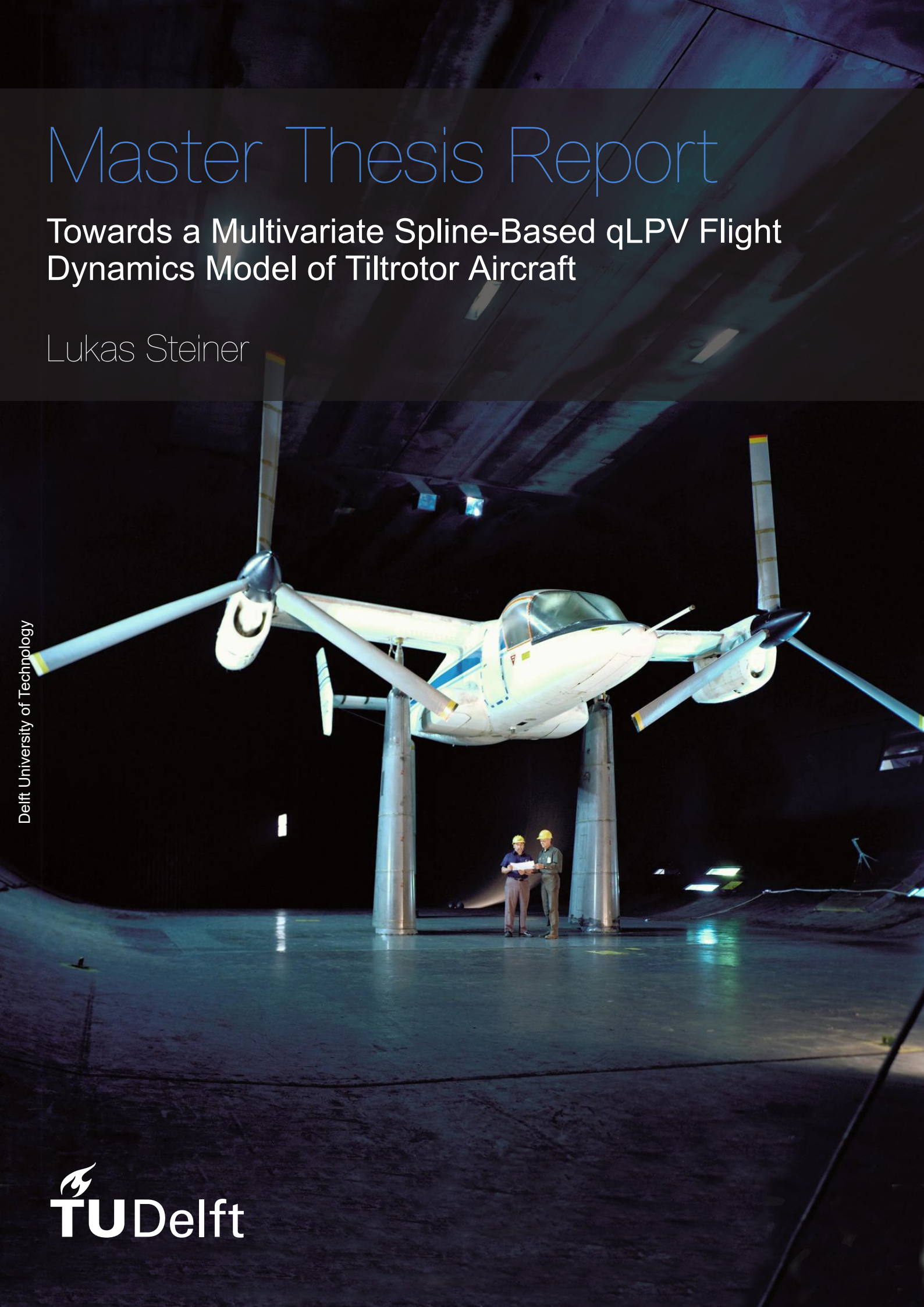
Master Thesis Report

Towards a Multivariate Spline-Based qLPV Flight Dynamics Model of Tiltrotor Aircraft

Lukas Steiner



Delft University of Technology

TU Delft

Master Thesis Report

Towards a Multivariate Spline-Based qLPV Flight Dynamics Model of Tiltrotor Aircraft

by

Lukas Steiner

to obtain the degree of Master of Science
at the Delft University of Technology.

Student Number: 4279832
Project Duration: February 2020 - February 2023
Thesis committee: Dr. Ir. C. C. de Visser, TU Delft, Supervisor
Dr. M. D. Pavel, TU Delft
Dr. Ir. W. van der Wal, TU Delft

TU Delft

Preface

This master thesis project was kicked-off in February 2020 and - much like a tiltrotor aircraft - went through major transition phases to land successfully. Both, the process and outcome are captured in this document. It is hoped that the reader finds it insightful, and that the presented techniques provide value for future research projects.

The report is divided into four parts: Firstly, the preliminary thesis presents a tiltrotor flight dynamics model, background information on multivariate simplex B-splines and an explanation of the observed identification data deficiencies. Furthermore, an initial project overview is provided. As the research objective and questions had to be revised at a later stage, the adjusted versions can be found in an appendix. Part two, which is the scientific paper, shows how simplex spline coefficient estimators can be augmented to reliably fit structurally deficient data sets. The third part contains conclusions and recommendations, followed by appendices in the fourth part.

It has been a long, yet ultimately rewarding journey. I would like to thank everyone who was a part of it. Dr. Ir. Coen de Visser, my primary supervisor, for his guidance, patience and enthusiasm. Ir. Noor Nabi for his indispensable help in the early project phases. The members of room SIM 0.08 for their camaraderie. And, of course, my friends and family for their unconditional support during difficult times.

Lukas Steiner
Delft, February 2023

Introduction

The need for using multivariate simplex B-splines on problematic identification data arose from replacing data tables in a tiltrotor aircraft quasi-Linear Parameter Varying (qLPV) flight dynamics model. Tiltrotors are characterized by their ability to transition between helicopter and fixed-wing aircraft mode. Therefore, no single linear state-space model is capable of adequately covering the entire flight envelope. The qLPV technique relies on interpolation between stability and control derivatives, as well as trim points.

For this purpose, a set of scheduling parameters, which have the largest influence on aircraft behaviour is identified. Typical examples for a tiltrotor are air speed, nacelle incidence and wing flap angle. Point models are obtained, which cover a certain scheduling parameter range. The result is pre-processed to fit the rigid, rectangular shape of a multidimensional look-up table and interpolated during runtime. This procedure has been successfully applied to a number of aerial vehicles. However, data tables scale badly when dimension or accuracy are increased. Furthermore, local updates using new data are inherently difficult.

It was therefore recommended to replace look-up tables by multivariate simplex B-splines, which do not suffer from these drawbacks. Their high approximation power enables the fitting of scattered, globally or locally non-linear data sets characteristic of aerospace systems. Also, computations are efficient and yield transparent results. There was, however, a data deficiency whose impact only became apparent late in the project. Large portions are coplanar due to an imbalance in scheduling parameter resolution. This so-called data collinearity caused ill-conditioned regression matrices and associated numerical problems with coefficient estimates. Consequently, the project focus shifted from converting the qLPV model to answering the following research question:

> How can multivariate simplex B-splines be used effectively and reliably to fit collinear aerodynamic data sets?

