

Smoothing Splines on the Torus

F. Egebrand*, M. Egerstedt** and C. Martin***

Abstract—An algorithm is presented for constructing smoothing splines on the torus. The algorithm uses a particular representation of the torus and is suboptimal. However it produces good trajectories and among all possible smoothing spline it does very well in terms of the cost function.

I. INTRODUCTION

A manifold is a mathematical space that locally resembles a standard Euclidean space. The dimension of the manifold is defined as the dimension of the specific Euclidean space for which the manifold has similar properties. In general topological spaces are said to be compact if it exists a finite family of open sets that covers the space. Hence the circle is an example of a compact manifold, whereas the line is not. Even though a manifold locally resembles the Euclidean space, the global structure is often very complicated. Regarding the two-dimensional manifolds there are only a few spaces that are easy to visualize. Three of these manifolds are the plane, the sphere and the torus. However the plane is not compact, thus the torus and the sphere are the two simplest compact two dimensional manifolds.

In Egerstedt and Martins book [1] a lot of research in the area of constructing control theoretic splines is presented. Among other things Egerstedt and Martin constructed smoothing splines on the sphere. Primarily [1] is the basic reference to all the material in the first part of this paper. The Matlab code used to generate the smoothing spline is based on Magnus Egerstedt's code[2]. The final purpose of Section 3 is to construct a suboptimal smoothing spline on the two-dimensional torus. The torus is a useful tool in many applications. The higher dimensional tori arises in several areas including biopharmokinetics and the two dimensional torus is used in modeling two degree of

rotational freedom in manipulations and could also be used when modeling the movement of the eye as seen in Section 4. When constructing this spline on the torus one has to take into consideration the periodicity of the points on the torus. There are an infinite number of points in R^2 mapping on the same point on the torus. Thus the selection of data points plays an important role when it comes to the optimality of the spline.

A lot of research is done regarding splines. There is some literature concerning the interpolating spline on the torus, for example the technical report [3] written by S. Gomes, A. Kushpel, J. Levesley and D. Ragozin and material from J. Saranen and J. Anttila[4]. In many areas there is a lot of literature regarding splines on manifolds. M. Hofer and H. Pottmann have in [5] for example focused on the energy minimizing splines on manifolds. If the data is exact the interpolating spline is preferred. However most real data has inaccuracies and smoothing splines were designed to deal with data with noise. When it comes to the smoothing spline the research is more limited, but in statistics the smoothing spline is a well known tool. Yishao Zhou has together with M. Egerstedt, C. Martin and W. Dayawansa done a lot of interesting work in that area[6], [7], [8]. In the early 1970s Grace Wahba stated the smoothing spline problem as an optimization problem in a Hilbert space[9]. This was an important contribution to the work of introducing the control theoretic aspects of the smoothing splines. Concerning the suboptimal smoothing spline on the torus basically no literature could be found. One difficulty in constructing smoothing splines on an arbitrary manifold is often the absence of a usable concept of distance.

A. Paper outline

This paper is divided into three main sections. This first section is an introduction that presents the topic of the paper and the references used.

* KTH, Stockholm, Sweden, USA: ** Georgia Institute of Technology, Atlanta, Georgia. ***Texas Tech University, Lubbock, Texas (email: clyde.f.martin@ttu.edu)

The second section is called Introduction to the eye movement problem. This section explains the connection between problem of controlling the gaze from one direction to another in an energy minimizing way and the construction of a smoothing spline on the torus. In Section 2 the complexity of the eye is also discussed. The third section is named Smoothing splines on the torus and the fourth is called The movement of the eye.

