



Delft University of Technology

System Identification using the Multivariate Simplotope B-Spline

Visser, Tim; de Visser, Coen; van Kampen, Erik-Jan

Publication date

2016

Document Version

Accepted author manuscript

Published in

Papers 12th Pegasus-AIAA student conference

Citation (APA)

Visser, T., de Visser, C., & van Kampen, E.-J. (2016). System Identification using the Multivariate Simplotope B-Spline. In *Papers 12th Pegasus-AIAA student conference: Valencia, Spain*

Important note

To cite this publication, please use the final published version (if applicable). Please check the document version above.

Copyright

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy

Please contact us and provide details if you believe this document breaches copyrights. We will remove access to the work immediately and investigate your claim.

System Identification using the Multivariate Simplotope B-Spline*

T. Visser,[†] C. C. de Visser[‡] and E. van Kampen[§]

Delft University of Technology, P.O. Box 5058, 2600 GB Delft, The Netherlands

In recent research efforts the multivariate simplex spline has shown great promise in system identification applications. It has high approximation power, while its linearity in the parameters allows for computationally efficient estimation of the coefficients. In this paper the multivariate simplotope spline is derived from this spline, and compared to its simplex counterpart in a system identification setting. Contrary to the simplex spline, the simplotope spline allows the user to incorporate expert knowledge of the system in his models. Whereas in the first spline all variables are included in a complete polynomial, in the latter the user can split the variables in decoupled subsets. By fitting models to specifically designed test functions it is shown that this can indeed improve the approximation performance in terms of both the error metrics and the number of B-coefficients required. This comes at the price of a higher total degree, and therefore an increased sensitivity to Runge's phenomenon in case of poor data distribution. Finally an attempt is made to apply the proposed methods to a set of flight data of the DelFly II, a flapping wing micro aerial vehicle. It is found that the used data set is not suitable for global system identification, as the data is concentrated in low-dimensional clusters in the five-dimensional state space. Therefore it is advised that a more suitable data set is obtained to validate the simplotope spline in a system identification setting.

I. Introduction

Due to the introduction of Uninhabited Aerial Vehicles (UAV), the possibilities of testing new flight concepts have been greatly increased. First the quadrotor was widely implemented to serve as a testbed for, among others, control and automation schemes. These were quickly followed by a wide range of new concepts, among which even flapping wing aircraft. At the same time many UAVs were miniaturized, leading to the Micro Aerial Vehicle (MAV) concept.

Both the miniaturization and the flapping wing lift generation pose new challenges in modeling MAV behavior. To describe and analyze their flight performance a system identification effort is required. That is, based on measurement data from flight tests a mathematical model should be made that describes the behavior of the MAV. For such efforts many different methods have been proposed in the past. Whereas industry mainly structures the measurements in data tables, many research groups have embraced artificial neural networks to approximate the underlying dynamics. Recently a new method was introduced which uses multivariate simplex B-splines instead of neural networks.¹⁻³ These are piecewise polynomials defined on triangulations of a domain, combined to form a smooth function.⁴ The main benefit of this new technique is the fact that these splines are linear in the parameters, allowing for the use of efficient least-squares solvers.⁵

One of the disadvantages of the simplex spline, however, lies in the triangulation of the state space on which the spline is based. Although intensively researched in two-dimensional space, optimal ways of triangulating higher-dimensional spaces are still unknown.⁶⁻⁸ On top of that simplices tend to take on more star-like shapes as the dimension increases.⁹ This makes it significantly harder to evenly distribute data points over these domain elements.

*An earlier version of this paper was presented at the 2015 AIAA Atmospheric Flight Mechanics Conference, under the title *Quadrotor System Identification using the Multivariate Multiplex B-Spline*.

[†]PhD Candidate, Astrodynamics Department, t.visser-1@tudelft.nl.

[‡]Assistant Professor, Control and Simulation Department, c.c.devisser@tudelft.nl, AIAA Member.

[§]Assistant Professor, Control and Simulation Department, e.vankampen@tudelft.nl, AIAA Member.

Recently a variant of the multivariate simplex spline was proposed by Govindarajan et al. to solve the Hamilton-Jacobi-Bellman equation, central in reinforcement learning and optimal control.^{10,11} They use a univariate spline to approximate the dependence on time of the coefficients of a multivariate spline, to simplify the expressions for partial derivatives.

In this paper we introduce a new type of spline that generalizes these tensor-product simplex splines. Effectively this generalization effort comes down to identifying the new domain element. Whereas the simplex spline is defined on a triangulation consisting of simplices, the resulting multivariate simplotope spline is defined on a more general tessellation consisting of simplotopes. These are polytopes that result from geometric products of simplices. In the past they were mainly studied for their role in branches of game theory and operations research, and as an object for triangulation research.^{12–14} Due to the difference in geometric basis, both splines also consist of different types of polynomials.

The goal of this paper is to introduce the multivariate simplotope spline and its building blocks, and to investigate its behavior in a system identification setting. For this purpose several test functions are used, as well as an actual dataset of in-flight measurements of the DelFly, the flapping wing MAV of the Delft University of Technology. Whereas the first tests are specifically designed to highlight the differences between the simplex and simplotope splines, the latter is an attempt to for the first time apply the simplotope spline in a real environment.

To properly compare the simplotope and simplex splines, a major part of this article is devoted to their mathematical definition. More specifically, in section II the simplotope spline is derived from the simplex spline in a step-by-step manner. In the process, the simplex spline will be described for those who are unfamiliar with the concept. In section III the approximation performance of both splines is evaluated and compared. The system identification effort performed on the DelFly dataset is described in section IV. Finally in section V the conclusions of the overall process are drawn, and recommendations for further research are made.

