Rose-Hulman Undergraduate Mathematics Journal

Volume 7
Issue 1

Article 12

Comparison of Numerical Techniques for Euclidean Curvature

Derek Dalle
University of Minnesota, dall0081@umn.edu

COMPARISON OF NUMERICAL TECHNIQUES FOR EUCLIDEAN CURVATURE

DEREK DALLE

ABSTRACT. This paper begins with a comparison of second-order numerical approximations to Euclidean curvature, and verifies that some of the approximations are invariant to Euclidean transformations. Also, higher-order Euclidean invariant numerical techniques are developed and tested. Consideration is given to strengths and weaknesses of each algorithm.

1. INTRODUCTION

Presented here is a brief discussion of some immediate comparisons of basic numerical approximations to curvature. The key objective will be to compare two numerical schemes that are invariant to Euclidean transformations. As an experimental control, the curvature is also calculated by using basic centered finite difference techniques.

The discussion verifies that the techniques are invariant to rotations, have the correct order of convergence, and compares the two invariant schemes. Further, a new scheme whose foundations were laid by Peter Olver [3] is shown to have similar overall performance to the geometric technique of Calabi, *et. al* [2]. After each technique is given a brief description, the paper shows data for individual points in several circumstances and concludes each section by showing the performance of the techniques on some region of the curve. Finally, an algorithm for generating higher-order invariant techniques by extending Olver's techniques is discussed and tested. The performance of the higher-order invariant techniques is shown to exceed that of any of the previous numerical approximations.

2. FINITE DIFFERENCE TECHNIQUES

Given a function in one variable $y = u(x)$, the analytical equation for curvature is given in (2.1). Alternatively, if a curve is parameterized by $\mathbf{r}(t) = (x(t), y(t))$, the equation for curvature is given in (2.2).

$$\kappa(x) = \frac{u_{xx}}{(1+u_x^2)^{3/2}} \tag{2.1}$$

$$\kappa(t) = \frac{x_t y_{tt} - y_t x_{tt}}{(x_t^2 + y_t^2)^{3/2}} \tag{2.2}$$

Thus the natural numerical approximation to $\kappa(x)$ is to find finite-difference approximations to the derivatives of $u(x)$ and simply use those in (2.1). In order to maintain a centered finite-difference approach, second- and fourth-order finite differences will be used. The formulas for equal-spaced finite differences are

$$u_x(x_1) = \frac{u(x_2) - u(x_1)}{2h} + O(2)\,, \tag{2.3}$$

$$u_{xx}(x_1) = \frac{u(x_2) - 2u(x_1) + u(x_0)}{h^2} + O(2), \tag{2.4}$$

$$u_x(x_2) = \frac{u_0 - 8u_1 - 8u_3 + u_4}{12h} + O(4) \text{ and} \tag{2.5}$$

$$u_{xx}(x_2) = \frac{u_0 - 16u_1 + 30u_2 - 16u_3 + u_4}{-12h^2} + O(4). \tag{2.6}$$

The term $O(n)$ means that if the step size decreases by a factor of 10, the error will decrease by approximately 10^n. Using these formulas, one can find approximations for points equally spaced along the x-axis. With very similar formulas, one can approximate the curvature of a parametric curve when the points are equally spaced with respect to the parameter, t. However, one would also like to see how well these finite difference techniques respond to rotations in the xu-plane.

To do this, proceed in the following manner. First, pick a set of 5 points by using equally spaced x-values. After computing the curvature using these points and formulas (2.3)-(2.6), map each point into the yv-plane by multiplying each point by a rotation matrix and compute the curvature again. Notice that although the spatial distance between points will not change, the points may no longer have equally spaced y-values. This is illustrated in the following figures.

Figure 2.1

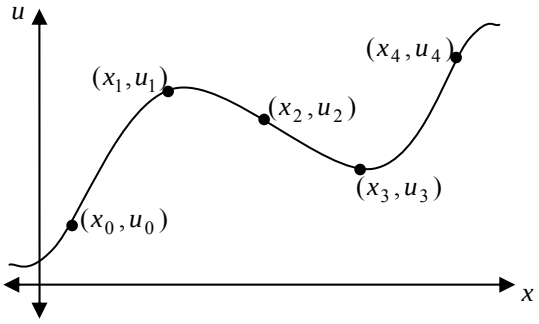
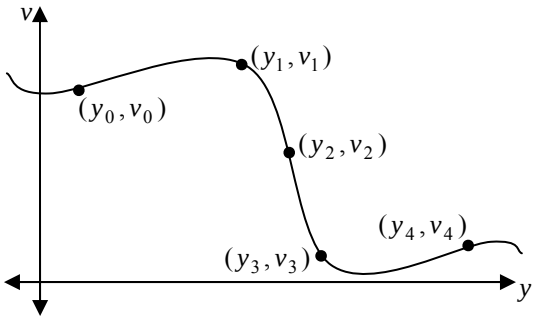


Figure 2.2



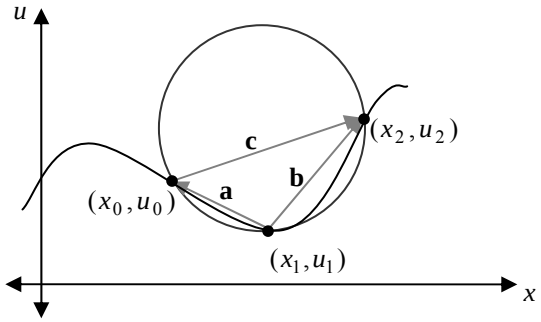
$$\begin{bmatrix} y_i \\ v_i \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x_i \\ u_i \end{bmatrix} \tag{2.7}$$

Notice that $y_1 - y_0 > y_2 - y_1$. In order to compute the curvature in the *yv*-plane using finite difference techniques, one needs formulas that do not rely any kind of equal spacing. This can be done by solving a system of equations given by equating terms of the Taylor series at each point. These formulas are not difficult to derive, especially with a computer algebra system, but they are long enough not to be presented here. See [5], for example, for a more thorough description.

3. GEOMETRIC TECHNIQUE

One clever technique is to match circles to three points [2]. Since the curvature is also defined as the reciprocal of the radius of the osculating circle, and three points uniquely define a circle, one can see how this method may lead to an approximation of curvature.

Figure 3.1



Here the radius of the circle is exactly four times the area of the triangle divided by the product of the distances between points [2]. However, one can see from (2.1) that the curvature may be negative or positive, depending entirely on the second derivative. Getting the sign correct for this numerical scheme may become important at some point, so one wishes to take care of it before it poses a problem. To do so, one simply attaches a sign to the area of the triangle. The equations below refer to the signed area of the triangle as Δ and also assume that $a = |\mathbf{a}|$, *etc*.

$$\kappa = \frac{1}{R} = 4\frac{\Delta}{abc} \tag{3.1}$$

$$\mathbf{a} = (x_0 - x_1, y_0 - y_1) \tag{3.2}$$

$$\mathbf{b} = (x_2 - x_1, y_2 - y_1) \tag{3.3}$$

$$\mathbf{c} = (x_2 - x_0, y_2 - y_0) \tag{3.4}$$

$$\Delta = \frac{1}{2}\det[\mathbf{b} \quad \mathbf{a}] \tag{3.5}$$

Because the calculations all rely on three points to calculate curvature, one may conjecture that this is a second-order numerical scheme. However, this is only the case if the points have equal spatial distance. This is not necessarily the case here since the points have equal horizontal spacing as opposed to equal spatial distances.

Since this method only depends on the vectors between the points, one expects that it will be invariant to the rotations illustrated in Figs. 1 and 2.

4. MOVING FRAMES TECHNIQUE

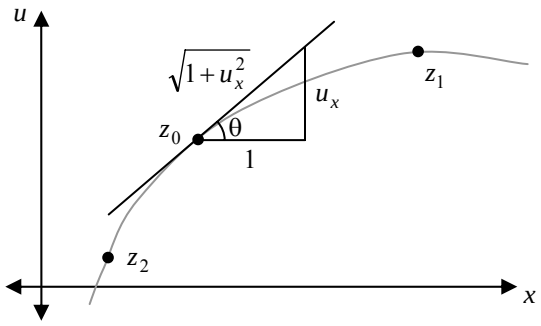
Another invariant technique is based on the concept of moving frames [3]. Traditionally the moving frames technique is described by writing the unit tangent and normal vectors

$$\mathbf{T} = \begin{bmatrix} 1/\sqrt{1+u_x^2} \\ u_x/\sqrt{1+u_x^2} \end{bmatrix}, \qquad \mathbf{N} = \begin{bmatrix} -u_x/\sqrt{1+u_x^2} \\ 1/\sqrt{1+u_x^2} \end{bmatrix},$$

and then noting that $d\mathbf{T}/ds = \kappa\mathbf{N}$ and $d\mathbf{N}/ds = -\kappa\mathbf{T}$. This description would be equivalent in the case of Euclidean curvature, but Olver's extension of moving frames allows one to find differential invariants for much more diverse groups of transformations. See Section 7 or [3] for further explanation of the new algebraic technique.

For the Euclidean curvature the general idea is that if the first derivative of u is very close to zero, then the curvature can be found by simply approximating the second derivative. Of course, one is also interested in points where the first derivative is not zero. However, since the curvature itself is invariant to rotations, one can rotate the graph about a point until the first derivative there is zero without changing the value of the curvature.