Collinearity and its negative effects have been acknowledged for decades and a substantial amount of work and time dedicated to solutions. A very popular remedy, due to its power and simplicity, is the Ridge Regression Estimator (RRE). Being part of the Tikhonov regularization family, it augments the basic least-square estimator with a penalty term and a tuning parameter. On the one hand this introduces bias to the estimation, however on the other variance is reduced. No regularization scheme could be found for multivariate simplex B-splines, despite a thorough literature study. This led to the following research sub-questions:

1. How can regularization be integrated in the existing B-coefficient estimation framework?
2. How well does it work against regression matrix ill-conditioning?
3. How can good tuning parameter candidates be determined?

Regularization of other, more commonly employed spline types is well-known. Eilers and Marx in 1996 developed what they called the P-spline by adding a coefficient difference penalty to an ordinary B-spline. Although it was invented and popularized as a tool for smoothing noisy data, the P-spline can cope with collinearity just like the ridge regression estimator. The initial article presented the concept for univariate B-splines, which later was extended by the same authors to multivariate tensor product B-splines. The P-spline inherited the tensor product spline's major limitations being an inherent difficulty to fit scattered data sets and rather cumbersome calculations in higher dimensions. Simplex B-splines, by contrast, do not suffer from these. Additional research sub-questions were therefore:

4. What is the simplex counterpart to the multivariate tensor product P-spline?
5. How does it compare to ridge regression in terms of ill-conditioning reduction, goodness-of-fit and smoothing properties?

This master thesis report is intended to provide answers and act as entry point for further research.

Contents

Part I

Preliminary Thesis Report

List of Symbols

A state matrix. 6, 10
B_{κ}^{d} multivariate B-spline basis function. 14
B control matrix. 6, 10
C output matrix. 6
D feed-through matrix. 6
H smoothness matrix. 15
V airspeed. 7
h altitude. 7
c_{κ} B-coefficient vector (lexicographically sorted). 14, 15
V_D design diving speed. 5
d polynomial degree. 14
u input vector. 6
k ridge parameter. 20
λ Lagrangian multipliers. 15
n dimension, index. 14
w non-scheduling states. 6
ξ exogenous scheduling variables. 7
x state vector. 6, 8
z scheduling states. 6
ϵ model residue. 16
δ_f wing flap angle. 7
β_i nacelle incidence angle. 7
ρ scheduling parameter vector. 6

List of Acronyms and Abbreviations

ECGLS equality constrained generalized least squares. 15, 16
EOM equations of motion. 8

GLS generalized least squares. 15
GTRS Generic Tilt Rotor Simulation. 5, 6

LPV Linear Parameter Varying. 6
LT lookup table. 4, 8, 23, 24

MASST Modern Aeroservoelastic State Space Tools. 8
MSE mean squared error. 20

NITROS Network for Innovative Training on Rotorcraft Safety. 4

OLS ordinary least squares. 15, 20

qLPV quasi-Linear Parameter Varying. ii, 4–7, 10, 13, 19, 22, 23

RECLS recursive least squares. 15
RMSE root mean sqare error. 16
RRE Ridge Regression Estimator. ii, 18, 20, 21, 25
RSS residual sum of squares. 20

SCAS stability and control augmentation system. 7
SDP simplex data points. 13
SMA simplex minimum angle. 13
SP spline. 8
SRLC radius of circumsphere center and location. 13
SRSC ratio shortest ridge and radius circumsphere. 13

VDP Variance-Decomposition Proportions. 19
VIF Variance Inflation Factor. 19
VTOL vertical take-off and landing. 4

1 Introduction

Since the early days of heavier-than-air aviation two major classes of flying vehicles have emerged in the form of fixed-wing and rotary-wing aircraft. Their distinctive principle of lift generation makes them suitable for different mission profiles. Helicopters are the natural choice if hovering or vertical take-off and landing (VTOL) capabilities are required. Fixed-wing aircraft, on the other hand, offer greater payload, range, service ceiling and airspeed but require runways. Engineers in the last 50 years have come up with numerous hybrid designs to combine the best of both worlds. Many of these are rather exotic, and only a handful made it past the prototype stage and entered serial production. As of now, these are exclusively military fighter jets and transports. The first group is powered by turbofan engines and has very limited hovering capabilities. The second are so-called tiltrotors. As their names implies, they feature large, tiltable rotors located at the tip of regular-sized aircraft wings. The tilt angle ranges between 90° during take-off and landing, and 0° in cruise configuration. In between lies a transition phase, where lift is generated by main wing and rotors simultaneously.

The main disadvantage of tiltrotors is their complexity. Like any other flying vehicle they need to be safe to operate throughout their *entire* flight envelope. For a hybrid the envelope is deliberately large, making design and certification a challenging, expensive and also dangerous task. Despite all efforts, tiltrotor flight testing campaigns have a long and sad tradition of fatal accidents. In order to improve rotorcraft safety in general the Network for Innovative Training on Rotorcraft Safety (NITROS) project has been initiated. It funds research in relevant areas, one of them being the development of ”innovative methods to reduce pilot workload in transformative VTOL aircraft”. In a joint effort Politecnico di Milano and the Delft University of Technology have taken on this challenge. Starting in 2018 they have been developing a quasi-Linear Parameter Varying (qLPV) flight dynamics model of a tiltrotor aircraft, which is to serve as basis for further research. A crucial element of this model is the interpolation of certain stability, control and trim parameters as a function of the current aircraft state. Currently, data points are stored in lookup tables (LTs) and interpolated by first-order Lagrange polynomials. However, for a number of reasons it might be favourable to use function approximators instead.

This report will outline the theory behind qLPV models in general and the model-stitching architecture in particular, with an emphasis on lookup table generation and evaluation. A number of different function approximators is presented and compared at high level to select the most suitable candidate. The multivariate simplex spline proved to be most promising, and is briefly explained in its own chapter. After the theory has been established, the thesis project is defined. The research objective and questions are listed and motivated. Finally, the project schedule is presented.

2

Tiltrotor Modelling

This chapter outlines the challenges in developing flight dynamic models of tiltrotor aircraft and gives an overview of past research in the field. It then explains the theory behind one state-of-the-art approach, quasi-Linear Parameter Varying, and its implementation for a specific model. Finally, an area of potential improvement is identified within this model.

2.1. Challenges and developments

Tiltrotors are hybrids of helicopters and fixed-wing aircraft, therefore combining features of both vehicle classes. This leads to a number of challenges, not only from a design perspective but also for mathematical modelling efforts. First and foremost, they have the unique capability of rotating (tilting) their engine nacelles as shown in Figure 2.2. When in helicopter mode (Figure 2.1a), the nacelles are at an angle of 90 degrees and all lift is generated by the two proprotors at the wing tips. Consequently, control is only possible by actuating the swash plates as aerodynamic surfaces are ineffective at low speeds. This changes in conversion mode (Figure 2.1b) when the nacelles are tilted to increase forward velocity. The helicopter controls are gradually phased out in favour of fixed-wing controls until the tiltrotor is fully converted to airplane mode (Figure 2.1c).