II. The simplotope spline framework

The origin of the simplotope spline lies in the tensor-product simplex spline of Govindarajan.¹⁰ That is, coefficients of one spline are represented by a spline in other variables. The simplotope spline results when this process is performed on a per-domain-element basis. The process and the effect on the defining elements of a simplex spline are discussed in this section. First, in subsection II.A the geometric basis of the simplotope spline is discussed. Then in subsection II.B the polynomials are characterized, followed by the coefficients in subsection II.C. Finally in subsections II.D and II.E respectively, the methods for defining continuity conditions and scattered data fitting are presented.

II.A. The geometric basis

Per definition a spline consists of polynomials defined on parts of the total domain. Therefore first a description is provided of these parts, namely the simplex and the simplotope. In the process the barycentric coordinates are presented as a way of describing states relative to these domain elements. Finally our definition of a tessellation in introduced in this subsection.

An n -simplex Δ^n is defined as the convex hull of $n + 1$ non-degenerate vertices $\mathcal{V} = \{\mathbf{v}_0, \dots, \mathbf{v}_n\}$, $\mathbf{v}_j \in \mathbb{R}^n$. Any point in \mathbb{R}^n can be uniquely defined using a vector of $n+1$ barycentric coordinates $\mathbf{b} := (b_0, \dots, b_n) \in \mathbb{R}^{n+1}$ relative to this set of vertices, such that $\sum_{j=0}^n b_j = 1$. For a general point $\mathbf{x} \in \mathbb{R}^n$ we have

$$\mathbf{x} = \sum_{j=0}^n b_j \mathbf{v}_j. \quad (1)$$

If $b_j \geq 0, \forall j \in [0, n]$, then \mathbf{x} lies in the simplex described by \mathcal{V} .

A ν -simplotope Γ^ν , with $\nu = (\nu_1, \dots, \nu_\ell) \in \mathbb{N}^\ell$ a multi-index, is defined as the product of the ℓ simplices of dimension $\nu_i, i \in [1, \ell]$. That is,

$$\Gamma^\nu = \Delta^{\nu_1} \times \dots \times \Delta^{\nu_\ell}. \quad (2)$$

The one-norm of the multi-index $|\nu| = n$ is equal to the dimension of the simplotope. For clarity the defining simplices Δ^{ν_i} are named *layers*. Note that a single-layer simplotope is a simplex, and therefore the simplex

is here considered to be a special case of the simplotope.

The product in the definition of the simplotope implies that the resulting polytope has vertices \mathbf{u}_ϕ , with $\phi \in \mathbb{N}^\ell$ again a multi-index, at linear combinations of the vertices $\mathbf{w}_{i\phi_i}$ of the layers. Therefore a point $\mathbf{x} \in \mathbb{R}^n$ can be described, using a collection $\mathbf{a} = (\mathbf{b}_1, \dots, \mathbf{b}_\ell) \in \mathbb{R}^{n+\ell}$ of barycentric coordinates $\mathbf{b}_i = (b_{i0}, \dots, b_{i\nu_i})$ in all layers, as

$$\mathbf{x} = (1 - \ell)\mathbf{w}_0 + \sum_{i=1}^{\ell} \sum_{j=0}^{\nu_i} b_{ij} \mathbf{w}_{ij}. \quad (3)$$

In the above, \mathbf{w}_0 represents the origin point that is shared by all layers. In a simplotope the barycentric coordinates sum to ℓ , as \mathbf{b}_i sums to one for all $i \in [1, \ell]$.

Multiple n -dimensional simplotopes Γ_k^ν can be combined to form a tessellation. We define a tessellation \mathcal{T} as the collective of simplotopes, such that

$$\mathcal{T} = \bigcup_{k=1}^N \Gamma_k^\nu, \quad \Gamma_k^\nu \cap \Gamma_m^\nu \in \{\emptyset, \Gamma^{\nu-\varepsilon}\}, \forall k \neq m, \quad (4)$$

where ε is an ℓ -element multi-index for which $0 \leq \varepsilon_i \leq \nu_i, \forall i \in [1, \ell]$. That is, the elements of a tessellation can share lower-dimensional simplotopic faces, but do not overlap. Note that in this paper we will only consider homogeneous tessellations, in which ν is equal for all elements. Heterogeneous tessellations (or: mixed grids) have been studied in the two- and three-dimensional case by Chui and Lai,¹⁵⁻¹⁷ and in a general setting by Visser et al.¹⁸

Tessellations can be constructed by taking the product of triangulations in the layers, which comes down to a tensor product of all domain elements. This is the preferred method in high-dimensional problems, as it allows for constructing optimal triangulations in the layers.

II.B. B-form polynomial

Polynomials are defined on a simplex by taking the weighted sum of Bernstein basis polynomials defined in barycentric coordinates. The B-form of such a polynomial is¹⁹

$$p(\mathbf{b}) = \sum_{|\kappa|=d} c_\kappa B_\kappa^d(\mathbf{b}), \quad B_\kappa^d(\mathbf{b}) := \frac{d!}{\kappa!} \mathbf{b}^\kappa, \quad (5)$$

where c_κ are *B-coefficients*, discussed in the next subsection, and $\kappa = (\kappa_0, \dots, \kappa_n) \in \mathbb{N}^n$ is a multi-index for which $\kappa! := \prod_j \kappa_j!$ and $\mathbf{b}^\kappa := \prod_j b_j^{\kappa_j}$. Every possible permutation of κ results in a separate *Bernstein basis polynomial*. The complete set forms a basis for the space of all polynomials of degree d in n variables.⁴

If c_κ in Eq. (5) is replaced by a polynomial in the B-form, a tensor-product between the basis polynomials results. The degree is defined per layer, leading to a degree vector $\mathbf{d} = (d_1, \dots, d_\ell)$. The multi-indices κ from the layers are renamed and combined to form one multi-index $\lambda = (\lambda_1, \dots, \lambda_\ell)$, where $\lambda_i = (\lambda_{i0}, \dots, \lambda_{i\nu_i})$, with $|\lambda_i| = d_i$, is the multi-index in the i^{th} layer. Applying these changes to Eq. (6) we obtain

$$\pi(\mathbf{a}) = \sum_{\substack{|\lambda_i|=d_i, \\ \forall i \in [1, \ell]}} c_\lambda \mathcal{B}_\lambda^d(\mathbf{a}), \quad \mathcal{B}_\lambda^d(\mathbf{a}) := \prod_{i=1}^{\ell} B_{\lambda_i}^{d_i}(\mathbf{b}_i) = \frac{d!}{\lambda!} \mathbf{a}^\lambda. \quad (6)$$

The tensor-product basis polynomials \mathcal{B}_λ^d are scaled versions of higher-dimensional, total degree basis polynomials. If only one layers is used, we again end up with the simplex polynomials. Therefore the simplex polynomials can be seen as a special case of the simplotope polynomials.