Figure 4.1



In Fig. 4.1, one can first translate the graph so that z_0 is at the origin and then rotate the graph clockwise by θ. Then $u_x = 0$, and $\kappa = u_{xx}$. For clarity, define a mapping from the xu-plane to a different plane, the HK-plane.

$$z_k = (x_k, u_k) \qquad I_k = (H_k, K_k) \tag{4.1}$$

$$\iota : z_k \mapsto I_k \tag{4.2}$$

$$\iota(z) = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}(z - z_0) \tag{4.3}$$

However, using Fig. 4.1, θ can be written in terms of u_x.

$$\cos\theta = \frac{1}{\sqrt{1+u_x^2}} \tag{4.4}$$

$$\sin\theta = \frac{u_x}{\sqrt{1+u_x^2}} \tag{4.5}$$

$$I_k = \frac{1}{\sqrt{1+u_x^2}}\begin{bmatrix} 1 & u_x \\ -u_x & 1 \end{bmatrix}\begin{bmatrix} x_k - x_0 \\ u_k - u_0 \end{bmatrix} \tag{4.6}$$

$$H_k = \frac{(x_k - x_0) + (u_k - u_0)u_x}{\sqrt{1+u_x^2}} \tag{4.7}$$

$$K_k = \frac{(u_k - u_0) - (x_k - x_0)u_x}{\sqrt{1+u_x^2}} \tag{4.8}$$

Once the points in the *HK*-plane have been obtained, one simply executes a finite difference scheme on the I_k points. However, to do the transformation of ι, one must first have an approximation for u_x. A convenient way to express these finite difference approximations is to use a set of divided difference notations. These can be defined recursively as follows.

$$[z_0 z_1] = \frac{u_1 - u_0}{x_1 - x_0} \tag{4.9}$$

$$[z_0 z_1 z_2] = \frac{[z_0 z_2] - [z_0 z_1]}{x_2 - x_1} \tag{4.10}$$

$$[z_0 \ldots z_n] = \frac{[z_0 \ldots z_{n-2} z_n] - [z_0 \ldots z_{n-1}]}{x_n - x_{n-1}} \tag{4.11}$$

$$u_0^{(n)} = n![z_0 \ldots z_n] + O(n) \tag{4.12}$$

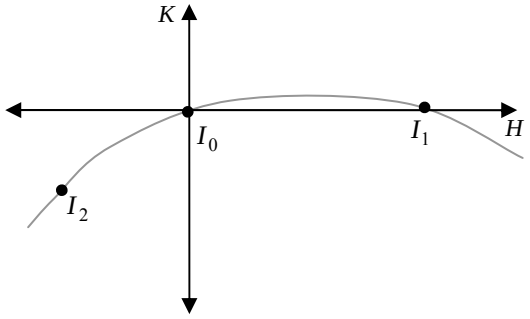
Hence the simplest approximation is the one that uses $u_x = [z_0 z_1]$ and $\kappa = 2[I_0 I_1 I_2]$. A consequence of the translation ι is that $I_0 = (0,0)$; however, since $u_x = [z_0 z_1]$, another result is that $K_1 = 0$. This, in turn, implies that $[I_0 I_1] = 0$.

$$\kappa = 2\frac{[I_0 I_2] - [I_0 I_1]}{H_2 - H_1} = 2\frac{K_2}{H_2(H_2 - H_1)}$$

$$\kappa = \frac{2[z_0 z_1 z_2]\sqrt{1+[z_0 z_1]^2}}{(1+[z_0 z_1][z_1 z_2])(1+[z_0 z_1][z_0 z_2])} \tag{4.13}$$

This is precisely the formula given by Olver. However, Fig. 4.2 shows how that the first derivative at I_0 is not exactly zero. In that figure, a transformation ι is applied, but instead of using the analytical value for u_x, the first-order approximations are used.

Figure 4.2



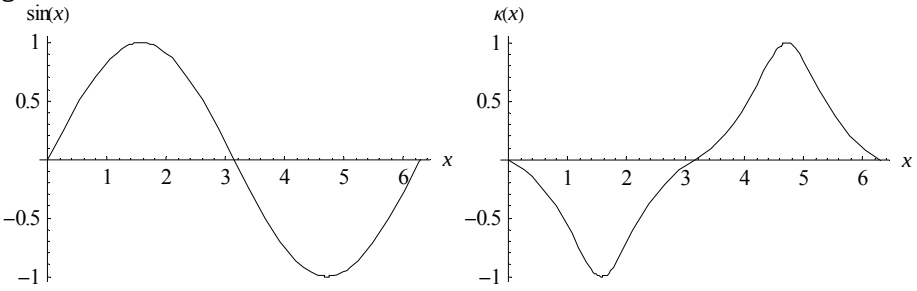
Notice that the first derivative at I_0 is clearly nonzero in Fig. 4.2. This suggests that a better approximation of u_x might lead to a better final approximation to κ. This is discussed in the section about higher-order methods.

5. Analysis at Individual Points

One experimental method to compare the assorted numerical schemes is to consider a few points of a curve at which the behavior is indicative of most properties of curves. For instance, one should consider a set of points that includes a point with high curvature and low slope, a point with high curvature and high slope, a point with low curvature and high slope, a point with low curvature and low slope and a point with extremely high curvature. In addition one may want to consider such points on different curves in case one of the numerical approximations is especially well-suited to a given curve, *e.g.* a circle for the geometric scheme or a polynomial for the finite difference schemes.

First consider one of the simplest functions of all, the sine function. Fig. 5.1 shows the analytical value of the curvature of the sine function.

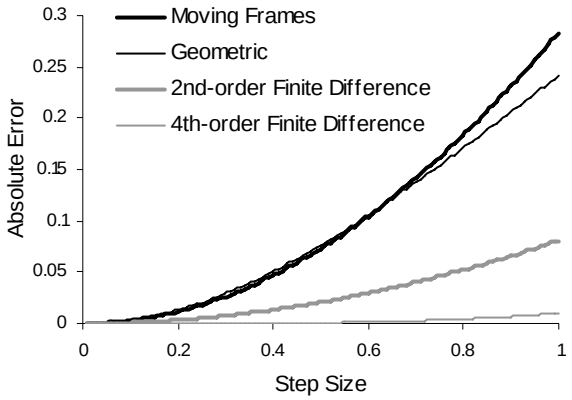
Figure 5.1



Since the sine function is odd, all four methods get the exact curvature at $x = 0$ ($\kappa(0) = 0$). Therefore the analysis will begin at $x = \pi/2$. One sees that $\kappa(\pi/2) = -1$.

The next figure illustrates the performance of the various techniques of the function sin x at the point $x = \pi/2$. The finite difference techniques worked very well as expected. Notice that the two invariant techniques performed almost identically as well, with the moving frame technique performing slightly better for step sizes less than 0.5.

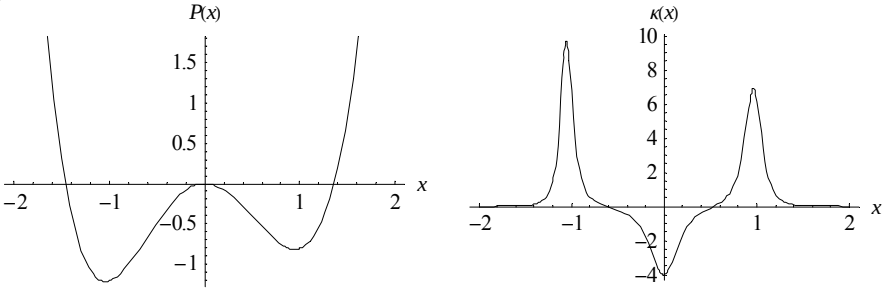
Figure 5.2



The next curve considered is a polynomial with rather dynamic behavior.

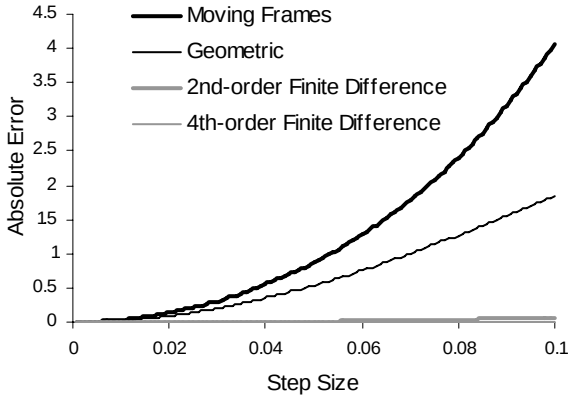
$$P(x) = -\tfrac{1}{10}x^5 + x^4 + \tfrac{3}{10}x^3 - 2x^2 \tag{5.1}$$

Figure 5.3



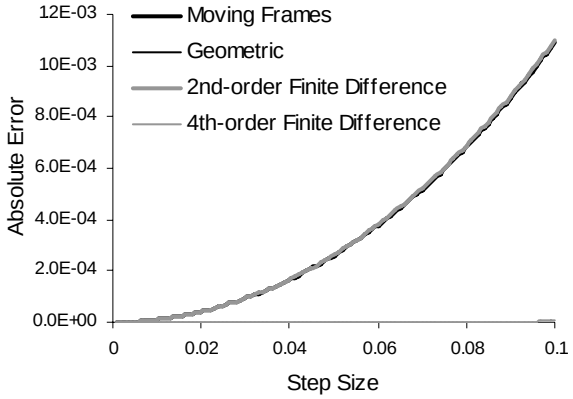
The point κ(–1.05) = 9.625 will be analyzed first. There the curvature is very high, but the slope is near zero. The geometric scheme produced better results than the technique derived from moving frames, but both appear to show the same order of convergence.