The aim of Section 3 is to construct and plot a smoothing spline on a torus. In the subsections 3.1, 3.2 and 3.3 some basic concepts and definitions that will be necessary in the continuation of the paper are presented. The three concepts summarized are; the torus in section 3.1, the spline in section 3.2 and the control system in section 3.3. In subsection 3.4 an example of the construction of a smoothing spline in R will be discussed. Subsection 3.5 will treat how the selection of the points $\alpha_i \in R^2$ will effect the smoothing spline on the torus. These are the points that the spline is requested to pass optimally close when the points are mapped on the torus. An algorithm for trying to find an optimal set of these points with respect to minimizing the energy of the control system is presented. In Subsection 3.6 the construction of the smoothing spline on the torus take place. Two splines in R is redone to one spline on the two dimensional torus. Finally, in Section 3.7 some statistics is presented regarding the importance of the selection of points and the optimality of the smoothing spline on the torus is discussed. This is done by comparing the spline constructed in Subsection 3.6 to splines constructed from randomly selected points.

II. SMOOTHING SPLINES ON THE TORUS

A. The torus

The torus is a compact two dimensional manifold and resembles the Euclidean space R^2 locally. An example of a torus is shown in Figure 2. There are several ways to describe the torus. In this section two different representations in the Euclidean space R^3 , one in R^4 and one as a quotient group are discussed. Out of these four the last representation, as a quotient group, is of most interest since it describes how a torus arises from periodic data. The four descriptions is listed below.

- First the torus is described via a polynomial equation. This gives an implicit representation of the torus in the three dimensional space. In order to obtain the standard torus with major radius R and minor radius r , $0 < r < R$, a circle centered at $(R, 0, 0)$ is rotated about the z -axis. In other words the points $(x, y, z) \in R^3$ that satisfies the polynomial equation in Equation 2 lies on the standard torus.

$$(x^2 + y^2 + z^2 + R^2 - r^2)^2 - 4R^2(x^2 + y^2) = 0 \tag{1}$$

This equation results in a torus as in Figure 2 with the standard axis. When using the polynomial equation in order to describe the torus, each point on the torus is defined by three coordinates $(x, y, z) \in R^3$. However this is unnecessary many. Two coordinates will in the next description show to be enough.

- The second description is a parameterizations of the torus in R^3 . The torus can be parameterized as:

$$\begin{aligned} x(u, v) &= (R + r \cos v) \cos u \\ y(u, v) &= (R + r \cos v) \sin u \\ z(u, v) &= r \sin v \end{aligned} \tag{2}$$

where R is set to be the bigger radius, r is set to be the smaller radius and u and v are angles. It is worth noticing that both u and v have a period of 2π . Unlike the implicit representation in Equation 1 where three coordinates are needed to describe a specific point on the torus, only two angles u and v are needed in the parametric representation. This is useful in many real situations when a two-degree of rotational freedom arises. In this cases the torus is a well suited tool to visualize the change in the two angles. Moreover the parametric representation is equivalent to the implicit representation in Equation 2 and it is possible to switch between the representations.

- Another common way to describe the torus is as the Cartesian product of the two circles $S^1 \times S^1$. This is sometimes called the Mathematicians torus. The circle S^1 has the radius set to one and can be described in the two dimensional plane as the set of all points (x, y)

that satisfies

$$x^2 + y^2 - 1 = 0 \quad (3)$$

Accordingly the torus $S^1 \times S^1$ is embedded in a four dimensional space as the set of all points (w, x, y, z) with

$$\begin{aligned} x^2 + y^2 - 1 &= 0 \\ w^2 + z^2 - 1 &= 0 \end{aligned}$$

Hence four coordinates are used to describe a point on the torus when it is described this way. However a point on one of the circles S^1 can also be described in only one angle coordinate.

Each point on the torus $S^1 \times S^1$ is determined as the position on the two circles and the position on each circle can be described in only one angle coordinate. Hence the point on the torus is again described in a u and a v each with a period of 2π .