To use efficient least squares solvers for estimating the coefficients, the basis polynomials are generally collected in a vector.

$$\mathcal{B}_\lambda^d(\mathbf{a}(\mathbf{x})) := \begin{cases} \mathcal{B}_\lambda^d(\mathbf{a}^{\Gamma_i}(\mathbf{x})) & \text{if } \mathbf{x} \in \Gamma_i \\ 0 & \text{otherwise.} \end{cases} \quad (7)$$

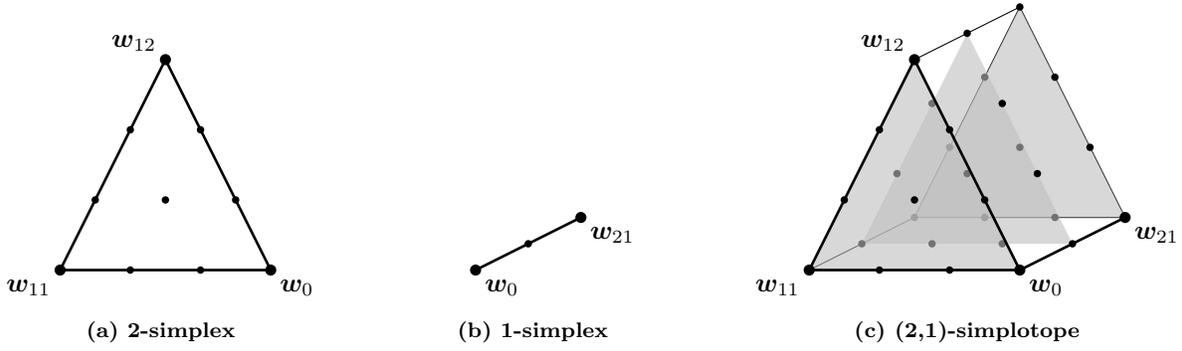


Figure 1: B-nets of the 2- and the 1-simplex, and the derivation of the (2,1)-simplotope B-net.

This vector can be obtained directly, or by taking the tensor-product of the vectors in the individual layers. The vectors can be collected in a matrix \mathbf{B} , such that the polynomial in Eq. 6 can be written as the matrix multiplication $\pi = \mathbf{B}\mathbf{c}$.

II.C. B-coefficients and the B-net

An important feature of the simplex spline is that the B-coefficients c_κ , introduced in Eq. (5) as the weights of the basis polynomials, have a spatial location in the simplex. They coincide with domain points, such that c_κ lies at $\mathbf{q}_\kappa \in \mathbb{R}^n$,

$$\mathbf{q}_\kappa = \frac{1}{d} \sum_{j=0}^n \kappa_j \mathbf{v}_j. \quad (8)$$

The collective of B-coefficients at these locations is called the B-net. It plays a central role in the definition of continuity conditions. Example B-nets in two and one dimension for degrees three and two can be found in figures 1a and 1b respectively.

The tensor-product in Eq. (6) is reflected in the B-net of the simplotope polynomials. Like the simplotope vertices, the domain points \mathbf{q}_λ can be found at the linear combinations of domain points in the layers.

$$\mathbf{q}_\lambda = (1 - \ell)\mathbf{w}_0 + \sum_{i=1}^{\ell} \frac{1}{d_i} \sum_{j=0}^{\nu_i} \lambda_{ij} \mathbf{w}_{ij}. \quad (9)$$

This results in a B-net consisting of parallel copies of the B-net of one layer, translated to the domain points of the other layers. This is illustrated in figure 1c by the parallel highlighted triangles.

Much like the basis polynomials, the B-coefficients are collected in a vector. We write $\mathbf{c}^{\Gamma_i} = (c_{\lambda}^{\Gamma_i})_{|\lambda_i|=d_i, \forall i}$, where the coefficients are ordered according to the multi-index in lexicographical order, and $\mathbf{c} = (\mathbf{c}^{\Gamma_1}, \dots, \mathbf{c}^{\Gamma_N})$. The superscript refers to the simplotope the coefficients belong to.

II.D. Continuity conditions

Two simplex polynomials defined on n -simplices Δ^n and $\tilde{\Delta}^n$ that share an $(n - 1)$ -edge can be joined with arbitrary continuity order $r < d$. The conditions to enforce this continuity are well-known from the simplex spline literature.^{1,4} From the work of Visser et al. it is concluded that continuity between two polynomials π and $\tilde{\pi}$ on simplotopes Γ^ν and $\tilde{\Gamma}^\nu$ can be defined per layer.¹⁸ This implies that the continuity conditions for simplex splines can be reused. By finding the conditions in each layer and then copying them for each parallel copy of the B-net (illustrated in Fig. 2), the total set of conditions is found. Rewriting the simplex continuity conditions, we find for the simplotope spline

$$\tilde{c}_{(m, \lambda_{11}, \dots, \lambda_{i\nu_i})} = \sum_{|\gamma|=m} c_{(0, \lambda_{11}, \dots, \lambda_{i\nu_i}) + \gamma} \mathcal{B}_\gamma^m(\mathbf{a}^\Gamma(\tilde{\mathbf{w}}_{10})), \quad \forall d : |\lambda_i| = d_i, \text{ and } \forall m \in [0, r], \quad (10)$$

where it is assumed that $\tilde{\mathbf{w}}_{10}$ is the out-of-edge vertex. A unique set of conditions is generated by the permutations of λ_1 , whereas the parallel copies of these conditions are generated by the different permutations of $\lambda_i, i \in [2, \ell]$.