Figure 5.4



In addition, the point $\kappa(0.5) = -0.079$ is analyzed. In this case, three of the schemes performed almost identically.

Figure 5.5

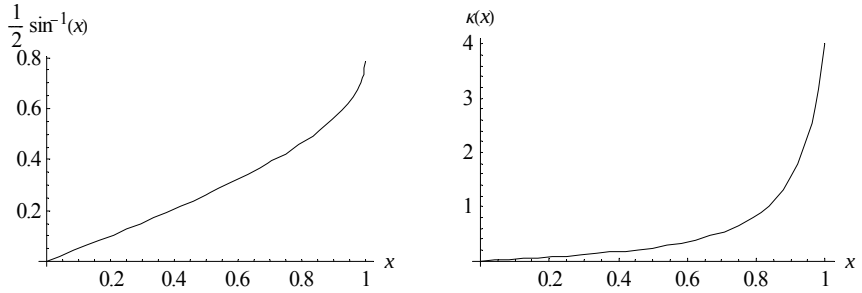


The point that remains to be analyzed is the point with high slope and high curvature. The function

$$u(x) = \tfrac{1}{2}\sin^{-1} x \tag{5.2}$$

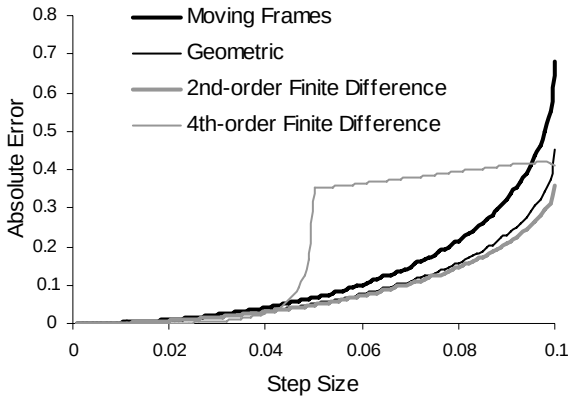
has such a point. This is simply the inverse of a curve with a point of high curvature and low slope.

Figure 5.6



To obtain the highest possible slope, $\kappa(0.9) = 1.542$ is analyzed. Notice the unusual cusp in the 4th-order technique. This is due to the fact that when the step size is greater than 0.05, this technique must draw a second point where $x > 1$. The moving frames technique performed slightly worse than the other methods, but again all methods had similar performance, and both invariant schemes seem to show second-order convergence.

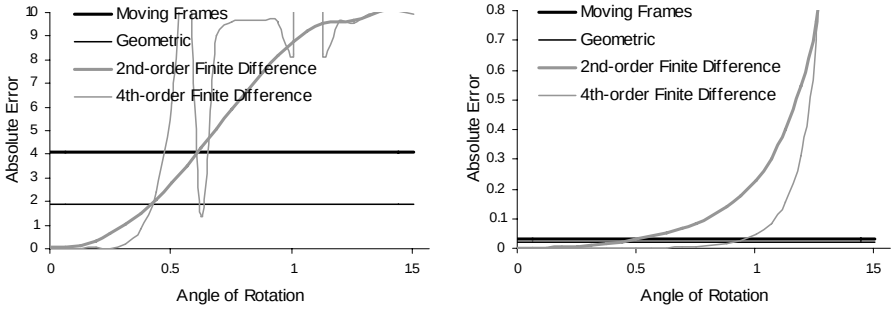
Figure 5.7



Before making final conclusions, one might wish confirm that the geometric and moving-frame techniques are indeed invariant to rotations. This was the case in all of the points considered, so the data will only be presented for a few of the most extreme cases.

Below are two graphs illustrating the responses to rotation using the polynomial above, $P(x)$, at the point $x = -1.05$. The angle of rotation is measured in radians. The points are rotated according to (2.7). In the first chart, a large step size ($h = 0.1$) is used while the second uses a small step size ($h = 0.01$). The invariant schemes retain errors, but the errors do not change.

Figure 5.8



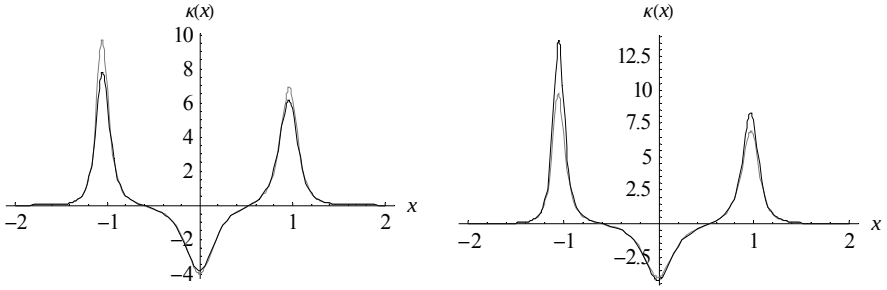
As a final measure of the success of these various methods, one may want to see how they perform over an entire interval. Below are several graphs depicting the analytical and predicted values for curvature using the various techniques. This will allow the reader to observe how each method behaves over a various range of curvatures and slopes. These graphs also demonstrate how the various numerical schemes react to sudden changes in the nature of the curve.

Somewhat surprisingly, the finite difference methods appear to be the most accurate methods at points with higher curvature. This may be due to the way in which points were selected equally along the x-axis. As a result of this point selection, the points with high curvature are the points where the distance between points changes most rapidly. The geometric scheme is known to lose an order of convergence when these distances are unequal.

What follows is two graphs comparing the predicted curvatures of $P(x)$ as defined in (5.1) to the analytical curvature of that function. Both graphs use a step size of $h = 0.1$ with no angle of rotation. In both cases, the analytical curvature is shown by the gray curve, while the black curve is the one predicted by the numerical scheme.

Since $P(x)$ is a polynomial, the finite difference scheme is very accurate, and its graph is not shown here. The left and right graphs are the geometric and moving frame techniques, respectively. While both show errors at the points of high curvature, notice that the geometric technique underestimates the curvature while the moving frames technique overestimates. This is the first time a fundamental difference in the two techniques' performances has been demonstrated.

Figure 5.9

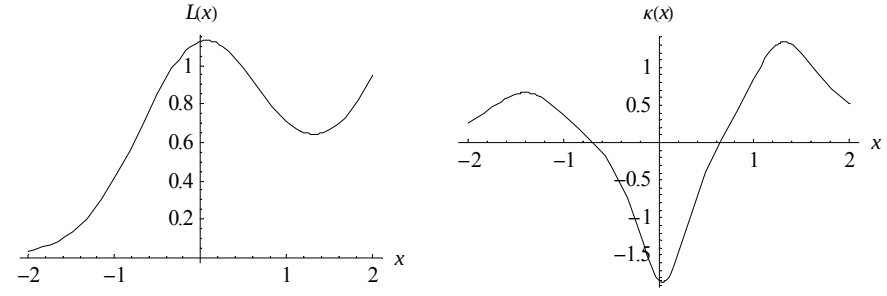


The following exponential function will provide useful information much like the three graphs above.

$$L(x) = \tfrac{1}{8}e^{x} + e^{-x^2} \tag{5.3}$$

This function has several useful properties. For one, it is an exponential function, which is a very important class of functions. Also, much like the polynomial $P(x)$ defined in (5.1), $L(x)$ has three local maxima in the magnitude of the curvature. However, the curvature of $L(x)$ is fundamentally different because it is not as concentrated near those extrema. These features will provide a check on some of the predictions made in the description of Fig. 5.9.

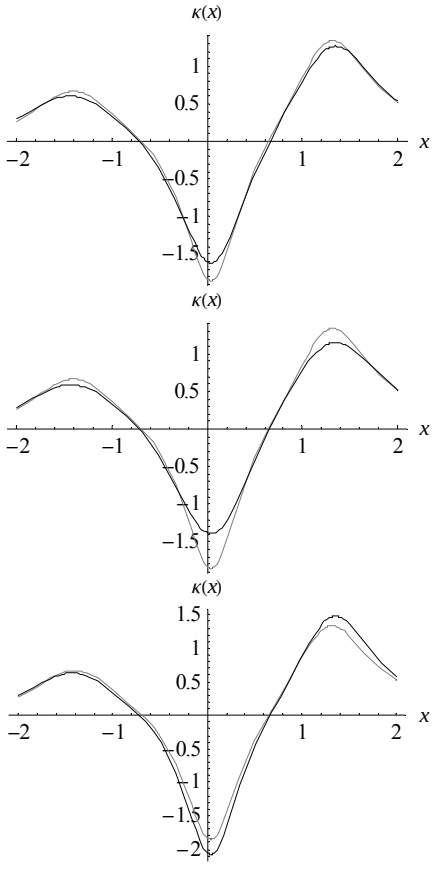
Figure 5.10



The following three graphs show the analysis of the function $L(x)$ as defined in (5.3). This time, a very large step size ($h = 0.5$) is used. Once again, the gray curve is the analytical curvature in all three cases. From top to bottom the graphs use second-order

finite difference, geometric and moving frame techniques, respectively. Notice that without points of high curvature, the three techniques seem to perform about equally. Also, notice that once again the moving frame technique overestimated while the geometric technique underestimated.

Figure 5.11



6. CONCLUSION–SECOND-ORDER TECHNIQUES

At this point, many of the previous predictions about the various numerical schemes have been verified, and the analysis has also contributed several new predictions. First, the verifications of the previously held predictions shall be summarized, and following that will be a summary of new details which have arisen from the various numerical experiments.