- Finally the torus can be described as a quotient group. This description is the one used in Section 3.6 when constructing the splines on the torus. The quotient group considered is the quotient of the Abelian group R^2/Z^2 , where R^2 is the group of two coordinate real numbers under addition and Z^2 is a subgroup of two coordinate integers. The cosets of Z^2 in R^2 are all sets of the form $(a + Z, b + Z)$ where $a, b \in [0, 1[$ are real numbers and Z is any integer. In a quotient group all points in a coset is considered equivalent. Hence the point $(4.5, -0.5)$ is considered equivalent to $(0.5, 0.5)$ and each coordinate has a period of one. The quotient group R^2/Z^2 is isomorphic to the circle group $S^1 \times S^1$. This is a result from the observation that in the circle group description a point on the torus could be described in only the two angles u, v where $u = \phi + 2\pi n$ and $v = \theta + 2\pi m$ $m, n \in R$ and $\phi, \theta \in [0, 2\pi[$. In other words; a point on the torus is described only in the positions on the two circles and the positions on each circle can be described in just one angle coordinate with the period 2π . In the quotient group all the points differing with any integer is considered equivalent. This could be visualized on a torus, where each point on the torus represent

a certain coset. In order to transforming the cosets to the torus $a, b \in [0, 1[$ is recalculated to angles.

$$\begin{aligned} \phi &= a \times 2\pi \\ \theta &= b \times 2\pi \end{aligned} \quad (4)$$

Thus $\phi, \theta \in [0, 2\pi[$ and can be consider as two angles on the two circles S^1 and therefore a point on the torus. This is visualized in Figure 6, where all points following the pattern will map on the same point of the torus. For example the point $(0.5, 0.5)$ will map at the same point on the torus as $(4.5, -0.5)$, since they both correspond to the same a and b and this leads in Equation 5 to the same angles ϕ and θ on the torus. Hence these two points are considered equivalent on the manifold and are represented on the same point on the torus.

B. The spline

A spline can be defined as piecewise polynomial curves that are stitched together as smoothly as possible, at the given nodal points. Splines are common in science and engineering. The standard spline is an interpolating curve, i.e it is required to pass through the data points, α_i . An example of an interpolating cubic spline is shown in Figure 4. This kind of spline is not well-suited to use when the data is inexact, because the classical spline tend to accent the errors in the data. To handle situations with inexact data the smoothing spline is a suitable tool. The smoothing spline is only required to pass close to the data points, and is therefore a better choice when approximating a curve based on inexact data. In fact, sometimes exact interpolation is not desirable. In many realistic situations passing close to the data points will be enough. One way to construct these smoothing splines is through control theory and it will be used here.

C. The control system

A system can be described in both an external and in an internal manner. A basic concept in the internal description is the state of the system. The state vector $x(t)$ should collect all important information about the system which is relevant for the output $y(t)$. The output vector will contain all aspects of the system that is of interest to measure.

When it comes to the smoothing spline, the output $y(t)$ will be a scalar. To be able to control the system, the control signal $u(t)$ is used. This is how the dynamics of the system could be changed. In the construction of the smoothing spline the control signal is also a scalar.

A linear control system is a control system which, with a fixed initial condition of the state vector $x(t_0)$, linearly maps input signals to output signals. The linear, time-invariant control system used to generate the spline is represented in state-space representation as:

$$\begin{aligned} \dot{x}(t) &= \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} x(t) + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u(t) \\ y(t) &= (1 \ 0) x(t) \end{aligned} \tag{5}$$

The choice of this system has many advantages that will make the calculations easier and produces the classical cubic spline.

D. Construction of a smoothing spline

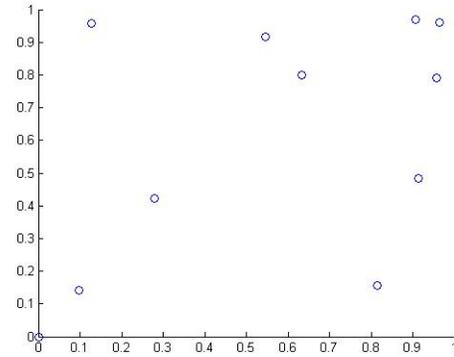
In this section the focus is on constructing a smoothing spline from points in R . Later, in subsection 3.6, it will show that this way of constructing the spline could be used even when the points are in R^2 on the torus. A smoothing spline in R is constructed by solving a control problem subject to the recommended single input single output control system in Equation 6. The smoothing spline is a result of an optimal control problem, where a cost function contains both the input signal and the distances between the data points, $\alpha_i^1 \in R$, and the output from the system at the given times, $y(t_i)$. The control problem is stated as:

$$\min_u \sum_{i=1}^N \omega_i (y(t_i) - \alpha_i^1)^2 + \rho \int_0^T u(t)^2 dt \tag{6}$$

$\omega_i, \rho > 0$

The resulting output curve $y(t) \in R$ is a spline that passes the N data points α_i^1 optimally close, depending on the choice of the parameters ω_i and the smoothing parameter ρ . Two examples of smoothing splines are shown in Figure 8.