Note that $\gamma = (\gamma_1, \dots, \gamma_\ell)$ in Eq. (10) contains many zeros by default, because any adjustment to λ that violates $|\lambda_i + \gamma_i| = d_i$ results in a non-existent B-coefficient. In other words, the degree cannot be changed in other layers than the one containing the out-of-edge vertex. In the case that the out-of-edge vertex lies in the first layer, we therefore have $\gamma = (\gamma_1, 0, \dots, 0)$.

The conditions in Eq. 10 are linear in the B-coefficients, and can therefore be written as the matrix multiplication of the smoothness matrix \mathbf{H} and the vector of B-coefficients \mathbf{c} .

$$\mathbf{H}\mathbf{c} = \mathbf{0}. \quad (11)$$

\mathbf{H} contains the same conditions as the ones in Eq. (10). It is a sparse matrix, because each row describes a single constraint between two simplices. That is, in each row a maximum of $1 + \frac{(d+n-1)!}{(d-1)!n!}$ elements are nonzero.¹

II.E. Fitting scattered data

The above can be combined into a single equality constrained least squares problem.

$$\begin{aligned} \min \|\mathbf{y} - \mathbf{B}(\mathbf{a}(\mathbf{X}))\mathbf{c}\| \\ \text{subject to: } \mathbf{H}\mathbf{c} = \mathbf{0}. \end{aligned} \quad (12)$$

In the above, $\|\bullet\|$ is the 2-norm, \mathbf{X} is a matrix containing the input data, and \mathbf{B} is the matrix of basis polynomials. The output vector is given as \mathbf{y} .

The solution of Eq. (12) can be found using the method of Lagrange multipliers. This is discussed in detail by De Visser et al.⁵ Because the basis polynomial matrix \mathbf{B} is block diagonal and \mathbf{H} is sparse, distributed solvers can be employed.

III. Approximation performance

From the previous discussion it is clear that the simplotope spline has a different structure than the simplex spline. In this section these differences will be described (in subsection III.A), and their effect on function approximation performance will be illustrated using test functions (in subsection III.B).

III.A. Simplotope spline characteristics

The two most striking differences between the simplotope and the simplex spline are their geometric basis and their polynomial structure, collectively referred to as the model structure. In practical applications these two properties can only be changed simultaneously, namely by changing the subdivision of variables over the layers. The consequences on approximation performance may however be conflicting. Therefore both effects are discussed in more detail below.

The most important difference between the simplex and the simplotope lies in the distribution of its content. With increasing dimension, the content of a simplex moves away from the center towards the corners. Therefore the polytope becomes less spherical and more star-like, making it more difficult to properly, uniformly fill a simplex with data points as the dimension increases. To be more precise, the corners extend far from the center of the simplex, where most of the data points lie. The lack of data near the corners and edges leaves the polynomials free to oscillate. Especially in system identification efforts where four or more variables are involved this has a detrimental effect on the approximation performance near the bounds of the domain.

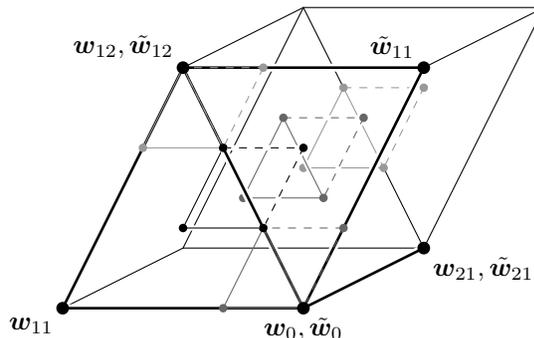


Figure 2: Continuity conditions between two (2,1)-simplotopes. All conditions on the foreground are shown, parallel copies only of the middle condition.

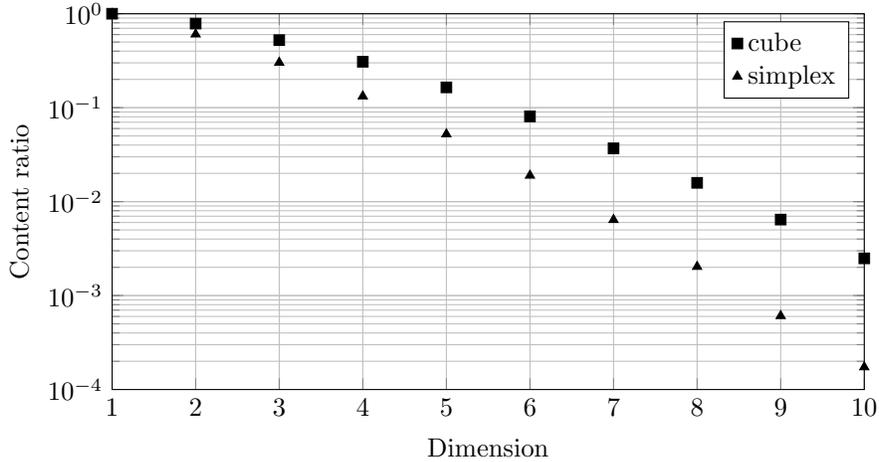


Figure 3: The ratio of the content of the inscribed sphere to that of the n -simplex and the n -cube as a function of dimension n .

