and S_i are constants. Note that in Eq. (6), each segment function is used to define only one of the boundaries of the segment, although polynomial functions are defined over two boundaries (i.e. segment 1 is defined over β_1 and β_2).

2.1   Continuity Conditions

Further equations are derived from ensuring continuity on the displacement function and its derivatives. The condition of continuity throughout the cam profile can be expressed as a set

of equations evaluating the polynomial motion functions at the boundary points. In the case of continuous displacement of the follower (which is necessary for the manufacture of the cam), continuity will be given by the following $(n-1)$ equations;

$$\begin{cases} y_i(\beta_{i+1}) = y_{i+1}(\beta_{i+1}), \ i = 1,...,n-2 \\ y_{n-1}(\beta_1) = y_1(\beta_1) \end{cases} \tag{7}$$

The latter ensures continuity at the end of the profile. Similar sets of equations can be deduced for each of the displacement derivatives;

$$\begin{cases} v_i(\beta_{i+1}) = v_{i+1}(\beta_{i+1}), \ i = 1,...,n-2 \\ v_{n-1}(\beta_1) = v_1(\beta_1) \end{cases}$$

$$\vdots \tag{8}$$

$$\begin{cases} s_i(\beta_{i+1}) = s_{i+1}(\beta_{i+1}), \ i = 1,...,n-2 \\ s_{n-1}(\beta_1) = s_1(\beta_1) \end{cases}$$

The expressions in Equations (5) provide a set of equations that, expressed as a linear system of equations, can be used to determine the polynomial coefficients b_{ij} of the cam displacement profile.

2.3 Polynomial Order

The designer must calculate the polynomial order k that will not under or over constrain the system depending on the specific design requirements.

The system will need to solve $k \cdot (n-1)$ coefficients defined in Eq. (1); therefore, the same number of equations will be required to solve them. The boundary conditions described in Eq. (5) and Eq. (6) provide $n-1$ equations for each of the displacement derivatives. Similarly, the continuity conditions in Eq. (7) and Eq. (8) provide $n-1$ equations for each of the displacement derivatives.

On the other hand, the polynomial function describing each segment (1) has k unknown coefficients, making the total number of unknowns in the profile $k \cdot (n-1)$. In order to solve them as a linear system, the same number of equations is required.

The procedure to calculate the appropriate polynomial order is easily illustrated using an example. The designer of a cam profile with 3 breakpoints has specific requirements for the cam's displacement;

$$y(\beta_1) = 0, \quad y(\beta_2) = 50mm \tag{9}$$

In addition, the designer requires that the velocity and acceleration at all breakpoints be equal to zero:

$$v_i(\beta_i) = 0, \quad i = 1,..,n-1;$$

$$a_i(\beta_i) = 0, \quad i = 1,..,n-1; \tag{10}$$

Finally, it is an implicit requirement for any cam profile that the displacement function is continuous throughout, as it would not be possible to manufacture the cam otherwise.

The polynomial order can be derived as follows. The designer would have $(n - 1)$ equations relating to the displacement boundary conditions Eq. (9), $(n - 1)$ equations ensuring continuity of the follower's displacement Eq. (7) and a further $2(n - 1)$ equations given by Eq.   (10).   Therefore the system will not be over or under-constrained when;

Number of coefficients to solve = Number of equations:

$$k \cdot (n-1) = 4 \cdot (n-1) \Rightarrow k = 4 \tag{11}$$

Wang and Yang erroneously state that an eighth-order polynomial is necessary to ensure that the motion gives continuous displacement, velocity and acceleration profiles and to allow the user to specify a dwell period between any two cam-rotation breakpoints. For each set of design requirements, the number of equations available must be calculated and the required polynomial order must be derived from Eq. (11).

Only when the designer provides breakpoint boundary conditions for four derivatives of the displacement profile (displacement, velocity, acceleration, jerk or s) and has set the continuity conditions on all of them, will an eighth order polynomial be suitable for defining each segment of the profile.

Thus, the first improvement that can be made to the method is to allow the designer more flexibility in deciding the polynomial order according to the boundary conditions available. The displacement derivatives used need not be consecutive. Depending on the requirements of the application any combination can be used. Similarly continuity can be imposed on any of the displacement derivatives.

2.4   Solving for the Polynomial Coefficients

Once the polynomial order has been defined, the explicit form of the motion profile can be expressed as the polynomial function Eq. (1). The objective of imposing the different boundary and continuity conditions is to create a linear system of equations that will enable the designer to solve for the polynomial coefficients defined in the displacement function.

Case Study #1 is a worked example of the process using Maple and is described below. It is based on the design of a single dwell cam for which the boundary conditions applied are summarised in Table 1.

Table 1 . Boundary Conditions for Piecewise Polynomials (Case Study #1)

Breakpoint No.	Cam Angle β (degrees)	Displacement (mm)	Velocity (mm/rad)	Acceleration (mm/rad^2)	S (mm/rad^4)
1	0°	0	0	0	0
2	90°	100	0	0	0
3	270°	100	0	0	0
4	360°	0	0	0	0

1. Establish the number of breakpoints required: $n = 4$;

    Note that the boundary conditions for breakpoint 1 at 0° is the same as those for breakpoint 4 at 360°.

2. Ascertain the boundary conditions to be used:

$y, \dot{y}, \ddot{y}, y^4$ plus continuity in all of them; a total of 8 conditions.

3. Define the polynomial order according to the boundary conditions specified above.

   $k = 8;$

4. Derive the equations from the boundary and continuity conditions for all segments, in terms of the polynomial coefficients.

   From Eq. (5) and Eq. (6) we can obtain 12 boundary condition equations:

   $$b_{1,1} = Disp_1; \quad b_{2,1} = Disp_2; \quad b_{3,1} = Disp_3; \quad b_{1,2} = Vel_1; \quad b_{2,2} = Vel_2; \quad b_{3,2} = Vel_3;$$

   $$2b_{1,3} = Accel_1; \quad 2b_{2,3} = Accel_2; \quad 2b_{3,3} = Accel_3; \quad 24b_{1,5} = s_1; \quad 24b_{2,5} = s_2; \quad 24b_{3,5} = s_3;$$
   $$\tag{12}$$

   Continuity in displacement (see Eq. (7)) can be expressed as

   $b_{1,1} + 1.571b_{1,2} + 2.467b_{1,3} + 3.876b_{1,4} + 6.088b_{1,5} + 9.563\, b_{1,6} + 15.022b_{1,7} + 23.596\, b_{1,8} = b_{2,1}$

   $b_{2,1} + 1.571b_{2,2} + 2.467b_{2,3} + 3.876b_{2,4} + 6.088b_{2,5} + 9.563\, b_{2,6} + 15.022b_{2,7} + 23.596\, b_{2,8} = b_{3,1}$

   $b_{3,1} + 3.142b_{3,2} + 9.870b_{3,3} + 31.006b_{3,4} + 97.409b_{3,5} + 306.020\, b_{3,6} + 961.389b_{3,7} + 3020.293b_{3,8} = b_{1,1}$
   $$\tag{13}$$

   Similar sets of equations are derived from continuity conditions for velocity, acceleration and s from Eq. (8).

