3rd International Conference
″Research & Innovation in Engineering ″
COMAT 2014
16-17 October 2014, Braşov, Romania

USING CURVE FITTING AS A METHOD TO ANALYZE MOTION ANALISYS DATA FOR SPORTS

M. Mihălcică[1], V. Guiman[2], V. Munteanu[3]
University „Transilvania" of Brasov, ROMANIA, Department of Mechanics, mircea.mihalcica@unitbv.ro
University „Transilvania" of Brasov, ROMANIA, Department of Mechanics
University „Transilvania" of Brasov, ROMANIA, Department of Mechanics

Abstract: *In this paper we will describe the way experimental data captured by a cheap and portable motion analysis system dedicated to amateur and juniors sports (developed mainly for athletics) can be analyzed using curve fitting. We will present things related to the ways the mathematical model can be designed, using MATLAB code, and how the data can be interpreted and presented in order to offer athletes and their trainers ways to improve the athletes performances.*
Keywords*: curve fitting, motion analysis, sports, biomechanics*

1. INTRODUCTION

We will start considering that we already made the first steps to obtain raw data for athletic disciplines using a cheap and portable motion analysis system and now we own a few series of recordings, presented in the form of Excel tables and containing X and Y coordinates of the markers of the main joints of athletes during motion. What we aim is to transform this data so that it can be mathematically processed, and, in the end, to be able to extract some information and to obtain a set of results from the mathematical models applied. We found that curve fitting in MATLAB would be a great method to achieve these tasks.

2. MEANS AND METHODS

Once we have the data, we can just import the tables to MATLAB and use the command *cftool* in order to obtain a form of basic curve fitting for the data corresponding to a marker in motion during an athlete's try. Ideally, we would aim to obtain a single polynomial to fit the data points, and we will try to find the best option for such a polynomial - the factors which are being taken into consideration are the order of the polynomial and its ability to resemble the shape of the motion (usually, there is a trade-off here, meaning that, the higher the order of the polynomial, the better the results). The results of such a process are shown in Figure 1.

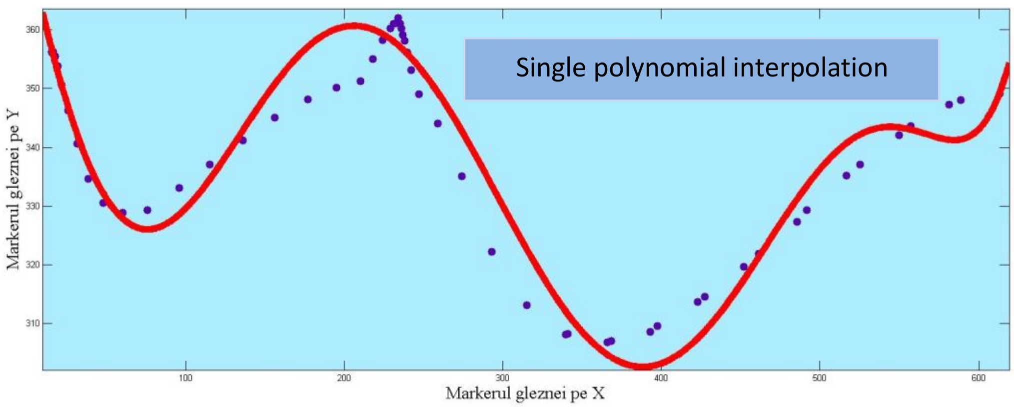

Fig 1. Curve fitting interpolation of the ankle marker of an athlete during running using a 7th order polynomial (the blue points are the experimental data being interpolated, the red curve being the curve fitting polynomial)

The results obtained will practically be the coefficients of the interpolant - we have a 7th degree polynomial, so it will be something like this:

f(x) = p1*x^7 + p2*x^6 + p3*x^5 + p4*x^4 + p5*x^3 + p6*x^2 + p7*x + p8, (1)

where MATLAB will provide us the values for p1 to p8.

As we can easily see, the results are good, but far from perfect. While in many cases these results can be considered good enough (the differences of shape and distance between the polynomial and the points is considered acceptable), there are a lot of situations where these kind of results are not considered good. For those situations, one of the best solutions to solve the problem is using a solution which interpolates the data segment by segment, like cubic spline interpolation, hermite interpolation and others. For our models we choose cubic spline interpolation and we determined the cubic spline functions in MATLAB, the results being shown in Figure 2.

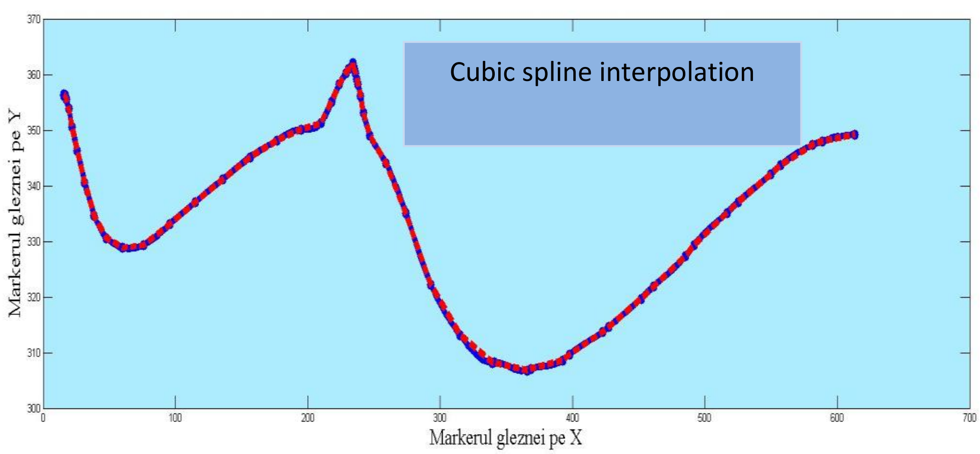

Fig 2. Curve fitting interpolation of the ankle marker of an athlete during running using cubic spline segments (3rd order polynomials)

We can easily see that the results presented by the cubic spline interpolant are much better than the single polynomial's. The shape of the interpolant almost perfectly matches the shape of the motion, and all points are included. This kind of interpolant unfortunately contains a polynomial (of a very low order, in our situation 3rd order polynomials were used) for each segment of the motion, for each two consecutive points of the motion. The results, in mathematical form, will also be coefficients, but now each segment will have its own set of coefficients. This kind of interpolant is much harder to work with from a mathematical (and programming) point of view.

3. CONCLUSIONS AND FUTURE WORK

Curve fitting is probably one of the best methods to be used when it comes to analyze the raw data captured with a motion analysis system. This can be achieved in manly two ways: using a single high order polynomial or using segmented curve fitting as cubic spline functions (low degree). The decision is for the analyzer to make: shall he use a single polynomial and lose some accuracy or shall he use the segments variant and much extra work for results.

ACKNOWLEDGEMENT: This paper is supported by the Sectoral Operational Programme Human Resources Development (SOP HRD), financed from the European Social Fund and by the Romanian Government under the project number POSDRU/159/1.5/S/134378.

REFERENCES

1. Dufy, DG. Advanced Engineering Mathematics with Matlab, 3rd ed., CRC Press, Taylor & Francis Group (2010)
2. Floater MS, Surazhsky T. Parameterization for curve interpolation, in Topics in multivariate approximation and interpolation, K. Jetter et al. (eds.), pp. 39-54, Elsevier (2006)
3. Sarfraz M, Raheem MA. Curve Designing Using a Rational Cubic Spline with Point and Interval Shape Control, IEEE Conference on Digital Object Identifier, Information Visualization (2000)