The most important qualities that can be concluded from the data presented in this paper is that all four of the techniques converge to the curvature for suitably small step size, and the geometric and moving frame methods are invariant to rotations. What was not mentioned was that all of these techniques (including the finite difference methods) are indifferent to rigid transformations. As a consequence, the geometric and moving frame techniques are invariant to all Euclidean transformations [3]. Thus (2.7) could be rewritten to include arbitrary offsets.

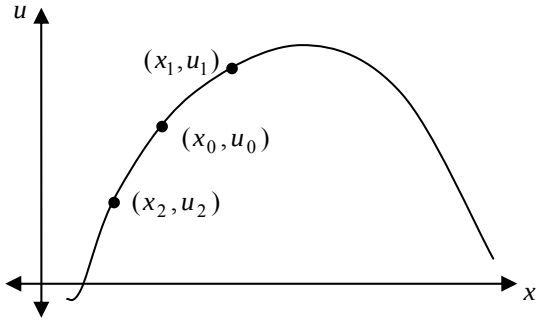
$$\begin{bmatrix} y_i \\ v_i \end{bmatrix} = \begin{bmatrix} a \\ b \end{bmatrix} + \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x_i \\ u_i \end{bmatrix} \tag{6.1}$$

Specifically, the data showed that the two invariant techniques had in general at least first-order convergence, but often behaved like second-order techniques (See Figs. 5.3, 5.5 and 5.9.). As predicted, the difference in behavior appears to rely on the equality of distance between points.

Another primary result is that the curvature numerical approximation based on the geometric scheme consistently outperformed the numerical approximation derived from moving frames. On several occasions (Figs. 5.3 and 5.5), the two gave nearly identical results, while in one occasion (Fig. 5.2), the moving frame technique actually performed substantially better for relatively small step size. However, in difficult situations, the geometric technique seems to be the most stable. As a key example, the errors of the moving frame technique were uniformly higher than those of the geometric technique at all points with high curvature.

Despite the similar behavior of the two invariant techniques, there is one fundamental difference that manifested itself in the numerical data. Namely, the invariant scheme tended to overestimate the magnitude of the curvature while all of the other schemes seemed to underestimate the magnitude of the curvature. One can relatively easily see why the geometric scheme predicts lower values by observing Fig. 3.1. Namely, the circle fit is larger than the true osculating circle unless there is a nonzero local minimum of the absolute value of the curvature enclosed among the three fitted points of the circle [4]. However, the following analysis of Figs. 4.1 and 4.2 shows that the moving frames technique will *almost always* predict a higher curvature magnitude than the analytical value.

Figure 6.1



Recalling the discussion in the Section 4, when calculating the curvature, one assumes the first derivative is precisely 0. Let the curve in the *HK*-plane be called *K*(*H*). Now K_H is zero to a first-order approximation. However, Fig. 4.2 illustrates a case where K_H is noticeably greater than zero. Since K_H is a real number, $K_H^2 \geq 0$. The analytical curvature in Fig. 4.1 at z_0 is identically that of I_0 in Fig. 4.2.

$$\kappa = \frac{u_{xx}}{(1+u_x^2)^{3/2}} = \frac{K_{HH}}{(1+K_H^2)^{3/2}} \tag{6.2}$$

The technique using (4.13) merely calculates K_{HH}. Call this approximation $\tilde{\kappa}$.

$$\tilde{\kappa} = [I_0 I_1 I_2] = K_{HH} + O(2) \tag{6.3}$$

$$\kappa = \frac{K_{HH}}{(1+K_H^2)^{3/2}} \le K_{HH} = \tilde{\kappa} + O(2) \tag{6.4}$$

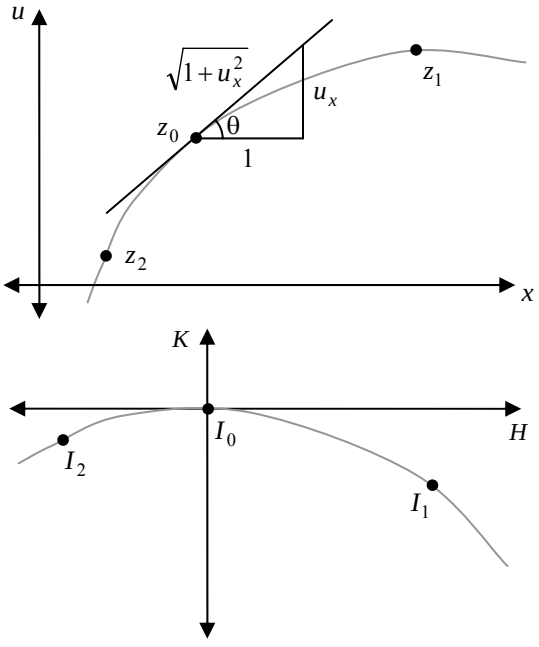
Therefore in order to have $|\tilde{\kappa}| < |\kappa|$, the second-order error of K_{HH} must overwhelm the first-order error of K_H.

7. HIGHER-ORDER INVARIANT TECHNIQUES

As discussed in the section on the moving frames technique, one can construct an invariant scheme by considering the points on a curve in a coordinate system in which the first derivative is zero or close to zero.

The first graph below shows a point chosen at random from a curve, and the second graph shows the curve in the first graph translated by $-z_0$ and rotated by –θ. The second graph is similar to Fig. 4.2, but in this case, the curve has been rotated by exactly –θ whereas the curve in Fig. 4.2 was rotated so that I_1 was on the *H*-axis.

Figure 7.1



Here are the useful formulas and conventions for the transformation illustrated above. Notice also that $\iota : \mathbb{R}^2 \to \mathbb{R}^2$ and $\iota \in SE_2$.

$$\iota : z_k \mapsto I_k \tag{4.2}$$

$$\iota(z) = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} (z - z_0) \tag{4.3}$$

$$\cos\theta = \frac{1}{\sqrt{1+u_x^2}} \tag{4.4}$$

$$\sin\theta = \frac{u_x}{\sqrt{1+u_x^2}} \tag{4.5}$$

The only approximation to be made in this transformation is that of u_x. In the original technique, the approximation $u_x = [z_0 z_1]$ was used. The first- second-order approximations for u_x in terms of divided differences are

$$u_x = [z_0 z_1] + O(1), \text{ and} \tag{7.1}$$

$$u_x = [z_0 z_1] + [z_0 z_2] - [z_1 z_2] + O(2). \tag{7.2}$$

In order to analyze the error associated with the curvature, set $v(H) = \iota(u(x))$.

$$\kappa = \frac{u_{xx}}{(1+u_x^2)^{3/2}} = \frac{K_{HH}}{(1+K_H^2)^{3/2}} \tag{6.2}$$

According to (7.1), $K_H = 0 + O(1)$. In addition, using a Taylor expansion shows that $(1+O(1)^2)^{3/2} = 1 + \frac{3}{2}O(2) = 1 + O(2)$.

$$\kappa = \frac{[I_0 I_1 I_2] + O(2)}{(1+O(1)^2)^{3/2}} = \frac{[I_0 I_1 I_2] + O(2)}{1+O(2)} \tag{7.3}$$

Now another Taylor expansion is needed.

$$\frac{a+x}{1+x} = a + (1-a)x + O(2) \tag{7.4}$$

$$\kappa = [I_0 I_1 I_2] + O(2) \tag{7.5}$$

Thus the approximation using (7.1) is indeed a second-order approximation to the curvature. When (7.2) is used, $K_H = O(2)$. Then $(1+K_H^2)^{3/2} = 1 + O(4)$.

$$\kappa = \frac{[I_0 I_1 I_2] + O(2)}{1+O(4)} = [I_0 I_1 I_2] + O(2) \tag{7.6}$$

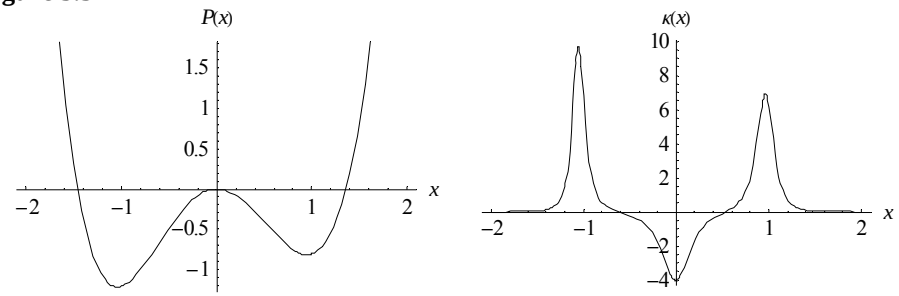
Somewhat surprisingly, the order of the error is not affected by choosing either (7.1) or (7.2). However, one might still like to see how the two techniques compare. One reason for this is that the geometric technique (when the points are essentially equally spaced), the second-order finite difference technique and the moving frame technique from before are all predicted to have second-order convergence. However, the three methods had some differences, as shown in Figs. 5.1-11.

8. COMPARISON OF SECOND-ORDER MOVING FRAMES TECHNIQUES

In addition to comparing the various second-order techniques, the following section employs the use of logarithmic plots of errors to test if the correct order has indeed been attained.

Consider once again the polynomial defined in (5.1).