5. Write these equations in matrix form, constructing a linear system in terms of the polynomial coefficients.

   $$\left[\bar{M}\right]\cdot\left[\bar{b}_{i,j}\right] = 0 \tag{14}$$

6. Solving the system, each coefficient can be expressed as a function of the breakpoints $\beta_1 \ldots \beta_{n-1}$ and the values for the boundary conditions $Disp_1 \ldots Disp_{n-1}, Vel_1 \ldots Vel_{n-1}, \ldots, s_1 \ldots s_{n-1}$.

7. Replacing the boundary conditions of Table 1, the polynomial coefficients can be calculated explicitly. In this case :

   $$\begin{bmatrix} b_{1,1} & b_{1,2} & b_{1,3} & b_{1,4} & b_{1,5} & b_{1,6} & b_{1,7} & b_{1,8} \\ b_{2,1} & b_{2,2} & b_{2,3} & b_{2,4} & b_{2,5} & b_{2,6} & b_{2,7} & b_{2,8} \\ b_{3,1} & b_{3,2} & b_{3,3} & b_{3,4} & b_{3,5} & b_{3,6} & b_{3,7} & b_{3,8} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 168 & 0 & -205 & 131 & -24 \\ 100 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 100 & 0 & 0 & -21 & 0 & 6 & -2 & 0 \end{bmatrix}$$

8. Replacing the values above into Eq. (1), the polynomial function describing the displacement of the follower is completely defined in terms of the boundary values given by the designer.

   $$y_1 = 168\theta^3 - 205\theta^5 + 131\theta^6 - 23.8\theta^7 \qquad \forall \theta \in \{\beta_1 \le \theta \le \beta_2\}$$
   $$y_2 = 100 \qquad \forall \theta \in \{\beta_2 \le \theta \le \beta_3\}$$
   $$y_3 = 100 - 21.2(\theta - \pi)^3 + 6.32(\theta - \pi)^5 - 2(\theta - \pi)^6 + .182(\theta - \pi)^7 \qquad \forall \theta \in \{\beta_3 \le \theta \le \beta_4\}$$

Figure 1 shows the displacement function and its derivatives. All the boundary and continuity conditions are met.
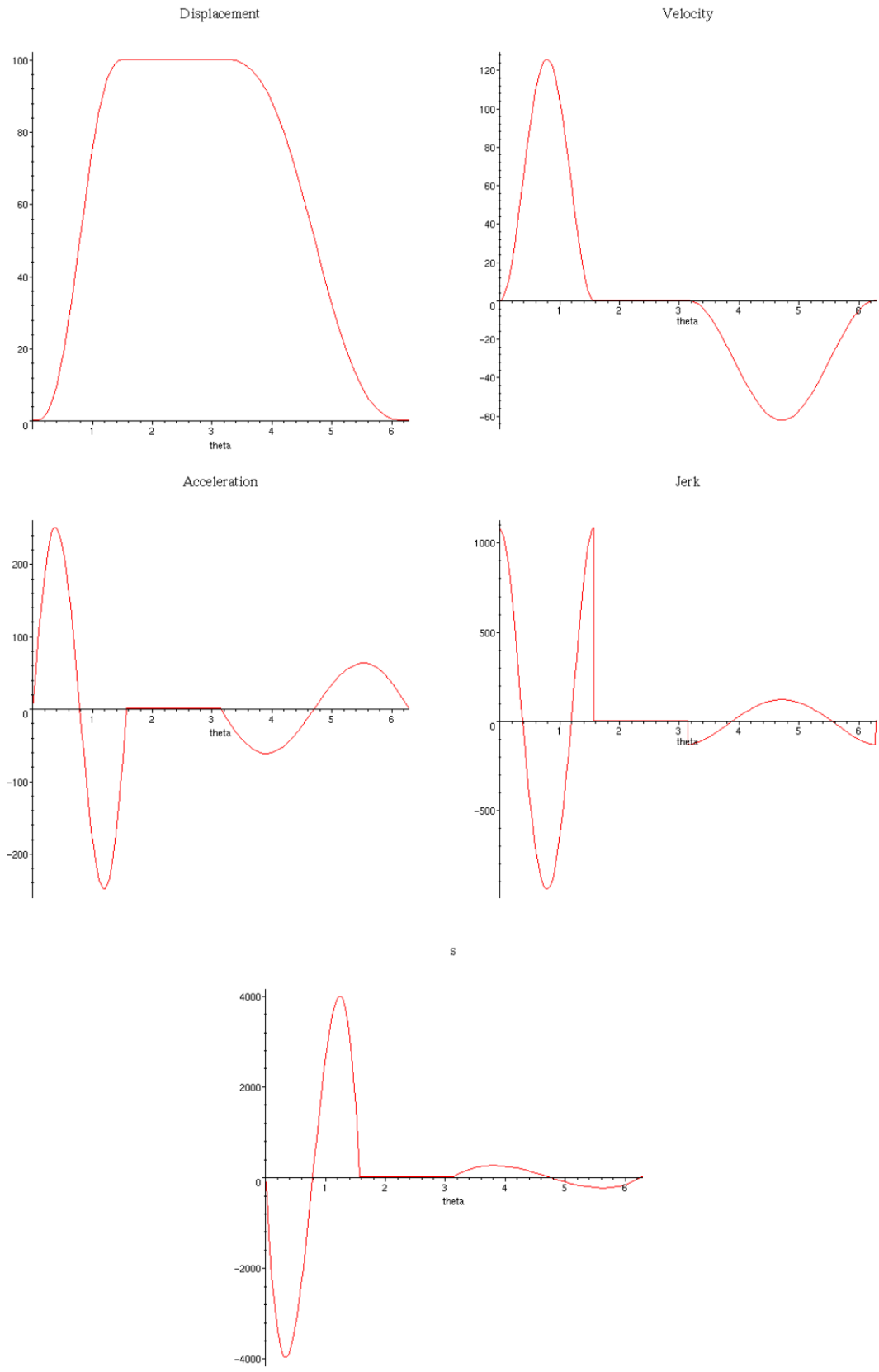
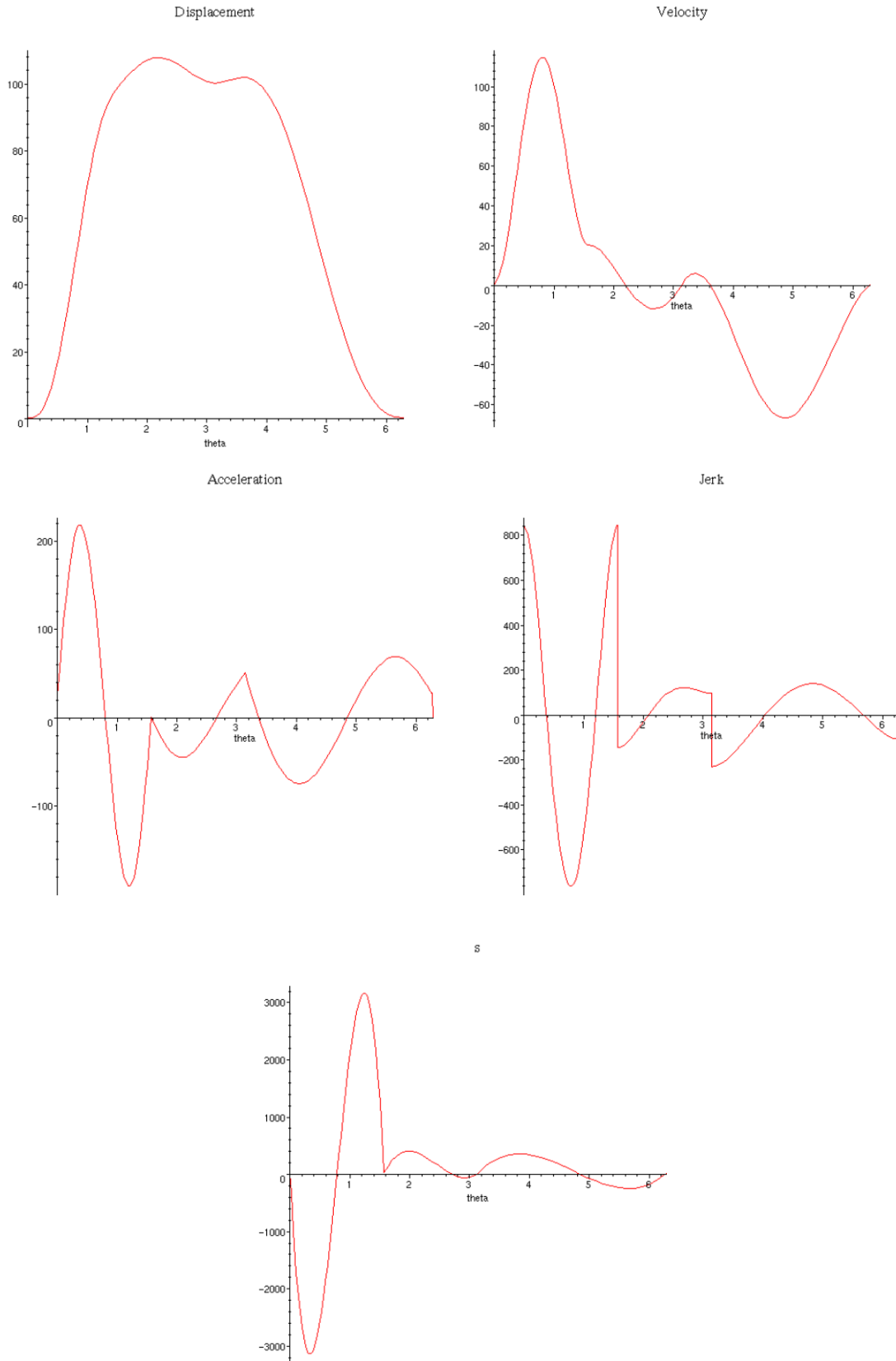Figure 1. Displacement derivatives of cam profile using piecewise polynomials.