Fig. 1. As a start all data points are situated in this unit square.



E. The data points

When constructing the spline on the torus it is convenient to think of the torus in terms of the quotient group R^2/Z^2 . This is the fourth description in subsection 3.1. The data points on the torus that the spline is supposed to pass optimally close is stated as a given set of two coordinate points $[\alpha, \beta, \gamma, \dots]$ where $\alpha = (\alpha^1, \alpha^2) \in R^2$ and so on. In this paper a set of ten points is considered and the points is at first stated in the interval $[0, 1[$. A possible scenario how this set of points is obtained is that the points are picked from the image of a torus, like the one in Figure 2 and that it is decided that the coordinates have a period of one. However, as seen in Figure 6 there are an infinite number of points in R^2 mapping on the same point on the torus and the given set is not definite. Any integer could be added to any of the coordinates of a data point and the point will still correspond to the same point on the torus. The data points in the first set are all situated in the square in R^2 showed in Figure 5.

This square is sometimes called the unit square. Because of the discussed periodicity it is impossible to tell in which unit square the data points that should be used when constructing the spline are situated and the points could very well be in different unit squares. Any combination of these points will be a correct choice. Hence a spline on the torus will not be unique. However, the different combinations of points in R^2 will lead to different input of the control system used to construct the spline. Hence the choice of data points could be optimized depending on situation and in that way

construct a unique spline. When constructing the spline in this paper the optimization is done with respect to the energy of the control system which should be minimized.

1) *Selection of the optimal data set:* In order to get the spline on the torus to be optimal in some sense, the choice of the data points α_i is, as prior mentioned, an essential issue. Since the optimization in this case is done with respect to the input energy of the control system, the selection of data points follows the same concept. The energy it takes to go from one point on the torus to the next depends on in which unit square the points are situated and the aim is to minimize this energy. In the starting data set $[\alpha, \beta, \gamma \dots]$ only the first point $\alpha \equiv \alpha_1$ is considered fixed. This assumption is the same as picking which unit square in Figure 6 to start in. Since all is symmetric it does not impose any restrictions on the spline to state $\alpha \equiv \alpha_1$ and start the optimization from the standard unit square. The other points are free to be chosen in an infinite number of ways. The final set of data points is then stated as $[\alpha_1, \alpha_2, \alpha_3 \dots]$.

The selection is performed in an intuitive way, where only nine close points are taken into consideration as the next optimal point. In Figure 7 α_1 is marked out as well as all the nine points that corresponds to a possible α_2 . The nine points are called β_i , $i = 1, 2, \dots, 9$. In order to decide which of these point to choose, a comparison of a cost from α_i to the nine β_i is made. The cost function used is the standard cost function:

$$J(u) = \int_0^{t_\beta} u^2(t) dt \quad (7)$$

This cost function is considered to be the minimum input energy used of the control system for moving from point α_1 to point β_i in the time t_β , in other words how to control the system from one fix point to another fixed point using as little energy as possible. In the Splinebook [1] this is called the minimum energy transfer problem. It is worth noticing that this way of choosing points does not take into consideration that the spline is smoothing; when the cost is calculated the points are interpolated. However it should give

an approximative solution to the optimal spline problem. With help from the Splinebook [1] u is rewritten in the parameters α_1 , β_i , t_β and x_0 . Of this only x_0 is considered unknown. Hence an exponential mean value approximation is done with respect to x_0 .