(a) Helicopter Mode **(b)** Conversion Mode **(c)** Airplane Mode

Figure 2.1: Tiltrotor operation modes (image credit: NASA)

It is not possible to have arbitrary combinations of nacelle angle and airspeed, as can be seen in Figure 2.2. A lower limit exists due to attitude limitations and wing stall, an upper one due to mechanical limitations of the tilting mechanism, rotor shaft torque limits and eventually the design dive speed V_D. This so-called conversion corridor plays an important role in modelling tasks as every valid configuration needs to be accounted for. It is also not unique but depends on aircraft gross weight, flap/flaperon setting, rotor tip speed and atmospheric conditions.

The large variety of configurations and the resulting large flight envelope makes creating high-fidelity, continuous flight dynamic models a challenging task. The first systematic efforts were undertaken in the 1970s and 1980s as part of NASA's XV-15 tiltrotor research program. Ferguson in 1988 describes the mathematical model of a Generic Tilt Rotor Simulation (GTRS) for real-time flight simulation purposes,

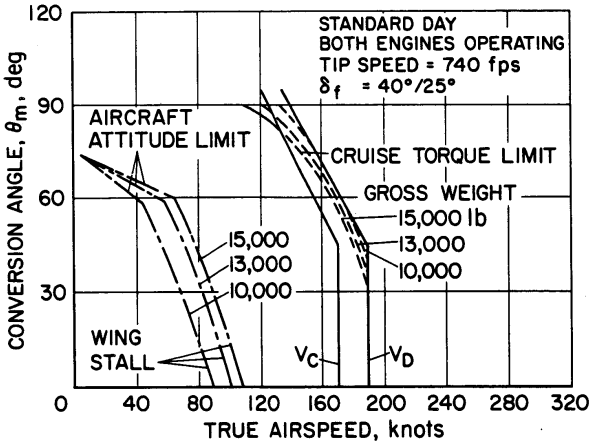


Figure 2.2: XV-15 conversion corridor, as depicted in the familiarization document [18]

based on a proprietary XV-15 model [10]. While the GTRS is relatively low-fidelity, it still serves as a benchmark in current research. In the following years, multiple other models have been developed not just for flight simulation but also aerolastic load predictions. However, they are either low-fidelity as well or limited to a number of discrete flight conditions [26].

A way to improve upon the latter issue is employing the Linear Parameter Varying (LPV), or alternatively the closely related quasi-Linear Parameter Varying (qLPV) technique. Converting non-linear systems to (q)LPV produces *linear* state-space systems whose entries depend on a number of scheduling parameters. This not only allows them to cover an aircraft's full flight envelope but also enables better control methods than classic gain scheduling [19].

In the past, a number (q)LPV frameworks were developed for rotorcraft, fixed-wing aircraft and also tiltrotors. One of the earliest rigorous investigations on how to apply LPV in this domain is the work by Marcos and Balas [19], who compared three different model-building approaches to a high-fidelity non-linear model of a commercial airliner. Lawrence, Malpica, and Theodore [16] successfully developed a large civil tiltrotor simulation, albeit somewhat limited in number of states and scheduling parameters. Other implementations were presented by Tischler and Tobias [27], who constructed *stitched* flight dynamics models of a Cessna CJ1 business jet and a UH60 utility helicopter and demonstrated their quality and usefulness.

At the Delft University of Technology, a high-order qLPV flight dynamics model of an XV-15 tiltrotor aircraft is currently under development by Nabi and Quaranta [25][26]. It improves upon existing models by increasing the number of scheduling parameters, including a rotor speed governor and increasing the number of states for higher fidelity. As its improvement is the primary goal of this master thesis project, it will be closely examined in Section 2.3 after Linear Parameter Varying (LPV)/qLPV theory is introduced in the next section.

2.2. (q)LPV modelling theory

LPV and qLPV systems are special forms of linear state-space systems, which depend on a set of time-varying scheduling parameters. Generally, a LPV system is of the form shown in Equation 2.1 [19]:

$$\begin{bmatrix} \dot{x} \\ y \end{bmatrix} = \begin{bmatrix} A(\rho(t)) & B(\rho(t)) \\ C(\rho(t)) & D(\rho(t)) \end{bmatrix} \begin{bmatrix} x(t) \\ u(t) \end{bmatrix} \tag{2.1}$$

In this equation A, B, C and D are the usual system, input, output and feed-through matrices. x is the state, u the input and ρ the vector of scheduling parameters. In an LPV system the scheduling parameters are strictly exogenous, i.e. not part of the system state vector. qLPV, on the other hand, does not have this requirement and is therefore and extension of LPV. The complete state vector in a qLPV system can therefore be broken up into scheduling (z) and non-scheduling states (w):

$$x(t) = \begin{bmatrix} z(t) & w(t) \end{bmatrix}^T \tag{2.2}$$

Substituting Equation 2.2 into Equation 2.1 yields

$$\begin{bmatrix} \dot{z}(t) \\ \dot{w}(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} A_{11}(\rho(t)) & A_{12}(\rho(t)) & B_1(\rho(t)) \\ A_{21}(\rho(t)) & A_{22}(\rho(t)) & B_2(\rho(t)) \\ C_1(\rho(t)) & C_2(\rho(t)) & D(\rho(t)) \end{bmatrix} \begin{bmatrix} z(t) \\ w(t) \\ u(t) \end{bmatrix} \tag{2.3}$$

where the vector of scheduling parameter now consists of state and exogenous variables ξ:

$$\rho(t) = \begin{bmatrix} z(t) & \xi(t) \end{bmatrix}^T \tag{2.4}$$

The most common approach to obtain (q)LPV models from non-linear models is the Jacobian linearization, due to its simplicity and wide applicability [17]. Applying first-order Taylor series at equilibrium (trim) points, Equation 2.1 becomes:

$$\begin{bmatrix} \dot{z}(t) \\ \dot{w}(t) \\ \Delta y(t) \end{bmatrix} = \begin{bmatrix} A_{11}(\rho(t)) & A_{12}(\rho(t)) & B_1(\rho(t)) \\ A_{21}(\rho(t)) & A_{22}(\rho(t)) & B_2(\rho(t)) \\ C_1(\rho(t)) & C_2(\rho(t)) & D(\rho(t)) \end{bmatrix} \begin{bmatrix} \Delta z(t) \\ \Delta w(t) \\ \Delta u(t) \end{bmatrix} \tag{2.5}$$

where $\Delta z(t) = z(t) - z_{trim}(\rho(t))$, $\Delta w(t) = u(t) - w_{trim}(\rho(t))$, $\Delta u(t) = u(t) - u_{trim}(\rho(t))$ and $\Delta y(t) = y(t) - y_{trim}(\rho(t))$.