Figure 2. Displacement derivatives of profile with incorrect dwell boundary conditions

2.5   Dwells

The use of Piecewise Polynomials is ideal for computer aided cam design, as the polynomial function is calculated by a mathematical package leaving the designer the only task of specifying the boundary values. However, a drawback is that it cannot be utilised when the cam profile includes dwells. When a dwell is defined between two breakpoints, the displacement function is reduced to $y_i = Disp_i$, and its derivatives $\dot{y}_i = 0$, $\ddot{y}_i = 0$, $\dddot{y}_i = 0$,…. In this case, if treated as the eighth order polynomial of equation (1), $b_{i1} = Disp_i$; $b_{ix} = 0$ $x = 2,...,n-1$. Wang's optimisation method, as it will be explained later in this paper, fails to recognise this as a special case, leaving all coefficients undefined. The result is that while the method acknowledges the endpoint boundary condition and the continuity conditions, it allows the profile inside the endpoints to vary. That is, $b_{ix} \neq 0$ for $x = 2,...,n-1$. Instead of a dwell, the method could result in a bell shaped profile maintaining the start and end-points at the value required for the dwell. In order to avoid this, all boundary conditions referring to the adjoining breakpoints of a dwell must be specified as zero, as shown in the worked example above.

Following the same method as before with non-zero boundary conditions on the displacement derivatives of a dwell period, the displacement function can, again, be expressed as a set of Piecewise Polynomials. This function and its derivatives are plotted in Figure 2. It can easily be seen that the dwell portion no longer exists between the 2nd and 3rd breakpoints.

3.   Optimisation

Piecewise Polynomials are very useful especially when the designer knows the boundary values required for each breakpoint and each of the derivatives used. This is not necessarily the case in all applications. Many practical problems only have specific requirements for the follower displacement values, while others might have a requirement for velocity at the midpoint of a given segment of the profile, but not a displacement requirement at the same point. In his shape optimisation method, Wang proposes a procedure by which the jerk is 'optimised' (i.e. minimised) to avoid noise, vibration and possible structural damage using the unspecified boundary conditions as variables.

3.1   Procedure

The method is based on the assumption that the designer does not specify a value for all the boundary conditions, $Disp_1...Disp_{n-1}, Vel_1...Vel_{n-1},...,s_1...s_{n-1}$, and that jerk continuity has not been used as one of the boundary and continuity conditions to solve the polynomial coefficients. The latter assumption is not given by Wang; however it is important if the method is to be applied to any other type of optimisation. In general, it should be stated that the function to be optimised cannot be used to generate any of the equations that are used to form linear system Eq. (14), described in the previous section. This is because the equation used in the optimisation will be added later to the set of equations of the linear system. Therefore, if it is already present, the system will become under constrained and it would be impossible to solve.

Case Study 2 will be developed in parallel with the optimisation method explanation. The simplest cam profile will be used: a single rise-and-return profile with no dwells and amplitude of 100mm. In order to apply the optimisation method, only some of the boundary conditions must be specified. Table 2 summarises the application's requirements.

Steps 1 through to 6 of the Piecewise Polynomial method explained above are carried out for the new profile. In this case $n = 3$; and four displacement derivatives and continuity conditions will be used, making $k = 8$. After step 6, all the polynomial coefficients can be expressed in terms of the boundary conditions.

Table 2. Optimised Cam Application Requirements (Case Study #2)

Breakpoint No.	Cam Angle β (degrees)	Displacement (mm)	Velocity (mm/rad)	Acceleration (mm/rad^2)	S (mm/rad^4)
1	0°	0	0	-	-
2	180°	100	0	-	-
3	260°	0	0	-	-

Let $\bar{X}_S$ be defined as a vector containing all the boundary values used to define the linear system Eq. (14) for which the designer has specified a value, and $\bar{X}_D$ be defined as a vector containing the boundary conditions not specified by the designer; that is, the design variables that will enable the profiles to be modified to optimise a function. In this case study: $\bar{X}_S = [Disp_1, Vel_1, Disp_2, Vel_2] = [0,0,100,0]$ and $\bar{X}_D = [Accel_1, s_1, Accel_2, s_2]$ (values not defined in Table 2). Note that the boundary conditions for the last breakpoint are not used, as they are always the same as those of the first one.