The simplotope in general has a more spherical shape, making it less sensitive to this effect. This can be illustrated by plotting the content ratio of the inscribed sphere to the total content.⁹ The most spherical of simplotopes, according to this metric, is the hypercube. Therefore it is compared to the simplex in Fig. 3. Clearly the content remains more focused in the center of simplotopes than in that of a simplex. Therefore the simplotopes in general will be less sensitive to data distribution issues. However, it is clear from the comparison that this effect is quickly reduced as dimension increases. On top of that it will be less significant when fewer, higher-dimensional layers are chosen. Therefore, given a number of variables, this effect puts a lower bound on the amount of layers to use.

The difference in polynomial structure is clear from the definition in Eq. (6). Whereas a simplex polynomial is a complete multivariate polynomial, the simplotope polynomial consists only of tensor-product terms. The latter therefore allows for excluding high-degree terms of single variables, while allowing cross-terms of the same degree to occur. This can be beneficial if the states of the system are related to the output in fundamentally different ways.

A pitfall of the tensor-product polynomials however is the implicit total degree. When the diagonal of a simplotope is considered, the polynomials have a degree $d_t = |\mathbf{d}|$. This total degree can quickly rise when more layers are defined, leading to unwanted oscillations such as Runge’s phenomenon near the corners of the simplotope. For example, in a four-dimensional hypercube with quadratic polynomials in all layers, the total degree is eight. This effect therefore puts an upper bound on the amount of layers to use.

Combining the above described characteristics, it is clear that deciding on the model structure is not straightforward. In general we can say however that high-dimensional models can benefit from a larger amount of layers. In such cases however, the total degree should be kept low by using low-degree polynomials in all layers.

III.B. Function approximation performance

The simplotope spline characteristics described in the previous subsection can be illustrated by approximating test functions. To show the relevance of choosing a polynomial structure, splines will be fitted to a dedicated test function. After that the effect of data distribution is investigated by changing the shape of the data set beneath two comparable splines.

To test the effect of different polynomial structures, the following test function is used to generate a uniformly, yet randomly distributed data set.

$$y = x_3^2 \sin(1.4\pi x_4) F(2x_1 - 1, 2x_2 - 1), \quad \forall k : x_k \in [0, 1] \quad (13)$$

In equation (13), $F(x, y)$ is the Franke function. No noise was added to the generated data sets. Within the indicated domain the function values lie in the range $-1 \leq y \leq 1.4$.

In Eq. (13) three layers can easily be identified, namely the Franke function in (x_1, x_2) , a parabola in x_3 , and a sine in x_4 . Based on this structure, the following splines will be investigated: 4-simplex splines, (2,1,1)-

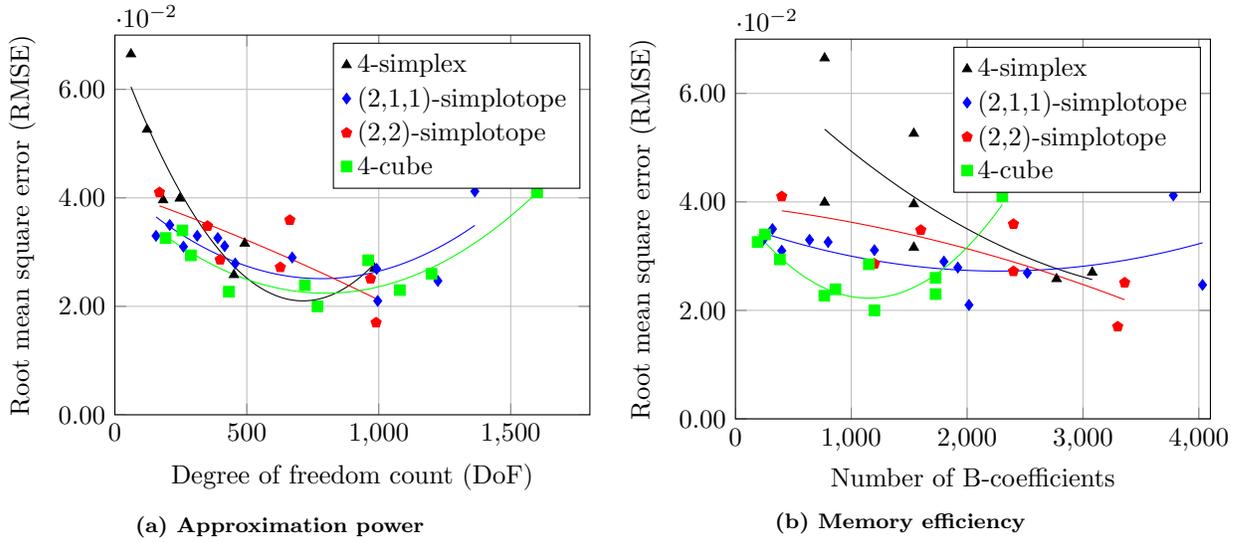


Figure 4: The approximation performance of four different types of splines for a wide range of tessellation densities and polynomial degrees.

simploptope splines, (2,2)-simploptope splines, and 4-cube (or: (1,1,1,1)-simploptope) splines. For each type of spline a wide range of degrees, continuity orders and tessellation densities is used to model the test function, bounded by high computation times at more than 4000 B-coefficients. To quantify the approximation performance of each of the splines, the RMSE (Root Mean Square Error) is computed based on a separately generated verification data set. This is done 10 times for different splits of the data set to verify a constant result. Because of the omission of noise, the RMSE can be seen as a measure of approximation power.

The mean absolute RMSE values are plotted against the number of degrees of freedom (DoF, the difference between the number of B-coefficients and the rank of the smoothness matrix H) in Fig. 4a. As an overall trend, all splines but the (2,2)-simploptope spline show the familiar quadratic trend of approximation within the plotted range. First the added DoF improve the fit, then at too high polynomial degrees overfitting occurs. The projected optimal performance lies around the same amount of DoF for all splines but the (2,2)-simploptope spline. This implies that more than the polynomial structure, the number of degrees of freedom defines the approximation performance. At the same time however, the simplex spline is outperformed by all the other splines at almost all DoF counts.