$$J(u) = ([2(\beta_i^1)^2 + 2(\alpha_1^1)^2 - 4\beta_i^1\alpha_1^1 + 4(t_\beta)^2]^2 + [2(\beta_i^2)^2 + 2(\alpha_1^2)^2 - 4\beta_i^2\alpha_1^2 + 4(t_\beta)^2]^2)^{1/2}$$

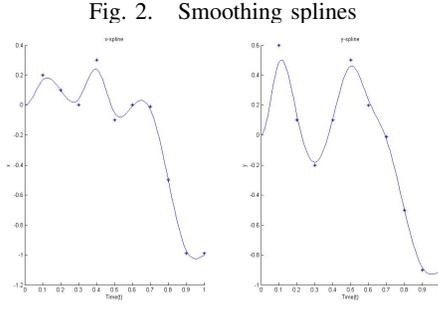
The superior index on α and β corresponds to which of the two coordinates it is referring to. The whole expression for the cost function is known and the algorithm could be performed by inserting the different values of β_i into the cost function.

In the first step the cost is calculated from the fixed optimal point $\alpha_1 = (\alpha_1^1, \alpha_1^2)$ to the nine points $\beta_i = (\beta_i^1 + [-1, 0, 1], \beta_i^2 + [-1, 0, 1])$ shown in Figure 7. t_β is the time when the spline is required to be optimally close to the data point β . When the cost has been calculated for all of the nine data points, the algorithm starts over again with the β_i corresponding to the smallest cost as the new fixed optimal point, $\alpha_2 = (\alpha_2^1, \alpha_2^2)$. An example of this is shown in Figure 7 when α_2 is not in the same unit square as the starting point which sometimes could be the situation. Changing unit square in the first step corresponds to that at least one of the coordinates is outside the interval $[0, 1[$. In Figure 7 the second component is negative. The algorithm will run until the complete set of ten optimal data points, $[\alpha_1, \alpha_2, \alpha_3 \dots]$, is obtained.

F. Constructing the smoothing spline on the torus

From the algorithm in 3.5.1 an optimal set of points in R^2 is obtained and stated as $[\alpha_1, \alpha_2, \alpha_3 \dots]$. The coordinates of the points do not have to be in the interval $[0, 1[$.

So far only scalar output and scalar input splines have been discussed. When constructing a spline on the torus the data points and the torus points have two coordinates. However the spline is constructed in a similar way. The difference is that the spline is generated from two different control systems on the same form as the one in Equation 6. In order to construct the spline the optimal dataset is considered component wise. From this two separate data sets, called the x-set and the y-set, two



splines can be constructed with standard methods. The resulting smoothing spline is called the x -spline and the y -spline. An example can be seen in Figure 8. This component wise construction of the spline on the torus is justified in this special case because of some certain properties of the cost function, that will be discussed later in this section. The two control systems generating the spline on the torus is stated in Equation 10 and Equation 11.

$$\begin{aligned} \dot{z}_1(t) &= \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} z_1(t) + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u(t) \\ x(t) &= \begin{pmatrix} 1 & 0 \end{pmatrix} z_1(t) \end{aligned} \quad (8)$$

$$\begin{aligned} \dot{z}_2(t) &= \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} z_2(t) + \begin{pmatrix} 0 \\ 1 \end{pmatrix} v(t) \\ y(t) &= \begin{pmatrix} 1 & 0 \end{pmatrix} z_2(t) \end{aligned} \quad (9)$$

The cost function for the spline on the torus is stated as:

$$\begin{aligned} J(u, v) &= \rho \int_0^T (u(t)^2 + v(t)^2) dt + \\ &\quad \omega_1 \sum_{i=1}^N (x(t_i) - \alpha_i^1)^2 + \\ &\quad \omega_2 \sum_{i=1}^N (y(t_i) - \alpha_i^2)^2 \\ &\quad \omega, \rho > 0 \end{aligned}$$