Let $\bar{X}$ be the vector produced when concatenating the previous two: $\bar{X} = [\bar{X}_S | \bar{X}_D]^T$. $\bar{X}$ should contain all the boundary values used in linear system Eq. (14) $Disp_1 \ldots Disp_{n-1}, Vel_1 \ldots Vel_{n-1}, \ldots, s_1 \ldots s_{n-1}$ to solve the polynomial coefficients. (Note that these do not include $Jerk_1 \ldots Jerk_{n-1}$ because jerk could not be included in the initial linear system)

Finally, let q be the number of elements in $\bar{X}$ and m be the number of elements in $\bar{X}_S$. The number of elements in $\bar{X}_D$ is $(q - m)$. In the example considered, $q = 8$, $m = 4$.

On the other hand, by imposing jerk continuity onto the polynomial function, $n - 1$ new equations are derived;

$$\begin{cases} 6b_{1,4} + 75.398b_{1,5} + 592.176b_{1,6} + 3720.753b_{1,7} + 20455.909b_{1,8} = 6b_{2,4} \\ 6b_{2,4} + 75.398b_{2,5} + 592.176b_{2,6} + 3720.753b_{2,7} + 20455.909b_{2,8} = 6b_{1,4} \end{cases} \quad (12)$$

Substituting the equations for the polynomial coefficients found earlier, two equations in terms of the boundary conditions are derived. Eq. (12) can then be re-written as another linear system;

$$[\bar{A}] \cdot \bar{X} = 0$$
$$(136)$$

where $[\bar{A}]$ is a fully defined $(n-1) \times q$ matrix. In the example $[\bar{A}]$ becomes the following 2x8 matrix;

$$[\bar{A}] = \begin{matrix} -2.709 & 2.709 & -0.36 \; 10^{-7} & 0.20 \; 10^{-8} & -1.273 & 4.456 & 0.105 & 0.419 \\ 2.709 & -2.709 & 0.38 \; 10^{-7} & -0.76 \; 10^{-7} & 4.456 & -1.273 & 0.419 & 0.105 \end{matrix}$$

Equation (16) can also be written as;

$$\left[\bar{\bar{A}}_S\middle|\bar{\bar{A}}_D\right]\cdot\begin{bmatrix}\bar{X}_S\\\bar{X}_D\end{bmatrix}=0\Rightarrow\left[\bar{\bar{A}}_D\right]\cdot\left[\bar{X}_D\right]=\left[\bar{B}\right]$$ (147)

where $\left[\bar{\bar{A}}_D\right]$ is a $(n-1)\times(cjvec-m)$ submatrix,

$$\left[\bar{B}\right]\equiv-\left[\bar{\bar{A}}_S\right]\cdot\left[\bar{X}_S\right], \text{ and}$$

$$\left[\bar{\bar{A}}_S\right] \text{ is a } (n-2)\times m \text{ sub-matrix.}$$

The right hand side of Eq. (147) represents a new linear system where $\bar{X}_D$ are the unknowns, since $\bar{X}_S$, $\left[\bar{\bar{A}}_S\right]$ and $\left[\bar{\bar{A}}_D\right]$ are specified. However, the system cannot be solved immediately, as $\left[\bar{\bar{A}}_D\right]$ is not a square matrix. The system should be under-constrained, that is, $(n-1)<(q-m)$. The additional constraints will be given by equations derived from the function being optimised.

$\left[\bar{\bar{A}}_D\right]$ can be further partitioned after applying the pivoting Gaussian elimination to the linear system in Eq. (147) in rows. The linear system can be re-written as:

$$\left[\bar{\bar{A}}_{DU}\middle|\bar{\bar{A}}_{DV}\right]\cdot\begin{bmatrix}\bar{X}_{DU}\\\bar{X}_{DV}\end{bmatrix}=\left[\bar{\bar{B}}\right]$$ (18)

where $\left[\bar{\bar{A}}_{DU}\right]$ is the $(n-2)\times(n-2)$ upper triangular matrix.

$\bar{X}_{DU}$ are the dependent design variables,

$\bar{X}_{DV}$ are the independent design variables.

$$\left[\bar{\bar{B}}\right]\equiv\left[\bar{\bar{P}}\right]\cdot\left(-\left[\bar{\bar{A}}_S\right]\cdot\left[\bar{X}_S\right]\right).$$

$\left[\bar{\bar{P}}\right]$ is the matrix containing the row interchange information from the Gaussian elimination.

After partitioning and Gaussian elimination, $\left[\bar{\bar{A}}_{DU}\right],\left[\bar{\bar{A}}_{DV}\right]$ and $\left[\bar{\bar{B}}\right]$ can be calculated explicitly. Rearranging Eq. (18), $\bar{X}_{DU}$ can be expressed as;

$$\bar{X}_{DU}=\left[\bar{\bar{G}}\right]\cdot\bar{X}_{DV}+\left[\bar{B}_U\right]$$ (19)

where $$\left[\bar{\bar{G}}\right]\equiv-\left[\bar{\bar{A}}_{DU}\right]^{-1}\left[\bar{\bar{A}}_{DV}\right],$$

$$\left[\bar{B}_U\right]\equiv\left[\bar{\bar{A}}_{DU}\right]^{-1}\cdot\left[\bar{\bar{B}}\right]$$

which can also be expressed in terms of $\bar{X}_D$, if the Gaussian elimination is done with row interchanges only;

$$\bar{X}_D^T = \begin{bmatrix} [\bar{G}] \\ [I] \end{bmatrix} \cdot \bar{X}_{DV}^T + \begin{bmatrix} [\bar{B}_U] \\ [0] \end{bmatrix} \tag{20}$$

The design variables contained in $\bar{X}_{DU}$ and $\bar{X}_{DV}$ have been called dependent and independent design variables respectively because, at this stage, those in $\bar{X}_{DV}$ can be assigned arbitrarily while those in $\bar{X}_{DU}$ have to satisfy Eq.(19).

The additional constraints will be given by equations derived from the function being optimised. Wang used the Jerk as his optimising function and expressed the total jerk throughout the cam profile in the form of the following equation:

$$J_{TOTAL} = \sum_{i=1}^{n-1} \int_{\beta_i}^{\beta_{i+1}} [\dddot{y}_i(\theta)]^2 \, d\theta \tag{21}$$

Wang and Yang [5] uses an approximation of Eq. (21) in which the integral is replaced by a sum over l segments:

$$J_{TOTAL} = \sum_{i=1}^{n-1} \sum_{j=1}^{l} [\dddot{y}_i(\theta)]^2$$

This is not necessary when using Maple Mathematics

Equation (21) is minimised when

$$\frac{\partial J_{TOTAL}}{\partial \bar{X}} = 0 \tag{22}$$

This is only true, however when $1 \le \dddot{y}_i(\theta) \le -1$. The most accurate way to express the total jerk is

$$J_{TOTAL} = \sum_{i=1}^{n-1} \int_{\beta_i}^{\beta_{i+1}} |\dddot{y}_i(\theta)| \, d\theta \tag{23}$$

However, Wang does not use Eq. (23) because it does not suit the Quadratic Programming method used later in the paper. This is because absolute value cannot be transformed into a linear system or expressed as a matrix, while the square value of a function can be.