Plotting the RMSE against the amount of B-coefficients, as in Fig. 4b, similar trends can be observed. Most striking however is the small size of most simploptope splines. It is likely that smaller simplex splines would have even worse performance than the ones plotted, as the degree would have to be decreased to two. The (2,1,1)-simploptope and 4-cube spline on the other hand reach adequate RMSE values below 1000 B-coefficients, with the optimum for the latter already at 1200 B-coefficients. The locations of the optima in this case clearly indicate a benefit for splines with more layers. This is most likely caused by the fact that with increasing dimension the continuity conditions become more intertwined. Referring to Fig. 2, each two-dimensional condition overlaps with the next condition in the same layer, but the parallel copies in the direction of the second layer are completely separate.

To test the effect of the data distribution on the approximation performance, the following test function is used.

$$y = \sin(2\pi x_1) + e^{x_2}, \quad x_1, x_2 \in [0, 1]. \quad (14)$$

A simple function is used, so as to not favor one of the polynomial structures. By using a single square as the basis for the bicubic (1,1)-simploptope, and two triangles as a basis for the cubic 2-simplex spline, both splines have the same amount of DoF when zeroth order continuity is enforced.

In Fig. 5 the approximation performance of these two splines on the test function is plotted for different data sets. On the x-axis the RMSE on a verification set inside the indicated data set range is plotted. The y-axis represents the same metric on a data set outside this range; that is, inside the inverse set. We can identify the uniform set, the diamond (leaving the vertices of the domain open), the inscribed circle, the inscribed

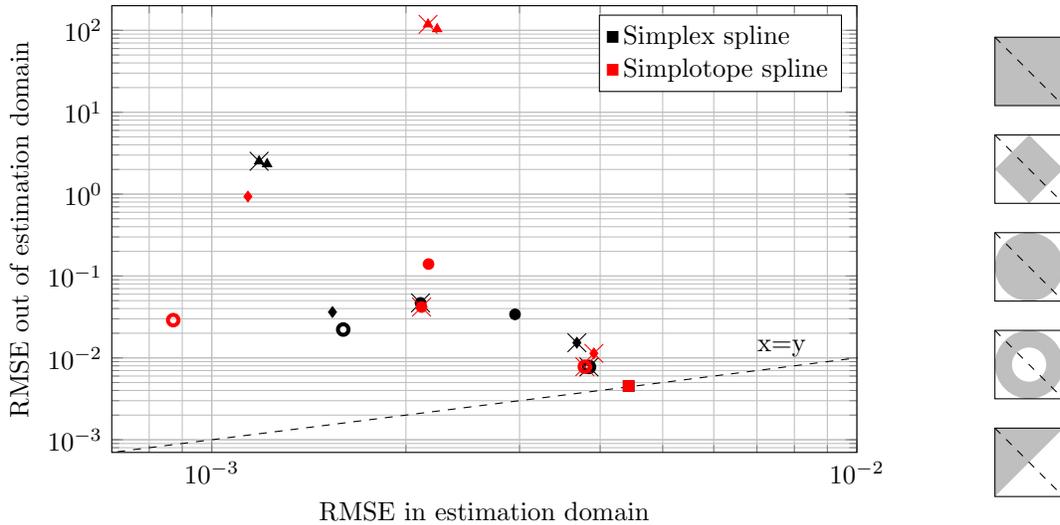


Figure 5: The approximation performance of the simplotope and simplex spline on the indicated data sets. Data sets in the legend to the right are highlighted in gray, the inverse datasets are white, and the dashed line indicates the hypotenuse of the simplices.

ring (with the inner radius half the outer radius), and the triangle (with the hypotenuse perpendicular to that of the simplices). These sets are displayed next to the plot in Fig. 5 for reference. The marks in the plot with a cross reflect the spline performance on the inverse set, where the notion of inside and outside the data range is inverted. Note that the performance on the uniform set is equal for both splines, as intended.

First of all we may note that all splines fit data better that lies inside the data range used for estimating the coefficients. The simplotope spline in this sense results in a smaller RMSE than the simplex spline. Considering data points outside this range however, the simplex spline in general performs better. This is especially clear in the estimation sets that keep the vertices open (diamond, circle, and ring). This is caused by the high total degree of the simplotope spline, introducing oscillations outside of the convex hull of the data set. These extremities are used to make a better fit inside the estimation domain, yet are not penalized for poor local estimation performance.

Second the performance on the inverse data sets clearly shows the importance of data being available near the edges of the domain. Especially the inverse diamond and ring domain show much better performance overall than their original counterparts.

IV. DelFly system identification

In the previous section it was shown that the simplotope spline may have several advantages over the simplex spline in system identification applications. To see up to what extent these advantages occur in real applications, a system identification effort is undertaken for a flapping wing MAV, the DelFly. In subsection IV.A the DelFly itself and its flight data are described. Then in subsection IV.B the approach and results of making spline models based on this data set are discussed.

IV.A. DelFly flight data

The DelFly II is a 16 gram, 28 centimeter span flapping wing aircraft with four wings and a conventional tail. Control is provided through a conventional elevator on the horizontal tail surface, a rudder on the vertical tail, and the flap frequency of the main wings. Depending on the center of gravity position the aircraft is capable of a slow forward and backward flight with an almost vertical attitude, and a fast forward flight with horizontal attitude.