An extension to these qLPV models, the "model stitching architecture" has been developed by Tischler and Tobias [27]. This technique combines (stitches) a set of individual linear models and trim data obtained at discrete equilibrium points, with the goal of creating a continuous, full-envelope flight dynamics model. Stitching is done along the vector of scheduling parameters, and interpolation between stability derivatives and trim points uses lookup tables. This interpolation scheme has a number of disadvantages, which will be pointed out later, but the overall result proved to be satisfactory for the business jet and helicopter study cases. Nabi and Quaranta [25] [26] use the stitching techniques for their current XV-15 tiltrotor aircraft model explained hereafter.

2.3. XV-15 Model Structure

The qLPV model of the XV-15 has been created in MATLAB Simulink using the standard elements of a qLPV model shown in Figure 2.3, augmented by additional systems such as rotor speed governor and stability and control augmentation system (SCAS). For an in-depth description the reader is referred to [26], however, it is important to understand the overall structure before focusing on the areas relevant for this research project.

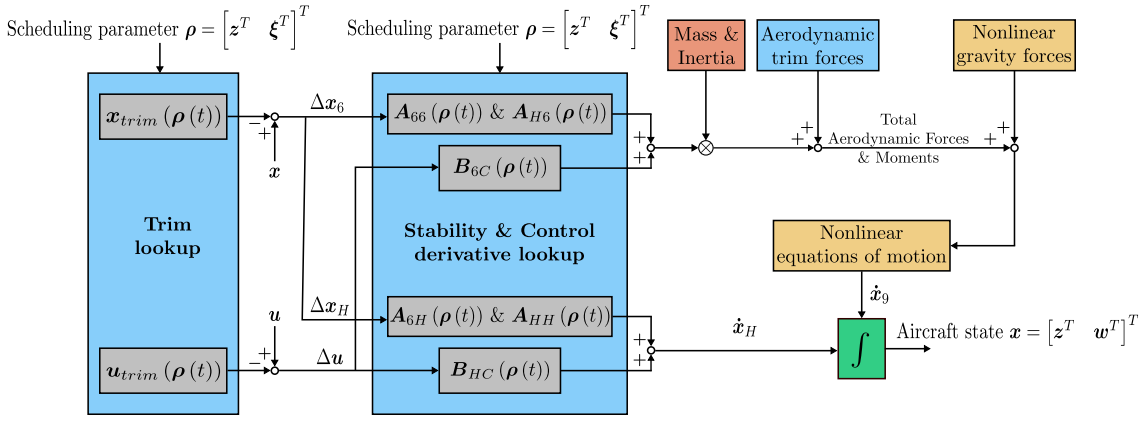


Figure 2.3: Generic qLPV model block diagram [26]

As any other qLPV model, the XV-15s is built around the state space matrices which change with the scheduling parameters. These have to be carefully chosen to ensure preservation of the original system's most important characteristics, while keeping computational complexity low. In this particular case altitude h, nacelle incidence angle β_i, wing flap angle δ_f and airspeed V have been identified.

It is, however, possible to run the model with a subset of the full vector, e.g. only nacelle angle and velocity.

Another choice is the number of states which the model uses. Table 2.1 shows the options, which are for the existing model collected under ”LT”. They differ in number of states available for wing and rotor modelling, and whether or not engine dynamics are included. Them number of rigid body states and control inputs, with the exception of engine throttle, is equal.

	LT			SP	
Degrees of freedom	91	71	41	94	85
Rigid body states	9	9	9	12	9
Wing bending mode	2	0	0	2	2
Blade bending modes in multi-blade coordinates (one collective & two cyclic) for each rotor	36	24	12	36	36
Blade torsional modes in multi-blade coordinates (one collective & two cyclic) for each rotor	24	24	12	24	24
Gimbal states in multi-blade coordinates (two cyclic) for each rotor	8	8	8	8	8
Inflow states for each rotor based on classical Pitt-Peters model	6	6	0	12	6
Engine dynamics	6	0	0	0	0
Control inputs					
Collective pitch θ_0 for each rotor	2	2	2	2	2
Lateral and longitudinal cyclic (θ_{1c},θ_{1s}) for each rotor	4	4	4	4	4
Aerodynamic control surfaces	4	4	4	4	4
Engine throttle	1	0	0	0	0

Table 2.1: States of XV-15 state-space point models

Since the model-stitching technique is used, the stability and control derivates are currently interpolated using lookup tables. The state vector x, or more precisely its perturbation Δx, is broken up into six rigid body states $x_6 = \left[\begin{array}{cccccc} u & v & w & p & q & r \end{array} \right]$ and higher-order states x_H. The three Euler angles ϕ,θ and ψ are not included in x_6 as their contribution is added in the non-linear part of the model. Perturbations are calculated on the left-hand side of the diagram by taking the difference between actual and trim state. Values of the latter are determined analogously to the stability and control derivatives, i.e. by using lookup tables and the vector of scheduling parameters.

The rigid-body and higher-order state accelerations can then be calculated as follows:

$$\left[\begin{array}{c} \dot{x}_6 \\ \dot{x}_H \end{array} \right] = \left[\begin{array}{cc} A_{66}\left(\rho\left(t\right)\right) & A_{6H}\left(\rho\left(t\right)\right) \\ A_{H6}\left(\rho\left(t\right)\right) & A_{HH}\left(\rho\left(t\right)\right) \end{array} \right] \left[\begin{array}{c} \Delta x_6 \\ \Delta x_H \end{array} \right] + \left[\begin{array}{c} B_{6C}\left(\rho\left(t\right)\right) \\ B_{HC}\left(\rho\left(t\right)\right) \end{array} \right] \Delta u \tag{2.6}$$

While higher-order states are directly integrated, the rigid body states are multiplied by the mass and inertia matrix to get the perturbed aerodynamic forces and moments. These are then added to the aerodynamic trim forces, which are functions of the trim Euler angles, and the non-linear gravity forces, which depend on the instantaneous aircraft Euler angles. The result is inserted into the non-linear equations of motion (EOM) and also integrated to obtain the current state.

2.4. Distribution of underlying state-space models

The linear state-space models, which contain the stability and control derivates, are generated by Modern Aeroservoelastic State Space Tools (MASST). This software tool, developed at Politecnico di Milano, is capable of trimming non-linear flight dynamic models at selected operating points throughout the flight envelope [22],[3]. In this particular case, the nonlinear tiltrotor model is trimmed in symmetric flight, i.e. $v = p = r = \phi = \psi = 0$. The operating points, which are referred to as anchor points in the

stitching framework, are determined by the choice of scheduling parameters. It is here where the conversion corridor (Figure 2.2) comes into play, as it limits the possible combinations of nacelle incidence angle and airspeed. Figure 2.4 shows the distribution of anchor points and how they are masked by the upper and lower corridor boundary. Models on the rectangular grid but outside the valid area are indicated for the sake of completeness, but do not actually exist.