$\dddot{y}_i$ can be expressed in vector form as $\dddot{y}_i = \bar{e}_i \cdot \bar{X}^T$, where $\bar{e}_i$ will have non-zero elements only when they relate to the ith and the $(i+1)^{th}$ boundary conditions in $\bar{X}$. $\dddot{y}_i^{\,2}$ can be expanded using a positive definite symmetric matrix as follows;

$$\dddot{y}_i^{\,2} = (\bar{e}_i \cdot \bar{X}^T)^2 = \bar{X} \cdot [\bar{q}^i] \cdot \bar{X}^T \tag{24}$$

where

$$[q_i] = \int_{\beta_i}^{\beta_{i+1}} e_i \cdot e_i^T \, d\theta$$

Substituting Eq. (24) into Eq. (21), the total jerk becomes;

$$J_{TOTAL} = \sum_{i=1}^{n-1} \bar{X} \cdot [\bar{q}^i] \cdot \bar{X}^T = \bar{X} \cdot \left(\sum_{i=1}^{n-1} [\bar{q}^i] \right) \cdot \bar{X}^T = \bar{X} \cdot [\bar{Q}] \cdot \bar{X}^T \qquad (25)$$

where $[\bar{Q}] \equiv \sum_{i=1}^{n-1} [\bar{q}^i]$, is a fully specified block diagonal matrix.

The expression for J_{TOTAL} can be transformed into a partition form to discriminate between the specified and the unspecified portions of $\bar{X}$:

$$J_{TOTAL} = [\bar{X}_S | \bar{X}_D] \cdot \begin{bmatrix} [\bar{Q}_{11}] & [\bar{Q}_{12}] \\ [\bar{Q}_{21}] & [\bar{Q}_{22}] \end{bmatrix} \cdot \begin{bmatrix} \bar{X}_S^T \\ \bar{X}_D^T \end{bmatrix} \qquad (26)$$

Expanding Eq. (26);

$$J_{TOTAL} = \bar{X}_S \cdot [\bar{Q}_{11}] \cdot \bar{X}_S^T + 2 \cdot \bar{X}_D \cdot [\bar{Q}_{21}] \cdot \bar{X}_S^T + \bar{X}_D \cdot [\bar{Q}_{22}] \cdot \bar{X}_D^T \qquad (27)$$

Note that $\bar{X}_S$ is known and $\bar{Q}$ can be computed. The only variable in Eq. (25) is $\bar{X}_D$. Let $\eta \equiv \bar{X}_S \cdot [\bar{Q}_{11}] \cdot \bar{X}_S^T$ and $\zeta \equiv [\bar{Q}_{21}] \cdot \bar{X}_S^T$, which can be treated as constants. Eq. (27) can be re-written as

$$J_{TOTAL} = \eta + 2 \cdot \bar{X}_D \cdot \zeta + \bar{X}_D \cdot [\bar{Q}_{22}] \cdot \bar{X}_D^T \qquad (28)$$

Substituting Eq. (20) into Eq. (28),

$$J_{TOTAL} = \eta + 2 \cdot \left(\begin{bmatrix} [\bar{G}] \\ [I] \end{bmatrix} \cdot \bar{X}_{DV}^T + \begin{bmatrix} [\bar{B}_U] \\ [0] \end{bmatrix} \right)^T \cdot \zeta + \bar{X}_D \cdot [\bar{Q}_{22}] \cdot \left(\begin{bmatrix} [\bar{G}] \\ [I] \end{bmatrix} \cdot \bar{X}_{DV}^T + \begin{bmatrix} [\bar{B}_U] \\ [0] \end{bmatrix} \right) \qquad (159)$$

This can be expanded and then simplified and rearranged in terms of $\bar{X}_{DV}$, applying the theorem that that the transpose of a product of matrices is the product of the matrices in reverse order, Kyrala [7].

$$J_{TOTAL} = \eta_v + \bar{B}_{UV} + 2 \cdot \bar{X}_{DV} \cdot \zeta_V + \bar{X}_{DV} \cdot [\bar{Q}_V] \cdot \bar{X}_{DV}^T \qquad (30)$$

where $\zeta_V \equiv \begin{bmatrix} [\bar{G}] \\ [I] \end{bmatrix}^T \cdot \left([\bar{Q}_{22}] \cdot \begin{bmatrix} [\bar{B}_U] \\ [0] \end{bmatrix} + \zeta \right)$, $\quad \eta_v \equiv \eta + 2 \cdot \begin{bmatrix} [\bar{B}_U] \\ [0] \end{bmatrix}^T \cdot \zeta$, $\quad [\bar{Q}_V] \equiv \begin{bmatrix} [\bar{G}] \\ [I] \end{bmatrix}^T \cdot [\bar{Q}_{22}] \cdot \begin{bmatrix} [\bar{G}] \\ [I] \end{bmatrix}$ and

$$\bar{B}_{UV} \equiv \begin{bmatrix} [\bar{B}_U] \\ [0] \end{bmatrix}^T \cdot [\bar{Q}_{22}] \cdot \begin{bmatrix} [\bar{B}_U] \\ [0] \end{bmatrix}.$$

The total jerk is minimised its derivative is equal to zero. See (22).

$$\frac{\partial J_{TOTAL}}{\partial \bar{X}_{DV}} = 2\zeta_V + 2[\bar{Q}_V] \cdot \bar{X}_{DV}^T = 0 \qquad (31)$$

Therefore, the jerk is minimised when:

$$\bar{X}_{DV}^T = -[\bar{Q}_V]^{-1} \cdot \zeta_V \qquad (32)$$

Substituting the values found for $\bar{X}_{DV}$ into Eq.(19), $\bar{X}_{DU}$ can also be calculated;

$$\bar{X}_{DU} = [\bar{G}] \cdot \bar{X}_{DV} + [\bar{B}_U]$$

That is, the unspecified boundary conditions should be set, in this example, to

$$\begin{bmatrix} X_{D1} \\ X_{D2} \\ X_{D3} \\ X_{D4} \end{bmatrix} = \begin{bmatrix} Accel_1 \\ Accel_2 \\ s_1 \\ s_2 \end{bmatrix} = \begin{bmatrix} 50.567 \\ -50.735 \\ -60.917 \\ 61.944 \end{bmatrix} \tag{33}$$

in order to obtain the minimum Jerk requirement.

Having calculated all the design variables, the cam's displacement PP functions and hence, the total jerk throughout the profile are fully defined. The total jerk can now be calculated from Eq. (21).

$$J_{TOTAL} = \sum_{i=1}^{n-1} \int_{\beta_i}^{\beta_{i+1}} \left[\ddot{y}_i(\theta) \right]^2 d\theta = 7842.71$$

Figure 3 show the displacement function and its derivatives for the case study. Notably, all the boundary conditions (design ones as well as the specified ones) are met and there is continuity of all the derivatives.