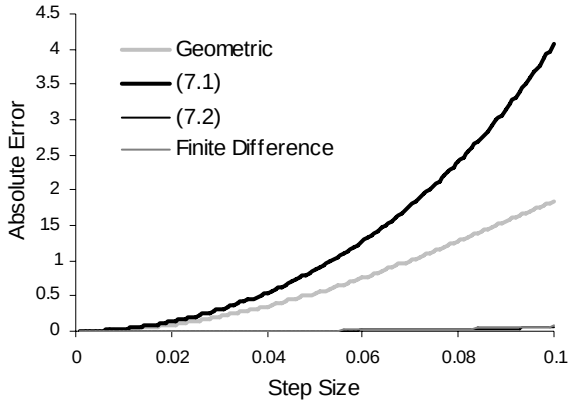
Figure 5.3

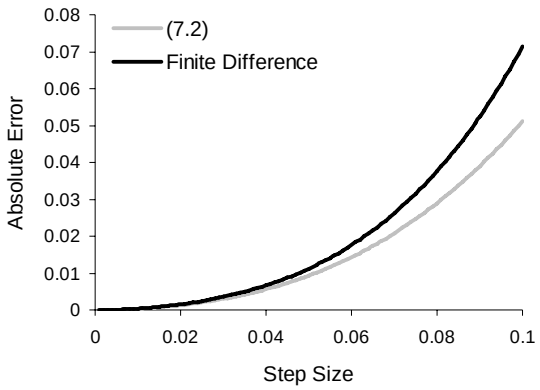


The point $\kappa(-1.05) = 9.625$ will be used for further analysis.

The first chart shows the absolute errors of the various second-order methods at the point $x = -1.05$. As expected, the techniques using (3.1) and (4.13) performed the worst, and the finite difference technique performed much better. Astonishingly, the invariant technique based on (7.2) performed even better than the second-order finite difference scheme, as shown in the second graph.

Figure 8.1





Although the four methods used in Fig. 8.1 are all predicted to have second-order convergence, two of them performed much better than the other two. What is somewhat surprising is that the formula using (7.2) showed even better accuracy than the finite-difference technique.

Before continuing to verify the performance of that particular invariant numerical scheme, the order of convergence of the three methods should be confirmed. Let $\tilde{\kappa}$ be an *n*th-order approximation to κ, that is $\kappa = \tilde{\kappa} + O(n)$. Then presumably the following formula holds for some constant *A* with step size *h*.

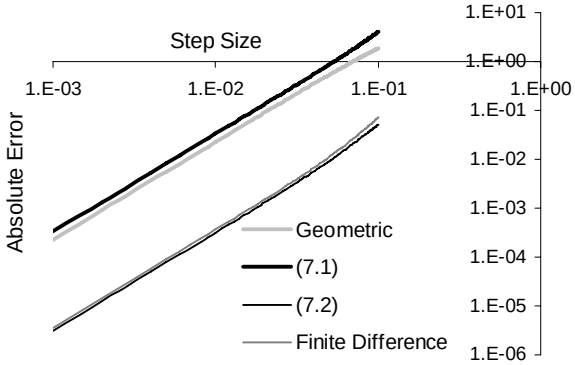
$$\kappa = \tilde{\kappa} + Ah^n + O(n+1) \tag{8.1}$$

Then, ignoring the $O(n+1)$ term, one realizes a linear relationship between the logarithms of step size and absolute error.

$$\log|E| = \log|\kappa - \tilde{\kappa}| = \log A + n \log h \tag{8.2}$$

Therefore, if the results from Fig. 8.1 are plotted on a logarithmic plot, the errors should form four lines whose slopes are all 2.

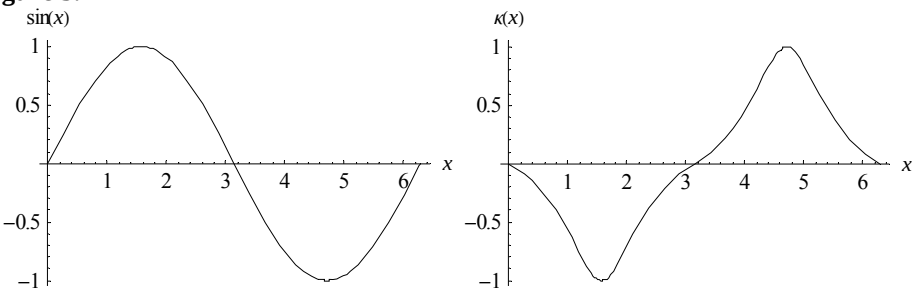
Figure 8.2



Using the result of (8.2), Fig. 8.2 confirms quite well that all four techniques were second-order numerical approximations to the curvature.

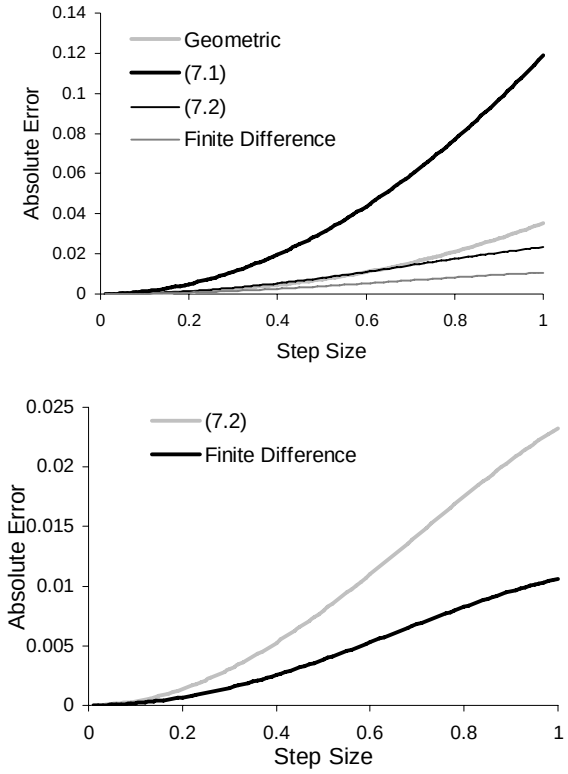
In order to test further the performance of the second moving frames technique, consider again the function $u(x) = \sin x$. At the point $x = 1$, the moving frames technique based on $u_x = [z_0 z_1]$ performed noticeably worse than the geometric scheme. Perhaps in this case the moving frames technique based on the second-order approximation to u_x will not perform as well as the second-order finite difference approximation to (2.1).

Figure 5.1



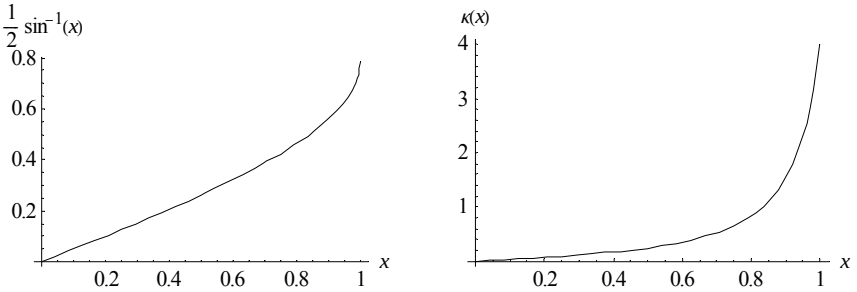
The errors are plotted for the second-order techniques for the function sin x at $x = 1$.

Figure 8.3



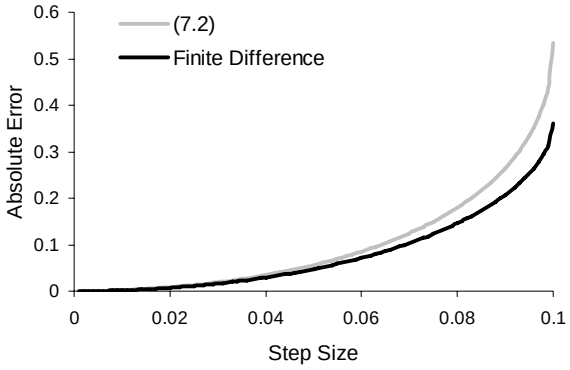
Once again the scheme using the second-order approximation to u_x performed much better than the scheme using the first-order approximation to u_x. However, this time the finite-difference approximation performed even better.

Figure 5.6



Once again the curve from Fig. 5.6 is analyzed at $x = 0.9$. Only the two best-performing second-order techniques are analyzed.

Figure 8.4



Apparently the finite difference technique performs in general slightly better than the invariant scheme. However, the accuracy of the invariant technique has been vastly improved.

9. GENERALIZATION OF ORDER

The analysis above showed that a better approximation of u_x caused a noticeably improved approximation for κ. If more points are allowed to be used, even better approximations for u_x can be found, and by mapping all of the points to the *HK*-plane, one can find a higher-order approximation of K_{HH}.