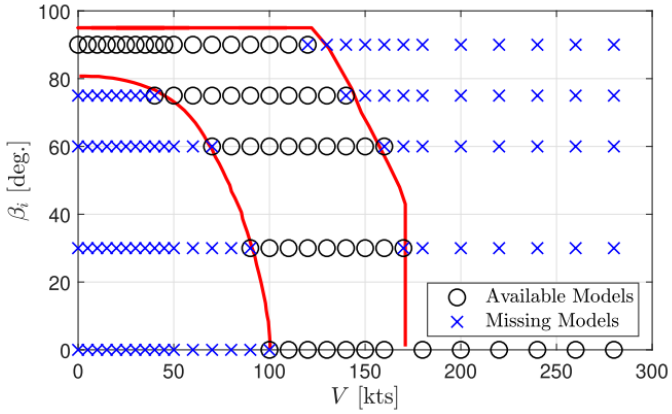


Figure 2.4: XV-15 discrete linear state-space models and conversion corridor [26]

On the y-axis five discrete nacelle incidence angles have been selected, $\beta_i = \begin{bmatrix} 0 & 30 & 60 & 75 & 90 \end{bmatrix}$ [deg]. The x-axis is divided into three resolution zones as shown in (see Table 2.2) to accurately capture the tiltrotor's behaviour at low speeds, i.e. when in helicopter mode.

V-range [kts]	V-spacing [kts]
0-50	5
60-170	10
180-280	20

Table 2.2: Velocity resolution zones

All models located inside the conversion corridor are available at flap settings $\delta_f = \begin{bmatrix} 0 & 20 & 40 & 75 \end{bmatrix}$ [deg] and altitudes $h = \begin{bmatrix} 0 & 10000 \end{bmatrix}$ [ft], so a four-dimensional scheduling vector can be used. If a lower dimension is desired, e.g. only nacelle incidence angle and velocity are scheduling parameters, the remaining ones need to be fixed or scheduled. Table 2.3 displays flap setting as function of β_i and V, where the latter is only relevant when in helicopter mode, and altitude is limited to sea level. Consequently, all other models are disregarded.

β_i [deg]	V[kts]	δ_f [deg]
90-80	0-35	75
90-80	40-280	40
75-65	all	40
60-35	all	20
30-5	all	20
0	all	0

Table 2.3: Flap scheduling

2.5. Current interpolation scheme

In the existing model determining current trim values, control and stability derivatives is done by linearly interpolating entries of lookup tables. These data structures are the de-facto standard in qLPV modelling and have been successfully used by numerous authors such as Tischler and Tobias [27], Lawrence, Malpica, and Theodore [16], Marcos and Balas [19] and Nabi and Quaranta [25]. The working principle of lookup tables, as their name suggests, is that values are computed in advance and stored in a structured way. Retrieving, and if necessary, interpolating are in most cases computationally considerably less expensive operations than evaluating the original functions. This is especially true for trimming aircraft models, which usually involves numerical minimization of a cost function [5].

In the given model, the trim states and control settings are stored in multidimensional arrays. The dimension depends on the number of scheduling parameters, reaching from 2-D for $n_\rho = 2$ to 4-D for $n_\rho = 4$. At every time step in the simulation an extraction operation is performed to retrieve the nearest adjacent data points (x_i, y_i) and (x_{i+1}, y_{i+1}). These are then used in first-order Lagrangian interpolation, the general form of which is shown in Equation 2.7 and Equation 2.8.

$$L(x) = \sum_{j=0}^{k} y_j l_j(x) \tag{2.7}$$

where:

$$l_j(x) = \prod_{\substack{0 \le m \le 1 \\ m \neq j}} \frac{x - x_m}{x_j - x_m} \tag{2.8}$$

As the dimension increases beyond one, the aforementioned two steps have to be performed along all axes, resulting in multilinear interpolation. Equation 2.9 shows the bilinear case for four points $P_{11} = (x_1, y_1)$, $P_{12} = (x_1, y_2)$, $P_{21} = (x_2, y_1)$ and $P_{22} = (x_2, y_2)$ on a rectangular grid. So in fact, the final result is a quadratic function.

$$\begin{aligned} f(x,y) &\approx \frac{y_2 - y}{y_2 - y_1} f(x, y_1) + \frac{y - y_1}{y_2 - y_1} f(x, y_2) \\ &= \frac{y_2 - y}{y_2 - y_1} \left(\frac{x_2 - x}{x_2 - x_1} f(P_{11}) + \frac{x - x_1}{x_2 - x_1} f(P_{21}) \right) + \frac{y - y_1}{y_2 - y_1} \left(\frac{x_2 - x}{x_2 - x_1} f(P_{12}) + \frac{x - x_1}{x_2 - x_1} f(P_{22}) \right) \end{aligned} \tag{2.9}$$

Using binary search for as index search method, Nabi et al. found the maximum amount of primitive operations necessary for the extraction step to be $\sum_{i=1}^{n_\rho} \log_2 k_i$ (k_i being the number of elements along each scheduling parameter). Similarly, the interpolation steps takes $3\left(2^{n_\rho} - 1\right)$ primitive operations [26]. Both steps have to be performed for every parameter, which for the state matrix A yields:

$$N_2(n_\rho, n_x) = {n_x}^2 \left(\sum_{i=1}^{n_\rho} \log_2 k_i + 3\left(2^{n_\rho} - 1\right) \right) \tag{2.10}$$

In the equations for control matrix B and trim vector the factor ${n_x}^2$ is replaced by $n_x n_u$ and $n_x + n_u$, respectively. Since the amount of system states exceeds the number of control inputs, Equation 2.10 represents the computationally most complex interpolation block. As a matter of fact, Nabi et al. show that it is the entire model's most complex part, which is hence $\mathcal{O}\left(N_t {n_x}^2 2^{n_\rho}\right)$. This is the benchmark which every alternative approach has to be pitted against.

It is important to note that for this concept to work all data has to be present on a rectangular grid, albeit uniform spacing is not required. Figure 2.4 has shown, however, that models are only available inside the non-rectangular conversion corridor. To circumvent this problem, Nabi et al. use a simple spline function to generate a finer, rectangular grid. The resolution of grid points can be chosen freely as they stem from an analytical function, however, in practise it is limited by real-time performance and memory demands. To illustrate the grid, trimmed right rotor gimbal pitch is displayed in Figure 2.5 using spacings of $\Delta V = 5$ [kts] and $\Delta\beta_i = 5$ [deg.]. This particular grid leads to a number of $[h \times \beta_i \times \delta_f \times V] = [2 \times 19 \times 4 \times 57] = 8664$ individual state-space models to be interpolated.

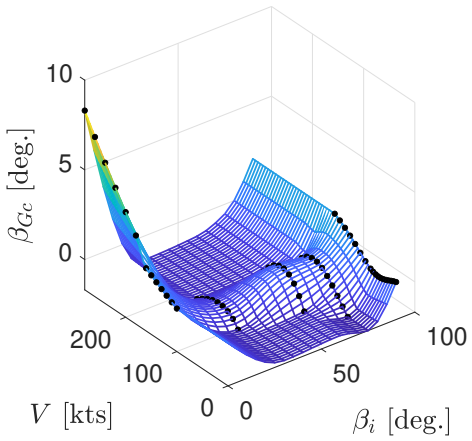


Figure 2.5: gimbal pitch (right rotor) trim surface sea level, flap setting $\delta_f = 20$ [deg.] [26]

In this figure the original data points are indicated by black dots, which correspond to the available models in Figure 2.4. As the velocity range is limited for each nacelle incidence angle, the edge models are kept to achieve a rectangular grid. In the figure, this manifests itself as relatively smooth curvature in the centre part, flanked by extruded two-dimensional profiles outside the velocity range.