3.2   Minimum Jerk

It can be demonstrated that the jerk is at minimum by plotting the total jerk function versus each of the design variables, while maintaining the others constant at the minimum values obtained for them.

In the case study, for instance, there were four design variables; $Accel_1, Accel_2, s_1, s_2$, and it was found that the condition of minimum jerk would be satisfied when;

$$\begin{bmatrix} X_{D1} \\ X_{D2} \\ X_{D3} \\ X_{D4} \end{bmatrix} = \begin{bmatrix} Accel_1 \\ Accel_2 \\ s_1 \\ s_2 \end{bmatrix} = \begin{bmatrix} 50.567 \\ -50.735 \\ -60.917 \\ 61.944 \end{bmatrix} \tag{33}$$

Since it would be impossible to visualise the jerk as a function of four variables, a graph of the jerk function in terms of each of the design variables is constructed. The other variables are set to the values in Eq. (33). It is clearly seen in Figure 4 that the jerk minimum occurs at the values predicted.

The value of the minimum jerk calculated will vary if the specified boundary conditions change. This optimisation method will provide the minimum total jerk given the specified conditions. Should the designer of the previous example decide to specify different set of boundary conditions, the displacement functions and their derivatives will change accordingly and the value of the minimum jerk will be different.
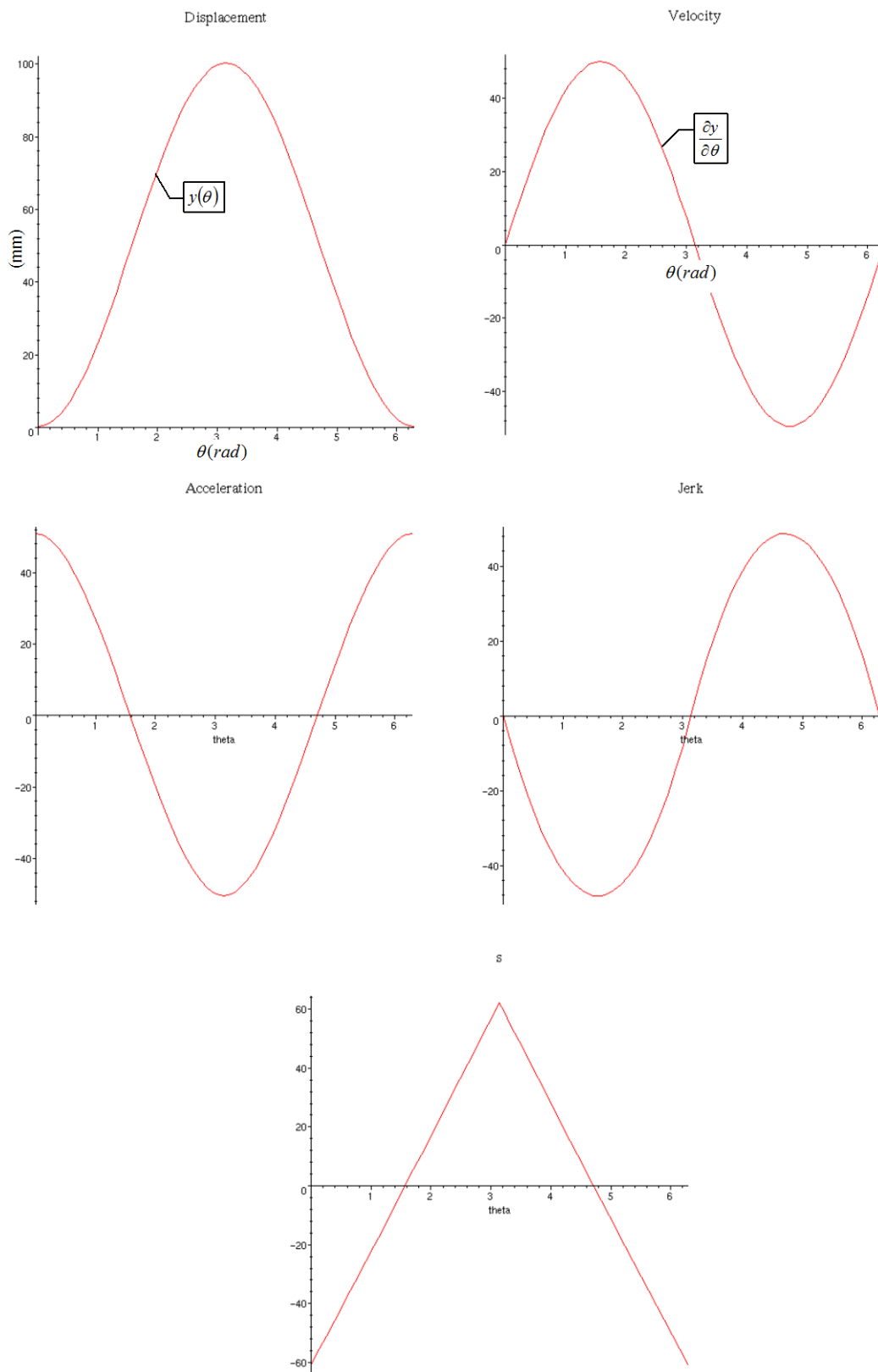
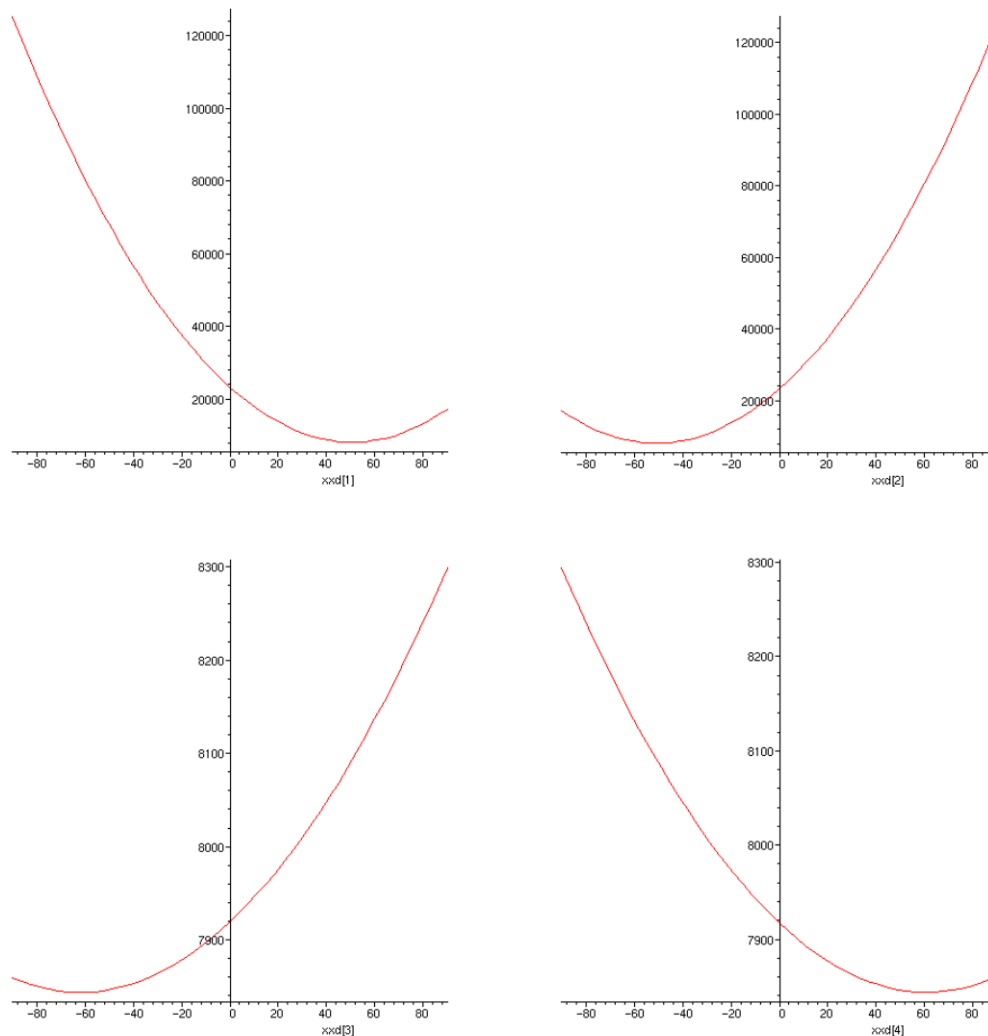Figure 3. Displacement derivatives for optimised case study