A general formula for the approximation to u_x is given by Newton's formula [5].

$$u(x) = u_0 + (x - x_0)[z_0 z_1] + \ldots + (x - x_0)\cdots(x - x_{n-1})[z_0 \cdots z_n] + \ldots$$

$$u(x) = u_0 + \sum_{i=1}^{n}\left([z_0 \cdots z_i]\prod_{j=0}^{i-1}(x - x_j)\right) + O(n) \qquad (9.1)$$

Subtracting u_0 from both sides and dividing by the common factor gives a formula for $[zz_0]$ if $z = (x, u(x))$. Then if $x \to x_0$, $[zz_0] \to u_x(x_0)$.

$$\frac{u(x)-u_0}{x-x_0} = \sum_{i=1}^{n}\left([z_0 \cdots z_i] \prod_{j=1}^{i-1} (x - x_j) \right) + O(n) \tag{9.2}$$

$$u_x(x_0) = \sum_{i=1}^{n}\left([z_0 \cdots z_i] \prod_{j=1}^{i-1} (x_0 - x_j) \right) + O(n) \tag{9.3}$$

Below are a few examples of the uses of (9.3).

$$u_x = [z_0 z_1] + O(1) \tag{7.1}$$

$$u_x = [z_0 z_1] + (x - x_1)[z_0 z_1 z_2] + O(2) \tag{9.4}$$

$$\begin{aligned} u_x = {} & [z_0 z_1] + (x_0 - x_1)[z_0 z_1 z_2] \\ & + (x_0 - x_1)(x_0 - x_2)[z_0 z_1 z_2 z_3] + O(3) \end{aligned} \tag{9.5}$$

One can also use Newton's formula to find a formula for higher derivatives. Of course, only the second derivative is of importance here. To do this, one directly differentiates Newton's formula [4, p. 66]. For convenience, say $\alpha_k = x - x_k$.

$$u = u_0 + \alpha_0[z_0 z_1] + \alpha_0\alpha_1[z_0 z_1 z_2] + \alpha_0\alpha_1\alpha_2[z_0 z_1 z_2 z_3] + \ldots \tag{9.6}$$

$$\begin{aligned} u_x = {} & [z_0 z_1] + (\alpha_0 + \alpha_1)[z_0 z_1 z_2] \\ & + (\alpha_0\alpha_1 + \alpha_0\alpha_2 + \alpha_1\alpha_2)[z_0 z_1 z_2 z_3] + \ldots \end{aligned} \tag{9.7}$$

$$\tfrac{1}{2}u_{xx} = [z_0 z_1 z_2] + (\alpha_0 + \alpha_1 + \alpha_2)[z_0 z_1 z_2 z_3] + \ldots \tag{9.8}$$

At first (9.7) appears to be in contradiction with (9.3)-(9.5). However when calculating the derivatives at $x = x_0$, $\alpha_0 = 0$. The second derivatives for this process are all calculated in the *HK*-plane, so the formula $\alpha_k = H_0 - H_k$ must be used. However, $H_0 = 0$. Below are some sample second derivatives.

$$K_{HH} = [I_0 I_1 I_2] + O(2) \tag{9.9}$$

$$K_{HH} = [I_0 I_1 I_2] - (H_1 + H_2)[I_0 I_1 I_2 I_3] + O(3) \tag{9.10}$$

$$\begin{aligned} K_{HH} = {} & [I_0 I_1 I_2] - (H_1 + H_2)[I_0 I_1 I_2 I_3] + \\ & - (H_1 H_2 + H_1 H_3 + H_2 H_3)[I_0 \cdots I_4] + O(4) \end{aligned}$$

Now set $K^{(2)} = K_{HH} + O(n)$ and $u^{(1)} = u_x + O(n/2)$. Then $K^{(1)} = O(\frac{n}{2})$, and the following gives the approximation for curvature.

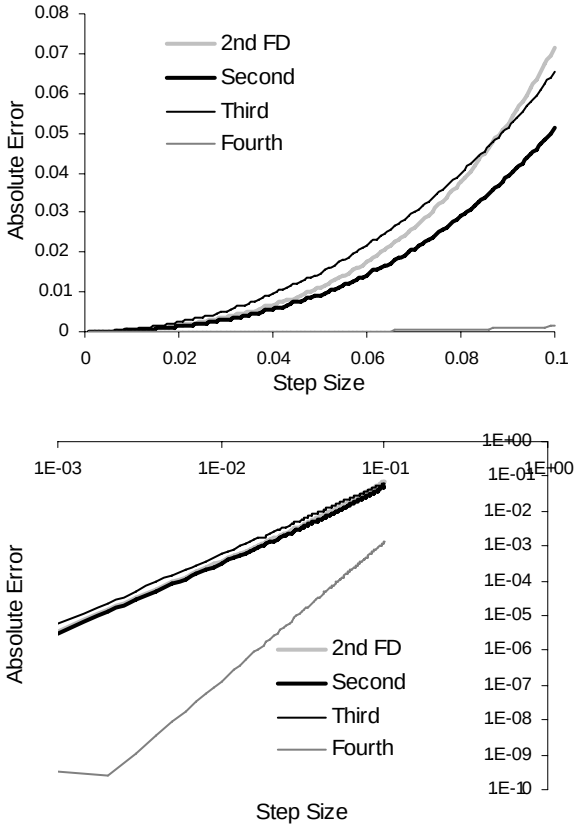
$$\begin{aligned} \kappa = \frac{K_{HH}}{(1 + K_H^2)^{3/2}} = \frac{K^{(2)} + O(n)}{(1 + O(n))^{3/2}} = \frac{K^{(2)} + O(n)}{1 + \frac{3}{2}O(n)} \\ = \left(K^{(2)} + O(n)\right)\left(1 - \tfrac{3}{2}O(n)\right) \\ \kappa = K^{(2)} + O(n) \end{aligned} \tag{9.11}$$

This shows that selecting an $n/2$-order approximation for u_x and an nth-order approximation to K_{HH} gives an nth-order approximation to the curvature. In fact, if the approximation for u_x is of order m and the approximation for K_{HH} is of order n, the order of the approximation for the curvature is min $\{2m, n\}$. However, the data from Figs. 8.1-3 seem to suggest that $n = m$ should give better results. In a practical sense, this also means that all $(n + 1)$ points are used in both approximations.

10. Comparison of Higher-order Moving Frames Techniques

First consider again $P(x)$ from (5.1) at $x = -1$. The three moving frame approximations compared to the second-order finite difference method all use $m = n$.

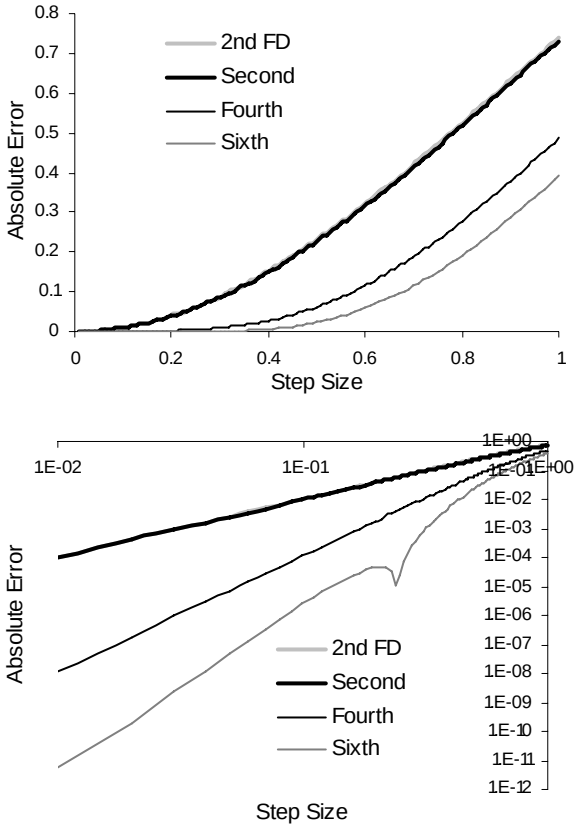
Figure 10.1



Everything performed as expected except for the third-order technique. However, the loss of some accuracy when using an even number of points is not entirely unanticipated. Instead further discussion should be limited to even-order approximations to the curvature. Indeed, the chart above demonstrates that the fourth-order technique performed as expected, and step sizes as large as 0.1 gave errors of only 0.001 in the curvature.

Again consider $L(x)$ as defined in (5.3) at $x = 0$. This time a sixth-order method is successfully executed, and the higher-order methods have errors less than 10^{-4} with a step size of 0.1. For applications, this error is much smaller than the what is required.

Figure 10.2



Notice that in the logarithmic plots of Figs. 8.1, 10.1 and 10.2, the error lines of the differentially invariant methods all intercept the vertical axis between 10^{-1} and 10. According to (8.2) that intercept is just the coefficient for the error term. If A represents this coefficient, then an upper bound in most cases for A appears to be 10. Therefore given a step size h and an approximation of order n, the maximum error should be $10h^n$.

11. DERIVATIVE OF CURVATURE

The simplest Euclidean invariant quantity of a curve apart from curvature is the derivative of the curvature with respect to arc length [3]. Each of the methods for curvature suggests at least one method of approximating the derivative of curvature. However, understanding the performance of these methods can be quite subtle, and some of the methods that at first appear obvious may not be the best methods.

Differentiating (2.1) with respect to arc length is done by using the chain rule.

$$\kappa_s = \frac{d\kappa}{ds} = \frac{(1+u_x^2)u_{xxx} - 3u_x u_{xx}^2}{(1+u_x^2)^3} \tag{11.1}$$

Since there is an explicit formula for κ_s, one can apply in a very straightforward manner finite difference approximations to each of the derivatives of u. This method has similar performance to the finite difference approximations to (2.1), *i.e.* it is very accurate unless there are points that are nearly vertical. Also, since u_{xxx} is in the formula for κ_s, the lowest approximation to (11.1) is third-order. However, third-order approximations have potentially poor behavior due to the asymmetry of points used, so discussion on this kind of technique will be limited to fourth-order approximations.