While this approach has been successfully applied by Nabi et al., they identified a number of deficiencies and possible areas of improvement. The following list has been aggregated based on remarks in [26], [25] and from direct consultation with the authors:

1. Lookup tables are a data structure, which makes them inherently difficult to adjust and update if new data becomes available
2. The necessity to keep the edge models to use spline interpolation over a rectangular grid can have a negative impact on the inside area where valid data is available
3. Better interpolation accuracy can only be achieved by increasing the resolution of trim points, which can lead to excessive memory requirements
4. Computational complexity increases exponentially with the number of scheduling parameters. Broadening the dynamic flight envelope can therefore severely affect the model's ability to run in real-time
5. Using linear interpolation leads to a lack of continuity between stability and control derivatives. This is an issue when advanced, model-based controller are to be used

2.6. Investigation of alternatives

To overcome the aforementioned shortcomings, it has been suggested by Nabi and Quaranta to use multivariate simplex B-splines polynomials in place of lookup tables. Applying analytical functions instead of data structures is indeed promising, however, a number of alternative methods exist which should not be omitted right away.

Replacing the lookup-tables is a typical multidimensional curve-fitting task. As such, it is necessary to first select a suitable function approximator, define its structure and then estimate all necessary parameters. de Visser, Chu, and Mulder [7] named four commonly used methods for creating accurate global models of non-linear systems: kernel methods, polynomial models, neural networks and multivariate simplex B-splines. While all of them are capable of fitting scattered data sets, they differ in approximation power, transparency, flexibility and computational complexity. The advantages and disadvantages are presented in Table 2.4, motivating the choice to choose multivariate simplex B-splines over the alternatives. Not only do they offer high approximation power, which makes them superior to models based on single polynomials, they are also computationally efficient. This is in contrast to kernel methods and neural networks. The latter also suffer from inherent intransparency, which means that performance can not be guaranteed everywhere. The only obvious downside of using this type of spline is the need for an underlying triangulation. Finding one which is optimal, or at least close, is not a trivial task. Furthermore, the geometric characteristics of simplices can lead to data-distribution

problems in higher dimensions [29]. However, for the time being this is not considered an issue as the amount of scheduling parameters limits the dimension to four.

Advantages	Disadvantages
Kernel Methods	
• Suitable for scattered data sets	• Non-parametric, hence computationally expensive
Polynomial Models	
• Simple, straightforward construction • Widely used	• Single polynomial has limited approximation power • Combining local models causes discontinuities which might require manual blending
Neural Networks	
• Powerful, general function approximator • Suitable for fitting any scattered data set	• Inherent intransparency, no performance guarantees • Global basis functions lead to non-sparse solution systems, which are inefficient to solve • Training poses a non-linear optimization problem
Multivariate Simplex B-Splines	
• High approximation power on global model scale • Computationally efficient due to parametric nature, linear-in-the-parameter property and local polynomial basis • Allows for local refinements and updates	• Triangulation required: no universal approach available to choose optimum, star-like shape in higher dimension can lead to data-distribution problems [29]

Table 2.4: Modelling method comparison, based on [7] unless specified otherwise

3

Multivariate Splines

This chapter presents the working principle of multivariate splines, explains why they are suitable for improving tiltrotor (qLPV) models and discusses their implementation as well as quality assessment methods.

3.1. Simplices, Barycentric Coordinates

As can be inferred from the name, basic polynomials of a simplex spline are defined on simplices. These geometric structures provide a minimal span of n-dimensional space and consist of $n+1$ non-degenerate vertices [6]:

$$V := \{v_0, v_1, \ldots, v_n\} \in R^n \tag{3.1}$$

The complex hull of V, which is the n-simplex t is then defined as:

$$t := \langle V \rangle \tag{3.2}$$

While lower-dimensional simplices such as the 1-simplex (line), the 2-simplex (triangle) and 3-simplex (tetrahedron) are easy to visualize, this becomes increasingly difficult in higher dimensions. Multiple non-overlapping simplices which are used to partition a domain are called a triangulation, regardless of dimension. Generally speaking, finding an optimal triangulation for a given data set and simplex spline is not straightforward. It is therefore imperative to asses the simplex quality using the metrics shown in Table 3.1. According to de Visser, Chu, and Mulder [7], minimum simplex data points (SDP) count is the most important one as violation leads to unsolvable regression problems.

Simplex metric	'Good'	'Bad'
radius of circumsphere center and location (SRLC)	SRLC $\in t, R_c \leq K$	SRLC $\notin t, R_c > K$
ratio shortest ridge and radius circumsphere (SRSC)	SRSC < 2	SRSC > 2
simplex minimum angle (SMA)	SMA $\geq 14°$	SMA $< 14°$
simplex data points (SDP)	SDP $\geq \hat{d}$	SDP $< \hat{d}$

Table 3.1: Guidelines for the metrics of well defined ('good') and sliver ('bad') simplices, from [7]

A compelling feature of simplices is that they have their own local coordinate system with respect to their barycenter. The location of each point $x = (x_1, x_2, \ldots, x_n)$ in Cartesian coordinates can be expressed as unique weighted vector sum of the simplex vertices, giving it a barycentric coordinate $b(x) = (b_1, b_2, \ldots, b_n)$:

$$x = \sum_{i=0}^{n} b_i v_{p_i} \tag{3.3}$$

where the vertex indices are sorted in ascending order. Also, barycentric coordinates are normalised (see Equation 3.4) and can be converted back to Cartesian coordinates in a straightforward way [7].

$$\sum_{i=0}^{n} b_i = 1 \tag{3.4}$$

3.2. Simplex Polynomials

Polynomials of degree d in barycentric coordinates can be expressed using the multinomial theorem for general polynomials:

$$(b_0 + b_1 + \ldots + b_n)^d = \sum_{|\kappa|=d} \frac{d!}{\kappa_0!\kappa_1! \ldots \kappa_n!} b_0^{\kappa_0} b_1^{\kappa_1} \ldots b_n^{\kappa_n} \tag{3.5}$$

For simplification purposes the multi-index κ is introduced:

$$\kappa = (\kappa_0, \kappa_1, \ldots, \kappa_n) \tag{3.6}$$

It has the following properties:

$$\begin{aligned} \kappa! &= \kappa_0!\kappa_1! \ldots \kappa_n! \\ |\kappa| &= \kappa_0 + \kappa_1 + \ldots + \kappa_n \end{aligned} \tag{3.7}$$