The cost function can easily be rewritten into two different cost functions, where each function

depends on a scalar input:

$$J(u) = \rho \int_0^T u(t)^2 dt + \omega_1 \sum_{i=1}^N (x(t_i) - \alpha_i^1)^2$$

$$J(v) = \rho \int_0^T v(t)^2 dt + \omega_2 \sum_{i=1}^N (y(t_i) - \alpha_i^2)^2$$

$$\omega, \rho > 0$$

This is the certain property of the cost function that will allow a separate construction of the spline for each coordinate, as prior mentioned. Since

$$\min_{u,v} J(u, v) = \min_u J(u) + \min_v J(v) \quad (10)$$

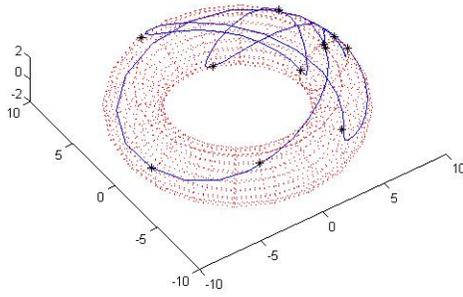
the minimization can be preformed for u and v separately. From Equation 5 it is clear that u corresponds to the first coordinate of the data points and v corresponds to the second. Hence the two optimal smoothing splines obtained from the points in R corresponding to u and v respectively will be optimal even when recalculated to the torus. In order to obtain these two splines standard single input and single output splines are constructed. An optimal cost problem subject to the cost function in Equation 13 and the control system in Equation 10 is solved in standard methods as in section 3.4. The output $x(t) \in R$ is as before a smoothing spline. The second optimal cost problem with the cost function in Equation 14 subject to the control system in Equation 11 is solved in the same way as the first and results in another smoothing spline $y(t) \in R$. These are actually the two splines, the x -spline and the y -spline, shown in Figure 8.

1) *Recalculation to torus coordinates:* It is an easy matter to construct a smoothing spline on the torus once the two splines in Figure 8 is obtained. In order to get the spline on the torus the two splines has to be coordinated and the points recalculated as points on the two dimensional torus. At first a transformation from points in R to angles is done via Equation 5. Hence two sets of angles u and v are obtained. In order to plot this angles on the torus the parametrization in Equation 3 is used. The resulting spline is shown in Figure 9.

G. The optimality of the constructed spline

1) *A comparison to random sets of points:* As prior mentioned, there are an infinite number of ways to choose the data points α_i . In subsection

Fig. 3. Spline on the torus

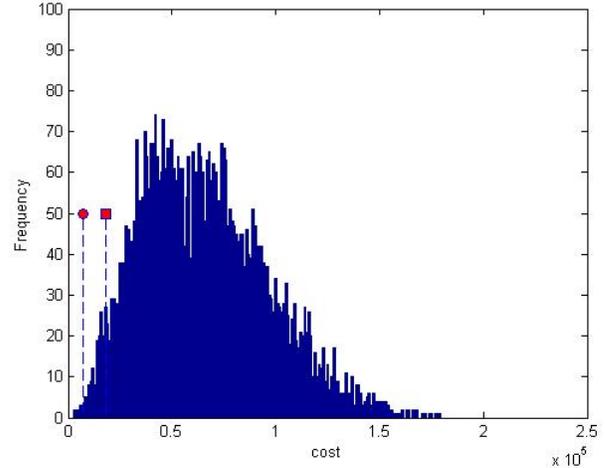


3.5 an attempt to minimize the energy of the input signal is done. However it is not likely that it is the optimal one. First of all the algorithm in Section 3.5.1 does not take into consideration that the resulting spline will be smoothing and second the algorithm only compare the energy from a given point to the nine nearest points. In order to estimate the optimality of that data set, enough many randomly chosen data sets is obtained via an algorithm that will be described. The input energy that is needed in order to produce the wanted spline with respect to the randomly chosen data sets are compared to the energy that is needed when producing the spline in subsection 3.6. The cost function used is the one in Equation 16.