The flight data of the DelFly is recorded by an optical tracking system in a test chamber of the Micro Air Vehicle Integration and Application Research Institute (μ AVIARI) test chamber, part of the US Air Force Research Laboratories. A system of cameras tracks the position of markers on the test subject at a frequency of 200Hz. The relative position of the markers is used to derive the attitude, whereas a three point

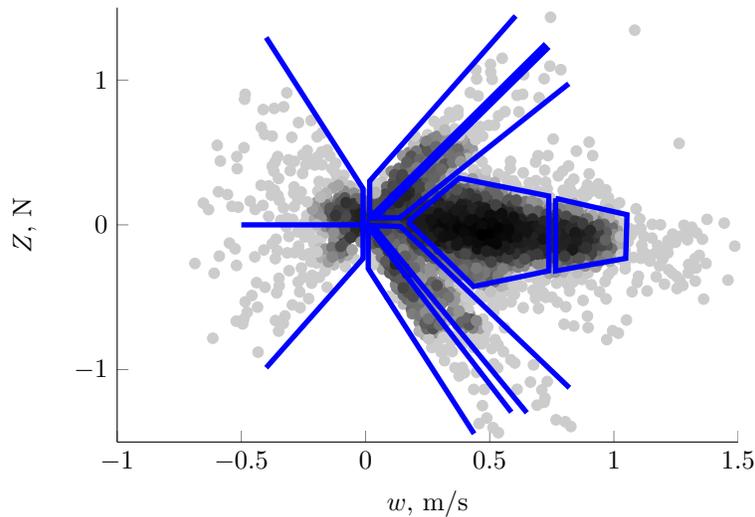


Figure 6: Projection of the DelFly data set onto the (w, Z) -plane. The local density of data is indicated by the color of the dots, regions of equal state are indicated with thick borders.

difference method is used to arrive at the velocities and accelerations.²⁰

The data set was intended for linear system identification.²¹ Therefore it consists of a large number of instances of the same elevator induced maneuver, in which the DelFly pitches up, stalls, falls in a near-vertical orientation, and then recovers. To reduce the dimensionality of the system, all episodes of lateral motion such as turns were removed from the data set. The remaining longitudinal motion will be described using the velocity components u and w , the pitch angle θ and rate q , and the flap phase ζ . The flap rate was found to be one-to-one related to the phase, so it is excluded from the analysis. As outputs the body forces X and Z are investigated. In the slow, almost hovering flight performed in the tests, the body x -axis, aligned with the body from tail to nose, is almost vertical. The z -axis points primarily in the direction of flight, perpendicular to the body.

A plot of the data density, projected onto the (w, Z) -plane, is shown in Fig. 6. A set of similar plots was used to analyze the distribution of data over the state space. From this analysis it was found that due to the performed maneuvers, the data distribution is highly non-uniform. For example, in Fig. 6 regions are marked with borders in which all data points reflect essentially the same state. That is, all data points in one of the highlighted regions in this projection, will also comprise a separate region in the entire, five-dimensional state space.

IV.B. Multivariate spline models

The data analysis described in the previous subsection suggests that constructing a global model using the provided data set will be difficult. This can be illustrated with a lower-dimensional analogy. Consider the situation of having a two-dimensional data set, clustered along a line through a two-dimensional state space. In fitting a surface to such a data set, one can expect a good approximation of the dynamics of the system along the line. Perpendicular to the line however, it is likely that little dynamics are present in the data to actually model system behavior. Therefore it is unclear what is being modeled in that direction. On top of this, the surface will be unconstrained away from the line, increasing the risks of overfitting to noise.

To test the above hypothesis, several simplex and simplotope splines were fitted to the described data set, both to model the X - and the Z -force. Here only the latter will be discussed in detail, as the conclusions for the first are similar. Because little is known of the optimal model structure for flapping wing aircraft, a large variety of splines was fitted to the data. The best splines in terms of RMSE are collected in table 1. Note that the data set is split in two: one to estimate the coefficients and one to assess the approximation quality. This is done for 10 different subdivisions of the data for each spline, resulting in a mean and standard deviation of the RMSE values.

Comparing the range of the actual force (nominally $-0.2 \leq Z \leq 0.2N$) to the RMSE, it must be concluded that all splines poorly approximate the data. As the 1σ confidence intervals of most splines overlap, it is also

Table 1: Approximation performance in terms of error metrics and number of coefficients and degrees of freedom of the spline models of the DelFly Z-force. The IDs refer to simplotope (T), simplified simplotope (S), simplex spline (X) and polynomial (P) models.

ID	Layers	Degrees	Coefficients	DoF	RMSE
T1	$(u, w), (\theta, q), \zeta$	(1,1,2)	702	140	0.144 ± 0.005
T2	$(u, w), \theta, q, \zeta$	(1,1,1,2)	540	142	0.141 ± 0.004
T3	$u, w, (\theta, q), \zeta$	(1,1,1,2)	576	144	0.143 ± 0.004
T4	u, w, θ, q, ζ	(1,1,1,1,2)	384	144	0.142 ± 0.004
S5	w, θ, q, ζ	(1,1,1,2)	192	72	0.147 ± 0.005
S6	$w, (\theta, q), \zeta$	(1,1,2)	288	72	0.148 ± 0.003
S7	$(w, q), \zeta$	(1,2)	72	24	0.157 ± 0.004
S8	$(u, w), \zeta$	(1,2)	72	24	0.157 ± 0.004
X1	(u, w, θ, q, ζ)	1	564	28	0.167 ± 0.006
X2	(u, w, θ, q, ζ)	2	1554	192	0.148 ± 0.005
P1	(u, w, θ, q, ζ)	1	6	6	0.184 ± 0.005
P2	(u, w, θ, q, ζ)	2	21	21	0.162 ± 0.004
P3	(u, w, θ, q, ζ)	5	252	252	0.140 ± 0.004

impossible to compare the performance of the different types of splines. It is however clear that having more layers reduces the number of B-coefficients without affecting the DoF, as was found before in subsection III.B.

V. Conclusion

In this paper the multivariate simplotope spline was presented and introduced as a system identification tool. It has a different geometric basis than the more established simplex spline, and consequently a different polynomial structure. Still, continuity conditions ensuring smooth joins between domain elements can be derived from those defined in the simplex spline framework, and the same constrained least squares problem must be solved to estimate the B-coefficients. Like in the simplex spline, the basis polynomials of the simplotope spline provide a stable local basis.