Figure 11.1

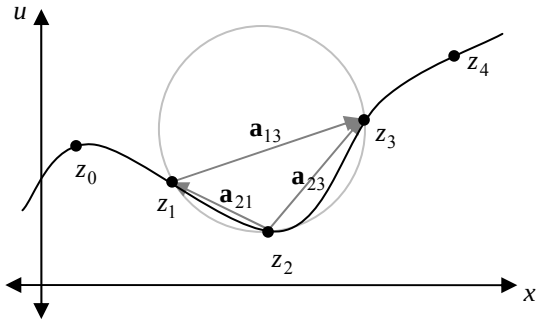


Fig. 11.1 is just an extension of Fig. 3.1 with a generalized notation. Using the geometric approximation to curvature, one can use (3.1) to find the curvature at z_1, z_2 and z_3. Knowing the curvature at these three points suggests many possibilities for applying finite differences to these curvatures, since the distance between points is an approximation of arc length. This choice of notation has the following properties.

$$z_k = (x_k, u_k) \tag{4.1}$$

$$\mathbf{a}_{ik} = z_k - z_i \tag{11.1}$$

$$a_{ki} = a_{ik} = |\mathbf{a}_{ik}| \tag{11.2}$$

Equation (11.3) is simply (3.1) rewritten in this new notation. This form has a very understandable way of handling the sign of κ. In the formulas, $\kappa(z_i)$ is the analytical curvature at z_i, and $\widetilde{\kappa}(z_i)$ is the approximation. The same idea is used for κ_s.

$$\widetilde{\kappa}(z_i) = \frac{2\det[\mathbf{a}_{i,i+1}\ \ \mathbf{a}_{i,i-1}]}{a_{i,i+1}a_{i,i-1}a_{i-1,i+1}} \tag{11.3}$$

There is another pattern to be noticed about the numerator of (11.3). If $\mathbf{A}_i$ is a matrix constructed in an intuitive way out of the distances between z_i and its immediately neighboring points, then a very compact formula for the curvature is available.

$$\mathbf{A}_i = \begin{bmatrix} 0 & a_{i-1,i} & a_{i-1,i+1} \\ a_{i,i-1} & 0 & a_{i,i+1} \\ a_{i+1,i-1} & a_{i+1,i} & 0 \end{bmatrix} \tag{11.4}$$

$$\widetilde{\kappa}(z_i) = 4\frac{\det[\mathbf{a}_{i,i+1}\ \ \mathbf{a}_{i,i-1}]}{\det \mathbf{A}_i} \tag{11.5}$$

This immediately suggests two approximations for κ_s [2].

$$\widetilde{\kappa}_s(z_i) = \frac{\widetilde{\kappa}(z_{i+1}) - \widetilde{\kappa}(z_i)}{a_{i,i+1}} \tag{11.6}$$

$$\widetilde{\kappa}_s(z_i) = \frac{\widetilde{\kappa}(z_{i+1}) - \widetilde{\kappa}(z_{i-1})}{a_{i-1,i+1}} \tag{11.7}$$

However, if the points are not equally spaced, these two approximations do not converge [1]. However, when the points are reasonably well-spaced, they provide a simple and useful approximation to κ_s. However, there are three less intuitive approximations which do converge [1].

$$\widetilde{\kappa}_s(z_i) = 3\frac{\widetilde{\kappa}(z_{i+1}) - \widetilde{\kappa}(z_i)}{a_{i,i-1} + a_{i,i+1} + a_{i+1,i+2}} \tag{11.8}$$

$$\begin{aligned}\widetilde{\kappa}_s(z_i) = &\frac{3}{2}\frac{\widetilde{\kappa}(z_{i+1}) - \widetilde{\kappa}(z_i)}{a_{i,i-1} + a_{i,i+1} + a_{i+1,i+2}} \\ &+ \frac{3}{2}\frac{\widetilde{\kappa}(z_i) - \widetilde{\kappa}(z_{i-1})}{a_{i,i-1} + a_{i,i+1} + a_{i-1,i-2}}\end{aligned} \tag{11.9}$$

$$\widetilde{\kappa}_s(z_i) = 3\frac{\widetilde{\kappa}(z_{i+1}) - \widetilde{\kappa}(z_{i-1})}{2a_{i,i-1} + 2a_{i,i+1} + a_{i+1,i+2} + a_{i-1,i-2}} \tag{11.10}$$

Finally, consider again the rotations that were utilized in Fig. 7.1. Recall that the mapping $\iota : z_k \mapsto I_k$ has the property that K_H is zero at the point of interest.

$$\begin{aligned}\kappa_s &= \frac{(1+u_x^2)u_{xxx} - 3u_x u_{xx}^2}{(1+u_x^2)^3} \\ &= \frac{(1+K_H^2)K_{HHH} - 3K_H K_{HH}^2}{(1+K_H^2)^3} \\ \kappa_s &= K_{HHH}\end{aligned} \tag{11.11}$$

Therefore all of the ideas from the moving frames that were applied to κ are also valid for κ_s. However, now general approximations to the third derivative are needed. In addition, since K_H appears in the numerator of the expression for κ_s, K_H must also be an approximation of order *n* if (11.11) is to be valid as an *n*th-order expression.

At this point, expressions for the third derivative are needed. Differentiating (9.8) with respect to *x* provides one method to do this, but differentiation can reduce the order of the approximation. As a result, more points may be needed when calculating the derivatives. For instance, the fourth-order scheme requires a maximum of seven points for the third derivative.

$$\tfrac{1}{6}u_{xxx} = [z_0 z_1 z_2 z_3] + \left(\sum_{i=0}^{3} \alpha_i\right)[z_0 z_1 z_2 z_3 z_4] + \ldots \tag{11.12}$$

Noting that $\alpha_k = H_0 - H_k$ and $H_0 = 0$ in the *HK*-plane, the following are sample expressions for the third derivative.

$$K_{HH} = 6[I_0 I_1 I_2 I_3] + O(3) \tag{11.13}$$

$$K_{HH} = 6[I_0 I_1 I_2 I_3] - 6(H_1 + H_2 + H_3)[I_0 \ldots I_4] + O(4) \tag{11.14}$$

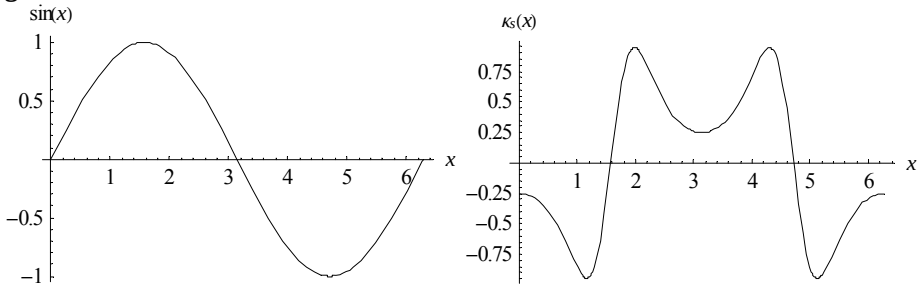
12. Performance of Derivative Approximations

Much like the experiments for the numerical approximations to curvature, the techniques for measuring κ_s should be compared at a very special set of points. Because the formula for the derivative of curvature is complicated, even simple functions like sin *x* have interesting κ_s graphs. The formula below is κ_s for $u = \sin x$.

$$\kappa_s = \frac{(-3 + \cos 2x)\cos x}{(1 + \cos^2 x)^3} \tag{11.15}$$

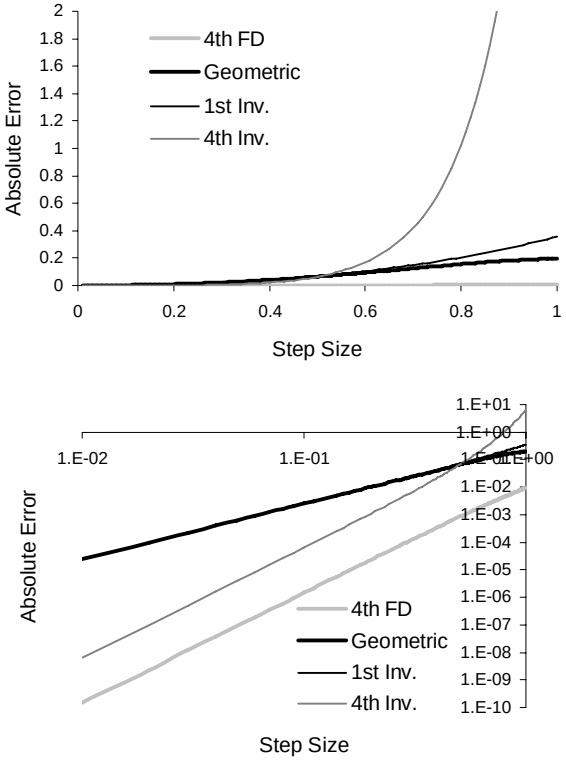
Another way to look at this is that the approximations of κ_s must analyze very difficult functions. The first point analyzed will be at $x = 0$ for $u = \sin x$.

Figure 12.1



In Fig. 12.2, the top chart again shows the absolute errors while the second shows the logarithmic plot of the top figure. The higher-order invariant technique had very large errors at large step sizes, but the approximation had the correct order of convergence. According to the logarithmic plot, the invariant technique outperforms the second-order techniques when the step size is smaller than about 0.5.

Figure 12.2



The results shown in Fig. 12.2 are much like those of Fig. 8.2; now the correct order has been attained, but the invariant technique is outperformed by the straightforward finite difference technique. In addition, both of the fourth-order techniques required seven points instead of five, but this was due to the finite difference approximation of the third derivative that was used. However, this time there appears to be no immediate solution that will make the invariant approximation more accurate without affecting its order.