Figure 4. Proof of minimum: jerk in terms of each of the design variables.

3.3. Case Study #3: Specifying Incorrect Boundary Conditions

Table 1 shows a new set of boundary conditions. The displacement values remain unchanged, but in this case, the designer wishes to specify the acceleration values at the breakpoints instead of the velocity values. Figure 5 shows the displacement function for the cam profile and its derivatives following the same optimisation procedure described earlier. The displacement profile is clearly unpractical, although it has not violated the boundary conditions stipulated by the designer. The polynomial found for the displacement is such that the cam displacement would become negative ($y(\theta) < 0$) before reaching β_3, ie, $y(\theta) = 0$ (the boundary condition for displacement at $\beta_3 = 360°$) making the total amplitude of the movement larger than intended.

This is a similar problem to the one encountered with giving incorrect boundary conditions for dwells. The optimisation method cannot guarantee the profile where the cam angle (θ) is not a boundary condition because it is based on piecewise polynomials.
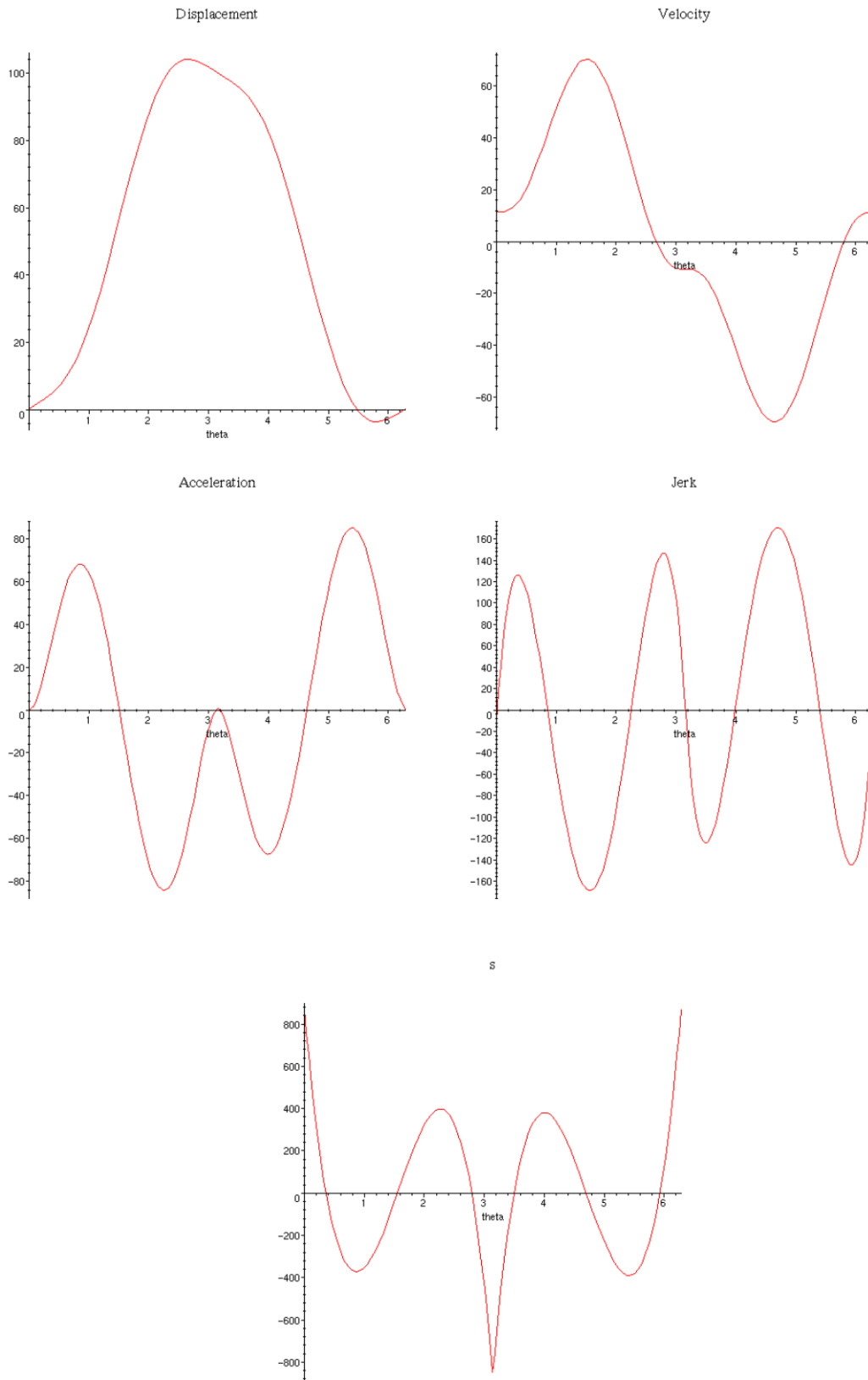
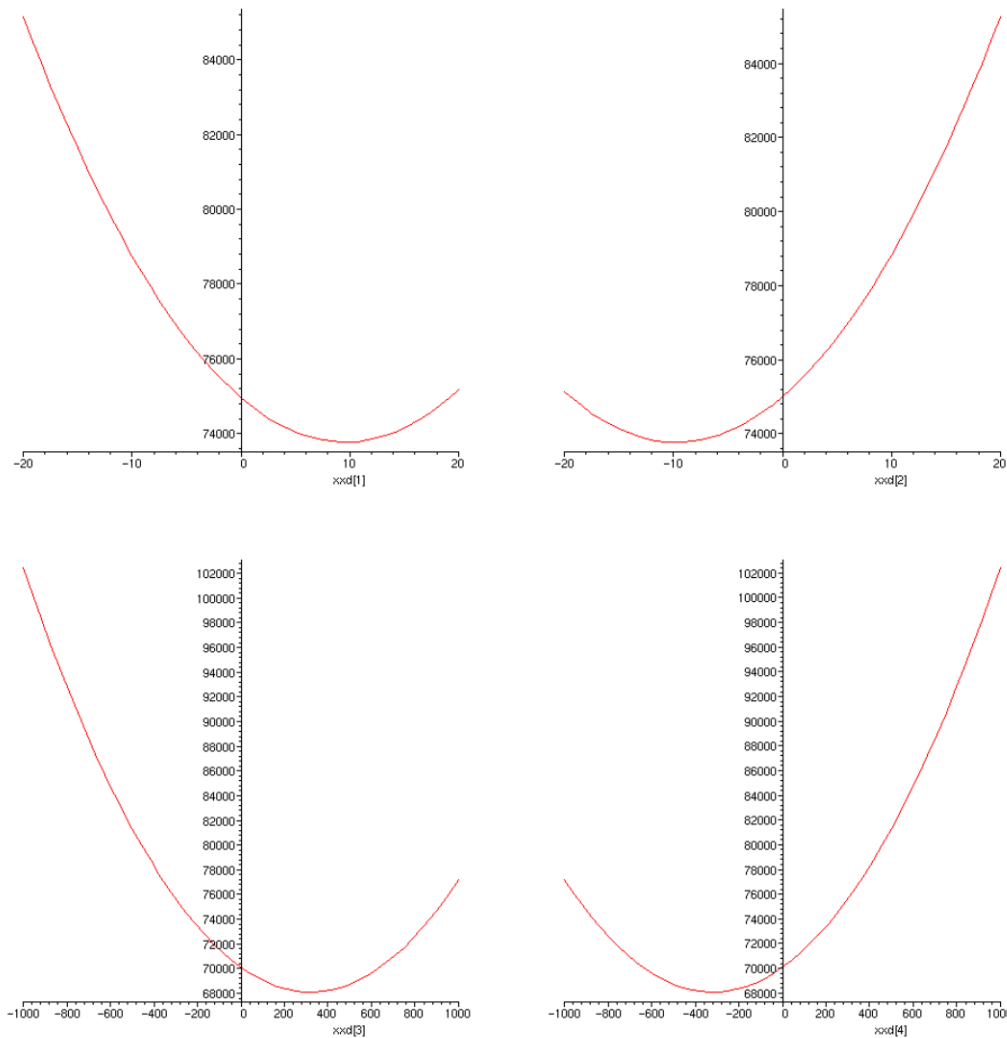Figure 5. Displacement Derivatives for Case Study #3.