$$J(u) = \int_0^T u^2(t) dt \quad (11)$$

The only difference to the cost function in Equation 8 is that the final time is set to T instead of t_β . T is the total time that the spline will exist. The algorithm of finding a random selection of points, starts with a random selection of ten coordinates, $\alpha_1^1, \beta^1, \gamma^1 \dots$ in the interval $[0, 1]$. This is of course also the points that will underlie the attempt of an optimal data set in Section 3.5.1. The first of these numbers, α_1^1 , is as before set to be fixed. Then a random choice of any component in the vector $v = [-1, 0, 1]$ called ε_1 is added to the second number, $\alpha_2^1 = \beta^1 + \varepsilon_1$. In order to get α_3^1 the ε_1 is kept and an other random choice in the vector v , called ε_2 is added. $\alpha_3^1 = \gamma^1 + \varepsilon_1 + \varepsilon_2$. The algorithm continues until ten numbers are received. Since the points of the torus are two dimensional,

Fig. 4. A histogram of the cost for 5000 randomly chosen datasets.



the algorithm is repeated for the second component of the ten random points. When the algorithm is finished a random set of ten data points in R^2 is obtained. To get a statistical acceptable result, the costs in Equation 16, is calculated for 5000 sets of randomly selected data points. A typical result is shown in a histogram in Figure 10.

2) *Discussion of the optimality:* The estimation of an optimal set of points from Section 3.5.1 gives a cost among the ten lowest costs, in the lower 0.2%, but it is not the optimal one. The cost is shown in Figure 10 as a circle. If the points are all selected to be in the unit square, in other words if all $\varepsilon_i \equiv 0$, the cost is in the lower 3% of the statistics. This is marked as a square in Figure 10. However the cost for the points in the unit square is in this case about twice as big as the cost from subsection 3.5.1 and the mean value of the costs is about ten times as big as that cost. Finally, to summarize the cost statistics in Figure 10, it is clear that the first data set does not give rise to an optimal spline on the torus. This was expected, since some restrictions in the algorithm was made. However constructing the spline subject to that first data set is more efficient than if the points were taken directly from the unit square without considering the infinite number of points mapping on one single point on the torus.

III. CONCLUSIONS

In this paper a suboptimal smoothing spline on the torus was constructed, based on M. Egerstedt's Matlab code for smoothing splines in R . However, this code had to be adjusted to suit the torus. When it comes to the torus the set of data points that the spline was supposed to pass optimally close to is not a unique set. This follows from the periodicity of the points on the torus. Hence the choice of an optimal data set became a part of the construction of the smoothing spline. The algorithm used when choosing this set resulted in an efficient smoothing spline, but not optimal. There were some restrictions in the algorithm that explains the non optimality.

An interesting future problem is to try to construct a more efficient algorithm in order to find the optimal data set when constructing the smoothing spline on the torus. One possible way is to involve dynamic programming.

REFERENCES

- [1] M. Egerstedt and C. Martin, Control theoretic splines: optimal control, statistics and path planning, Princeton Unit Press, 2010.
- [2] M. Egerstedt, Smoothsplines, personal communication.
- [3] S. Gomes, A. Kushpel, J. Levesley and D. Ragozin, Interpolation on the Torus using sk-splines with Number Theoretic Knots, University of Leicester, Technical reports, 1997.
- [4] J. Saranen and J. Anttila, The boundary element spline collocation for nonuniform meshes on the torus, In *j-NUM-MATH*, vol 106, pages 289-302, 2007.
- [5] M. Hofer and H. Pottmann, Energy-minimizing splines in manifolds, *ACM Transactions on Graphics*,23(3), pages 284293, 2004.
- [6] M. Egerstedt, C. Martin and Y. Zhou, Optimal approximation of functions, *Communication in Information and Systems*, 1(1), pages 101112, 2001.
- [7] Y. Zhou, W. Dayawansa and C. Martin, Control theoretic smoothing splines are approximate linear filters, *Communications in Information and Systems*, 4, pages 253272, 2004.
- [8] Y. Zhou, M. Egerstedt and C. Martin, Control theoretic splines with deterministic and random data, In *IEEE Conference on Decision and Control*, Seville, Dec. 2005.
- [9] G. Wahba, Spline models for observational data, In *CBMS-NSF Regional Conference Series in Applied Mathematics*, 59, Society for Industrial and Applied Mathematics, Philadelphia, 1990.