Equation 3.5 therefore becomes:

$$(b_0 + b_1 + \ldots + b_n)^d = \sum_{|\kappa|=d} \frac{d!}{\kappa!} b^\kappa = \sum_{|\kappa|=d} B_\kappa^d(b) = 1 \tag{3.8}$$

where $\frac{d!}{\kappa!}$ is known as the multinomial coefficient and B_κ^d defined as the basis function of the multivariate spline. It has been proven by de Boor in 1987 [4] that this basis is stable; all basis polynomials add up to unity at every point on the simplex. Consequently, any polynomial $p(b)$ of degree d can be written as linear combination of basis functions in the so-called B-form (Equation 3.9):

$$p(b) = \sum_{|\kappa|=d} c_\kappa B_\kappa^d(b) \tag{3.9}$$

The polynomial shape is determined by the vector of B-coefficients c_κ, the total number of which depends on polynomial degree d and dimension n:

$$\hat{d} = \begin{pmatrix} d+n \\ n \end{pmatrix} = \frac{(d+n)!}{n!d!} \tag{3.10}$$

3.3. B-net, Continuity and Parameter Estimation

A noteworthy property of the B-coefficients is their spatial distribution on the simplex, forming a net structure. The relationship between B-coefficient multi-index and location is:

$$b(c_k) = \frac{\kappa}{d} \tag{3.11}$$

Figure 3.1 shows the B-coefficient net, or B-net in short, for a $d = 3$ basis function on a triangulation of three simplices. The orientation with respect to its parent simplex is determined by the B-net rule, which states that B-coefficients with high multi-indices must be located at vertices of low index.

This sorting scheme is very important for inter-simplex continuity of splines, which relies on the following condition:

$$c_{\kappa_0,m,\kappa_1}^{t_2} = \sum_{|\gamma|=m} c_{(\kappa_0,0,\kappa_1)+\gamma}^{t_1} B_\gamma^m(v_*), 0 \leq m \leq r \tag{3.12}$$

Graphically, the relation of B-coefficients leads to a structure spanning over the simplex boundaries. In Figure 3.1 first-order (C1) continuity is indicated by bold lines. Generally, the higher the continuity order, the more coefficients are involved.

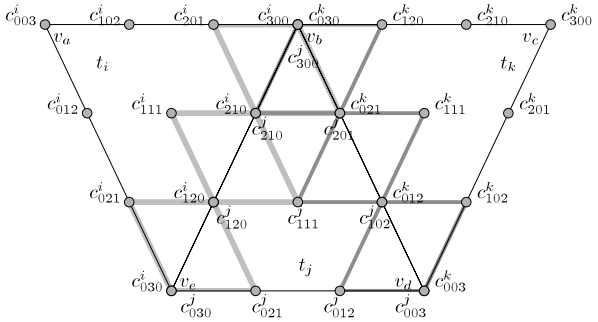


Figure 3.1: Example B-net structure [6]

From a mathematical point of view, the set of continuity conditions can be collected in matrix form:

$$Hc_\kappa = 0 \tag{3.13}$$

where H is the so-called smoothness matrix and c_κ the global vector of lexicographically sorted B-coefficients.

Multivariate simplex splines are frequently used in system identification and curve-fitting tasks. This requires the B-coefficients to be chosen in a way that achieves a good fit with a given data set. The most popular approach, due to its simplicity and availability of efficient solvers, is linear regression. The corresponding model is of the form shown in Equation 3.14. X is the regression matrix and consists of blocks corresponding to individual simplices. Y is the vector of measurements and r the residuals. If smoothness conditions have to be satisfied (Equation 3.13), the model is said to be equality constrained.

$$Y = Xc + r \tag{3.14}$$

Multiple methods exist to find a c_κ which minimises the residual, or more precisely, a related metric such as the residual square. They mainly differ by how strict assumptions on noise have to be, and how complex the required solving operations are. According to de Visser et al. [8], generalized least squares (GLS) estimators and constrained recursive least squares (RECLS) estimators are able to cover a wide range of typical applications. The following paragraphs will only treat GLS, as the second method is designed for use in real-time adaptive modelling applications and hence out of scope.

The equality constrained generalized least squares (ECGLS) optimization problem is shown below.

$$\hat{c} = \arg\min \left[\frac{1}{2} (Y - Xc)^\top \Sigma^{-1} (Y - Xc) \right] \text{ subject to } Hc = 0 \tag{3.15}$$

The assumptions made on noise are:

1. The residual noise is white and has variable magnitude; $E(r) = 0$, $\operatorname{cov}(r) = \Sigma$. The residual covariance matrix is usually estimated by using an (equality constrained) ordinary least squares (OLS) estimator beforehand.
2. There is no state noise present

Minimizing Equation 3.15 can again be done by a variety of methods. The one used by de Visser et al. [8] is Lagrangian multipliers. Reformulating the cost function as Lagrangian expression yields Equation 3.16, where λ is the vector of multipliers. It is also possible to introduce constraints beyond the smoothness matrix.

$$\mathcal{L}(c, \lambda) = \frac{1}{2} (Y - Xc)^\top \Sigma^{-1} (Y - Xc) + \lambda^\top Hc \tag{3.16}$$

Setting the partial derivatives to zero and writing the results in matrix form yields the Karush-Kuhn-Tucker (KKT) system:

$$\begin{bmatrix} X^\top \Sigma^{-1} X & H^\top \\ H & 0 \end{bmatrix} \begin{bmatrix} c \\ \lambda \end{bmatrix} = \begin{bmatrix} X^\top \Sigma^{-1} Y \\ 0 \end{bmatrix} \tag{3.17}$$

The ECGLS / Lagrangian estimator is consequently:

$$\begin{bmatrix} \hat{c} \\ \hat{\lambda} \end{bmatrix} = \begin{bmatrix} X^\top \Sigma^{-1} X & H^\top \\ H & 0 \end{bmatrix}^{-1} \begin{bmatrix} X^\top \Sigma^{-1} Y \\ 0 \end{bmatrix} = \begin{bmatrix} C_1 & C_2 \\ C_3 & C_4 \end{bmatrix}^{-1} \begin{bmatrix} X^\top \Sigma^{-1} Y \\ 0 \end{bmatrix} \tag{3.18}$$

with statistics:

$$\mathsf{cov}\left(\hat{c}\right) = C_1 \tag{3.19}$$

$$\mathsf{var}\left(\hat{c}\right) = \mathsf{diag}\left(C_1\right) \tag{3.20}$$

It is possible to use the Moore-Penrose pseudo inverse to solve Equation 3.18, however, this is a computationally very expensive method. A more efficient way is using an iterative solver. Defining Q as the dispersion matrix:

$$Q := X^\top \Sigma^{-1} X \tag{3.21}$$

The first iteration is:

$$\hat{c}^{(1)} = \left(2Q + \frac{1}{\varepsilon} H^\top H\right)^{-1} \left(2X^\top \Sigma^{-1} Y - H^\top \hat{\lambda}^{(0)}\right) \tag{3.22}$$

Subsequent iterations are:

$$\hat{c}^{(k+1)} = \left(2Q + \frac{1}{\varepsilon} H^\top H\right)^{-1} 2Q\hat{c}^{(k)} \tag{3.23}$$

where ε is a small number such as 10^{-6} and $\lambda^{(0)}$ an initial estimate for the Lagrange multipliers.

3.4. Spline Quality Assessment

A number of metrics exist to assess the quality of any multivariate simplex B-spline. They can be grouped into model residue analysis, statistical model quality assessment and model stability analysis.