Figure 6. Jerk in terms of the Design Variables for Case Study #3

Furthermore, following the same optimisation procedure, the values obtained for the design variables are

$$\begin{bmatrix} Vel_1 \\ Vel_2 \\ S_1 \\ S_2 \end{bmatrix} = \begin{bmatrix} 27.0 \\ 0.8 \\ 862.3 \\ -862.3 \end{bmatrix}$$

Figure 6 shows that the values calculated for the design variables in order to minimise the jerk do not correspond to the function's minimum.

In this case, the technique has not worked because the designer has specified the values of the boundary conditions for acceleration. The function being optimised is the integral of jerk, ie, acceleration.

$$J_{TOTAL} = \sum_{i=1}^{n-1} \int_{\beta_i}^{\beta_{i+1}} \dddot{y}_i(\theta) \cdot d\theta = \sum_{i=1}^{n-1} \left[\ddot{y}_i(\theta) \right]_{\beta_i}^{\beta_{i+1}}$$

$$= \sum_{i=1}^{n-1} \ddot{y}_i(\beta_{i+1}) - \ddot{y}_i(\beta_i) = \sum_{i=1}^{n-1} Accel_{i+1} - Accel_i$$

The linear system for which the coefficients are derived Eq. (12) can not contain the optimising function's boundary conditions. Therefore in the example given, acceleration boundary conditions cannot be specified. More generally, however, this case study demonstrates that the graphical proof of minimum will point out on any boundary conditions erroneously specified.

Table 3. Design Requirements Including Acceleration (Case Study #3)

Breakpoint No.	Cam Angle β	Displacement (mm)	Velocity	Acceleration	S
1	0°	0	-	0	-
2	180°	100	-	0	-
3	360°	0	-	0	-

4.   Conclusions

A method of synthesising a cam profile based on the use of Piecewise Polynomials together with an optimisation technique has been presented. Special cases and limitations have been discussed and illustrated, making the procedure complete and systematic for any design requirements. The implementation of the method requires a symbolic mathematics software application, as the calculations involved are lengthy. The examples shown were implemented using Maple Mathematics.

Polynomial profiles, which are defined with a number of breakpoints in the profile for which the designer can define boundary conditions, are the most suitable for optimisation because the displacement, velocity, acceleration and jerk can be controlled for any critical point in the cam profile hence, using Piecewise Polynomials, the complete cam profile can be designed as a combined linear system.

Furthermore, the optimisations technique described here enables the variables in the linear system to select the ideal combination in order to optimise a given expression.

Certain requirements for the method to work have been illustrated. Firstly, a procedure to choose the appropriate polynomial order has been illustrated. Secondly, it has been proven that dwells must be treated as special cases; all boundary conditions at the two breakpoints adjoining the dwell segment must be specified as zero. Thirdly, the linear system from which the coefficients are derived (Eq. 12) cannot contain the optimising function's boundary conditions. That is, in optimising acceleration of a given profile, the linear system cannot contain $Accel_1 \dots Accel_n$.

In addition, a method to prove the validity of the results has been explained. This proof should be carried out on any cam design to ensure that the designer has not, mistakenly, specified boundary conditions that will invalidate the result.

Acknowledgements

References

1.   Erdman A and Sandor G (1991), *Mechanism Design. Analysis and Synthesis*, New Jersey: Prentice Hall.

2.   Shigley JE and Uicker JJ (1983), *Theory of Machines and Mechanisms*, 1st Spanish Edition, Mexico: McGraw-Hill.

3.   Farouki RT and Manjunathaiah J (1998), Design of Raotational Cam Profiles with Pythagorean Hodograph Curves, *Mechanisms and Machine Theory*, Vol 33, No 6, 669-682, Elsevier Science Ltd

4   Yu Q and Lee HP (1998), Size Optimisation Of Cam Mechanisms With Translating Roller Followers, *Proceedings of the Institute of Mechanical Engineers*, Vol. 212 Part C, 381-386, IMechE.

5.   Wang LT and Yang Y (1996), Computer Aided Design of Cam Motion Programs, *Computers in Industry*, Volume 28, 151-161, Elsevier Science Ltd.

6.   Yao Y et al (2000), Motion control of Cam Mechanisms, *Mechanisms and Machine Theory*, Vol 35, No 4, 593-607, Elsevier Science Ltd.

7.   Kyrala A (1967), Theoretical Physics: *Applications of Vectors, Matrices, Tensors and Quaternions*, p 35, W.B. Saunders Company, London.