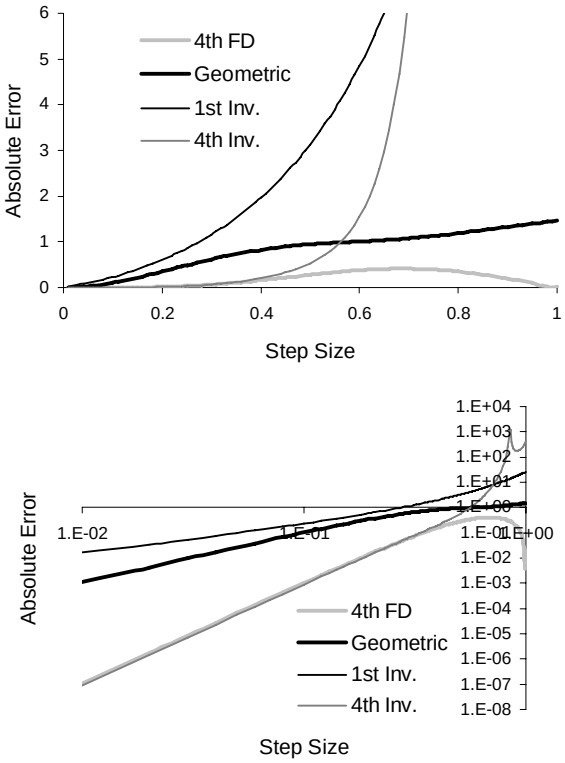
The next potential problem lies in curves with high second-derivatives. At $x = 0$, the curve $u = \sin x$ has a second derivative of zero. One can see how rotating the curve about the origin will not alter this fact. Now recalling the derivation of (11.11), one sees that if K_{HH} is large, then the approximation of κ_s may not have the correct order of convergence.

$$\kappa_s = \frac{(1+K_H^2)K_{HHH} - 3K_H K_{HH}^2}{(1+K_H^2)^3}$$

Once again the exponential function defined in (5.3) will be used for its moderate but nontrivial behavior.

At the point $x = 1.1$, $L''(x) \approx 1.22$. At this point, the accuracy is not particularly good, but the correct orders of convergence can be verified.

Figure 12.3



There are two properties in this example that deviate from the expected. First, the technique which has been labeled as "first-order invariant" and uses (4.13) and (11.11) showed only first-order convergence. This should not really be a surprise, except that the technique showed second-order behavior in Fig. 12.2. Furthermore, not only did the fourth-order invariant method indeed show fourth-order convergence, but it matched the performance of the finite difference method for step sizes smaller than about 0.4 (This can be seen from the logarithmic plot.). This is unusual because the nonzero second derivative in $L(x)$ was predicted to cause problems for the invariant techniques, not the finite difference methods.

13. EFFECTS OF SPACING

In all of the examples up to this point, the points of the graph were selected by equally dividing the x-axis and picking including the point of the curve directly above. That is to say that the points used in the formulas formed a set of the variety $Z = \{(x, u(x)), (x + h, u(x + h)),\ldots\}$. This is usually the most obvious way to select points if a curve is parameterized by $(x, u(x))$. However, a consequence is that adjacent points of Z are not equidistant.

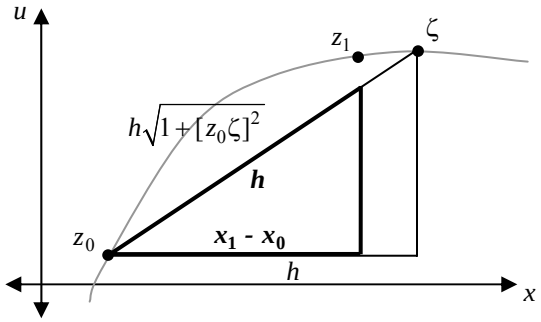
An algorithm to select points that are equally spaced to a high degree of accuracy is not terribly difficult. Suppose $z_0 = (x_0, u(x_0))$ is the current last point in Z, and one wishes to add a point to Z that is a distance h from z_0. To do so, use an initial guess $\zeta = (x_0 + h, u(x_0 + h))$ to estimate the first derivative, $[z_0\zeta]$. Now the actual distance

from z_0 to ζ is $h\sqrt{1+[z_0\zeta]^2}$. Then by scaling this distance down to just *h*, the horizontal distance is reduced. Therefore the following choice of z_1 is a distance $h + O(1)$ from z_0 .

$$x_1 = x_0 + \frac{h}{\sqrt{1+[z_0\zeta]^2}} \tag{13.1}$$

$$z_1 = (x_1, u(x_1)) \tag{13.2}$$

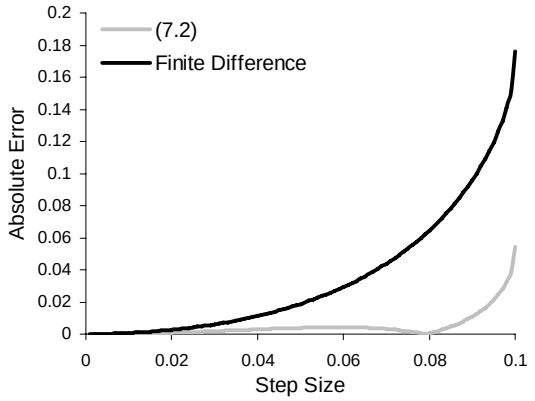
Figure 13.1



One effect of this point-selection scheme is that given the same value of *h*, the points in *Z* are slightly closer than they would have been in the old point-selection scheme. As a result all of the numerical approximations will appear to perform slightly better, but this is really just an artifact of the distances between points. The interesting issue concerns which methods will be affected most by the change in point-selection methods.

In Fig. 8.4 the invariant technique did not perform quite as well as the finite difference approximation to curvature. When the points are equally spaced, however, the invariant scheme performs much better, as shown in Fig. 13.2 below.

Figure 13.2



The function analyzed in Fig. 13.2 is $u = \frac{1}{2}\sin^{-1} x$, and the point analyzed is $x = 0.9$. At this point, the slope of the curve is very high. A large slope should not affect the invariant scheme, however, since the approximation is invariant to rotations. However,

since the second derivative of u is greater than 5 at $x = 0.9$, the distances between points in the old point-selection scheme are quite varied. To be sure that the moving frames technique actually works much better when the points are equally spaced, the data from Figs. 8.1 and 8.3 are recalculated using the equidistant point-selection technique.

In Fig. 8.1, the polynomial $P(x)$ is analyzed at $x = -1.05$. Fig. 13.3 shows the performance of the fourth-order techniques using equal spacing at the same point. In Fig. 8.3, the curve analyzed is $u = \sin x$ at $x = 1$. Fig. 13.4 repeats Fig. 8.3 using equally spaced points.

Figure 13.3

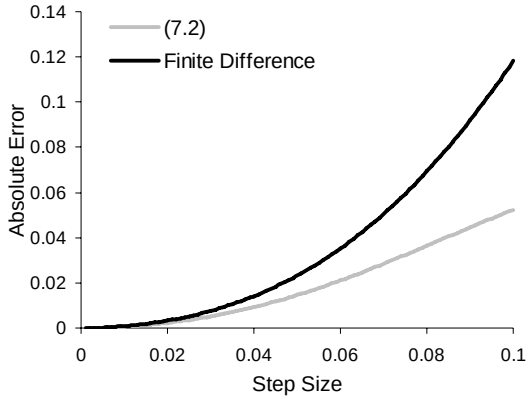
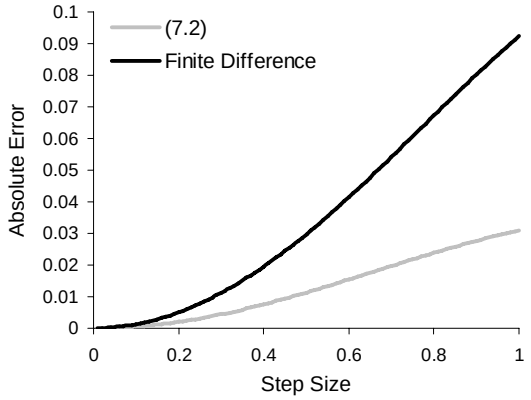


Figure 13.4



14. CONCLUSION

The primary result of this analysis was the development of a well-tested set of higher-order numerical approximations to Euclidean curvature that are invariant to rigid transformations. While the data showed that the geometric technique is perhaps the best second-order technique, the development of other invariant techniques, especially the fourth-order version, was a significant contribution.

With this analysis, numerical approximations to Euclidean curvature are understood fairly well. The experience with this simple form of curvature may lead to a better

understanding also of different types of curvature and differential invariants to more complicated groups of transformations.

BIBLIOGRAPHY

[1] Boutin, Mireille. "Numerically Invariant Signature Curves." International Journal of Computer Vision 40.3 (2000): 235-248

[2] Calabi, Eugenio, Peter J. Olver, Chehrzad Shakiban and Steven Haker. "Differential and Numerically Invariant Signature Curves Applied to Object Recognition." International Journal of Computer Vision 26.2 (1998): 107-135

[3] Olver, Peter J. "A Survey of Moving Frames." University of Minnesota, 2004.

[4] Osserman, Robert. "The Four-or-more Vertex Theorem." American Mathematical Monthly 92.5 (1985): 332-337

[5] Whittaker, Sir Edmund and G. Robinson. The Calculus of Observations. Blackie & Son Limited (1924).