A series of tests was performed to investigate the behavior of the simplotope spline, and compare it to the simplex spline. It was found that the simplotope in general has a better content distribution for data encapsulation than simplices, especially in high-dimensional space. This may reduce the risk of having insufficient data points in domain elements, causing the problem to become singular. On the other hand the introduction of multiple layers often increases the total degree of the polynomial, making the simplotope spline more sensitive to Runge's phenomenon. Future efforts will focus on applying differential constraints on the boundaries of the domain, and on using mixed tessellations to combine the best of both domain elements. Finally the increased flexibility in choosing the polynomial structure allows for a better integration of knowledge of the system dynamics into the model structure, although this remains to be proven in a practical application.

The attempt at constructing a global model for the DelFly II has proven the importance of proper flight test design. Although the data set has proven its value in the past in linear system identification efforts, a more complete data set is required for the estimation of a multivariate spline model. Only by performing a varied set of maneuvers designed specifically to excite coupled modes can lead to such a data set. Recent research into the accelerometer data of ESA's GOCE satellite suggests that this mission may provide such a data set.

Acknowledgments

The first author would like to thanks his supervisors, Coen de Visser and Erik-Jan van Kampen, for their support throughout this Master thesis project.

References

- ¹de Visser, C. C., *Global nonlinear model identification with multivariate splines*, Ph.D. thesis, Delft University of Technology, July 2011.
- ²de Visser, C. C., Chu, Q. P., and Mulder, J. A., “Differential constraints for bounded recursive identification with multivariate splines,” *Automatica*, Vol. 47, July 2011, pp. 2059–2066.
- ³Tol, H. J., de Visser, C. C., van Kampen, E., and Chu, Q. P., “Nonlinear multivariate spline-based control allocation for high-performance aircraft,” *Journal of Guidance, Control, and Dynamics*, 2014, in press.
- ⁴Lai, M. and Schumaker, L. L., *Spline functions on triangulations*, Encyclopedia of mathematics and its applications, Cambridge University Press, 2007.
- ⁵de Visser, C. C., Chu, Q. P., and Mulder, J. A., “A new approach to linear regression with multivariate splines,” *Automatica*, Vol. 45, No. 12, December 2009, pp. 2903–2909.
- ⁶Ruppert, J., “A Delaunay refinement algorithm for quality 2-dimensional mesh generation,” *Journal of Algorithms*, Vol. 18, No. 3, 1995, pp. 548–585.
- ⁷Bern, M., Eppstein, D., and Gilbert, J., “Provably good mesh generation,” *Journal of Computer and System Sciences*, Vol. 48, No. 3, 1994, pp. 384–409.
- ⁸Bern, M. and Eppstein, D., “Mesh generation and optimal triangulation,” *Computing in Euclidian Geometry*, edited by D. Du and F. Hwang, World Scientific Publishing, Singapore, 1992, pp. 23–90.
- ⁹Tarantola, A., *Inverse problem theory and methods for model parameter estimation*, Society for Industrial and Applied Mathematics, 2005.
- ¹⁰Govindarajan, N., de Visser, C. C., and Krishnakumar, K., “A sparse collocation method for solving time-dependent HJB-equations using multivariate B-splines,” *Automatica*, Vol. 50, No. 9, September 2014, pp. 2234–2244.
- ¹¹Govindarajan, N., de Visser, C. C., van Kampen, E., Krishnakumar, K., Barlow, J., and Stepanyan, V., “Optimal Control Framework for Estimating Autopilot Safety Margins,” *Journal of Guidance, Control, and Dynamics*, 2014, in press.
- ¹²Freund, R. M., “Combinatorial theorems on the simplotope that generalize results on the simplex and cube,” *Mathematics of Operations Research*, Vol. 11, No. 1, February 1986, pp. 169–179.
- ¹³Talman, A. J. J., “Intersection theorems on the unit simplex and the simplotope,” *Imperfections and Behavior in Economic Organizations*, edited by R. P. Gilles and P. H. M. Ruys, Vol. 11 of *Theory and Decision Library*, chap. 11, Springer Netherlands, November 1991, pp. 257–278.
- ¹⁴Loera, J. A. D., Rambau, J., and Santos, F., *Triangulations; structures for algorithms and applications*, No. 25 in Algorithms and computation in mathematics, Springer-Verlag, Berlin Heidelberg, 2010.
- ¹⁵Chui, C. K. and Lai, M. J., “On bivariate super vertex splines,” *Constructive Approximation*, Vol. 6, No. 4, December 1990, pp. 399–419.
- ¹⁶Chui, C. K. and Lai, M. J., “Multivariate vertex splines and finite elements,” *Journal of Approximation Theory*, Vol. 60, No. 3, March 1990, pp. 245–343.
- ¹⁷Lai, M. J., *On Construction of Bivariate and Trivariate Vertex Splines on Arbitrary Mixed Grid Partitions*, Ph.D. thesis, Hangzhou University, Hangzhou, China, August 1989.
- ¹⁸Visser, T., de Visser, C. C., and van Kampen, E., “Towards the multivariate simplotope spline: continuity conditions in a class of mixed simplotopic grids,” *Under review*, 2016.
- ¹⁹de Boor, C., “B-form basics,” *G. Farin, editor, Geometric Modeling: Algorithms and New Trends*, 1987.
- ²⁰Caetano, J. V., de Visser, C. C., Remes, B. D. W., de Wagter, C., and Mulder, M., “Modeling a flapping wing MAV: flight path reconstruction of the Delfly II,” *Proceedings of the AIAA Modeling and Simulation Technologies Conference*, edited by J. Scharl, 2013, pp. 1–12.
- ²¹Caetano, J. V., de Visser, C. C., de Croon, G. C. H. E., Remes, B. D. W., de Wagter, C., Verboom, J., and Mulder, M., “Linear aerodynamic model identification of a flapping wing MAV based on flight test data,” *International Journal of Micro Air Vehicles*, Vol. 5, No. 4, 2013, pp. 273–286.