The first model residue metric is the *relative* root mean sqare error (RMSE). The residual ϵ is defined as follows:

$$\varepsilon = Y - X\hat{c} \tag{3.24}$$

Using the regular root mean sqare error (RMSE):

$$RMS\left(\varepsilon\right) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\varepsilon\left(i\right)\right)} \tag{3.25}$$

And scaling with the difference between minimum and maximum value to obtain a more useful metric:

$$RMS_{rel}\left(\varepsilon\right) = \frac{RMS\left(\varepsilon\right)}{\max Y_V - \min Y_V} \tag{3.26}$$

Another metric is the coefficient of determination R2 shown below, where SS_{res} is the residual sum of squares and SS_{tot} the total sum of squares. It gives an indication of how much variation in the data can be explained by the regression model. As de Visser et al. notes [8] it should be used with caution though, as overconfidence in model quality can happen in some cases.

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}} = 1 - \frac{\sum\left(\hat{y}_i - \bar{y}\right)^2}{\sum\left(y_i - \bar{y}\right)^2} \tag{3.27}$$

Next to considering residual magnitude it can also be assessed whether or not they are correlated. The per-simplex autocorrelation function estimate is given by:

$$\hat{R}_t(k) = \frac{1}{N_t} \sum_{i=1}^{N_t-k} \varepsilon_t(i)\,\varepsilon_t(i+k), k = 0, 1, ..., N_t \tag{3.28}$$

The residuals can be considered uncorrelated if Equation 3.29 holds true for at least $\alpha \times N_t$ of N_t values of the per-simplex autocorrelation function:

$$\frac{\left|\hat{R}_t(k \neq 0)\right|}{\hat{R}_t(0)} \leq \frac{\mathcal{N}_\alpha}{\sqrt{N_t}}, k > 0 \tag{3.29}$$

The metric for statistical model analysis is the covariance of B-coefficients cov $(\hat{c})$, which has been shown before in Equation 3.19. Since the coefficients have a spatial distribution, so does their covariance. The resulting surface can be used to identify problematic areas.

The metric for model stability analysis are the B-coefficient bounds. These are unique to splines. If Equation 3.30 holds, then the spline is well behaved inside its domain.

$$\|p\| \leq \|c\| \tag{3.30}$$

where:

$$\|c\| := \max_{|\kappa|=d} |c_\kappa| \tag{3.31}$$

4

Collinearity

This chapter explains how the presence of collinearity affects the solution of regression problems, introduces a diagnostic method and presents possible remedies. One method, the Ridge Regression Estimator (RRE) is discussed in more detail.

4.1. Problem Description

During the initial data set examination it became clear that it suffers from a deficiency known as data collinearity. This can happen when data points are distributed on lines, either perfectly or very closely.

Collinearity is not a problem per se, which explains why gridded data is in most cases perfectly adequate for curve fitting. The tiltrotor set's specific issue is the imbalance in number of unique data points per independent variable. Figure 4.1 shows a scatter plot of the 2D data set, as well as two side views. It can be observed that, while there are a great number of velocity levels available, only a few nacelle incidence angle settings exist. The ratio of *effective* data points is hence 57/5, which becomes problematic when higher-order polynomials are to be used for linear least-square regression.

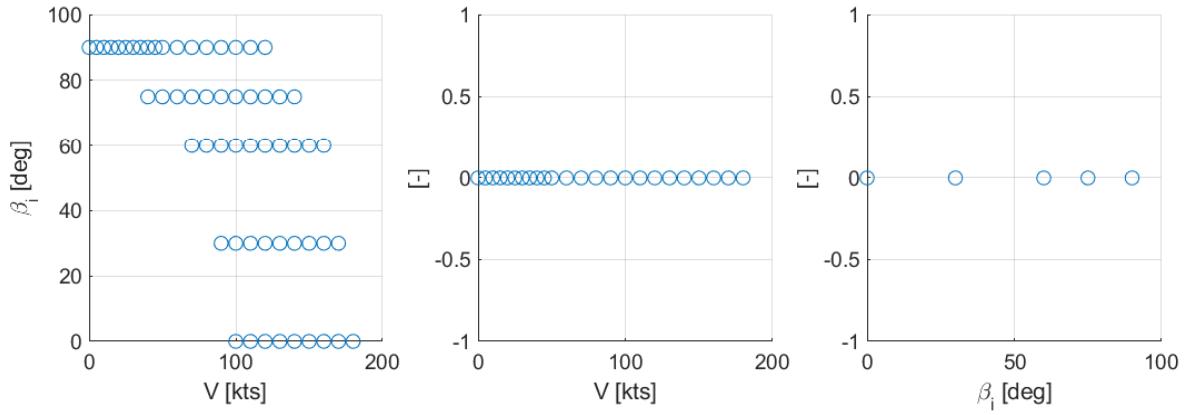


Figure 4.1: 2D Data Point Distribution

First of all, it needs to be noted that the presence of collinearity does not lead to a bad fit in terms of residuals but negatively affects the model parameter estimates. Assuming the collinearity is not perfect, which causes the cross-product matrix $\boldsymbol{X}'\boldsymbol{X}$ to be non-invertible, the main ill effects of collinearity are [15] [1]:

1. Very low estimation precision, which leads to large, potentially correlated errors. Also, parameter variance is excessively high.
2. The relative importance of parameters can be masked by their inflated size, potentially leading to erroneous elimination.

3. High sensitivity of coefficient estimates due to small changes in the data, reducing confidence in the model's prediction quality.

While we are not so much interested in the interpretation of coefficients, precise and reliable estimates are essential if the qLPV model is to be used for purposes such as controller design. Collinearity therefore needs to be diagnosed and, if not avoided altogether, contained.

4.2. Diagnostic

While collinearity is easily identified as root cause for the ill-conditioned data matrix in the given, low dimensional case, this might not always be possible. Specifically, a column in the data matrix might be the linear combination of multiple others [1]. To pinpoint the issue one can use Variance-Decomposition Proportions (VDP), which assigns variance of regression parameters to singular values. If almost all variance of a parameter can be attributed to a large singular value then (near-) collinearity is present.

An alternative, frequently employed metric are the Variance Inflation Factors (VIFs) [2]:

$$VIF_i = \frac{1}{1 - R_i^2} \tag{4.1}$$

where R_i^2 is the multiple correlation coefficient of one data matrix column regressed on the remaining ones. A large VIF points to collinearity, but the method has issues distinguishing coexisting near-dependencies and suffers from numerical instability when collinearity is present [1].

4.3. Possible Remedies

If one is determined to use multivariate simplex B-splines for fitting collinear data, a number of possible remedies exist. Naturally, each has its own advantages and drawbacks, which are briefly outlined here:

1. Introduce additional, well-conditioned data
   The preferred approach. It might, however, be difficult or expensive to get new data, or the data is inconsistent.
2. Add random increment to existing data ("joggling")
   Attractive because no change to the estimator is required, however, when collinearity is present even small data manipulation can have a large (unpredictable) effect [1].
3. Ridge Regression a.k.a. Tikhonov regularization
   Popular approach to solving ill-posed regression problems in statistics, trades bias for improved parameter variance. This appears to be the most promising approach, as the concept has been researched and applied for decades to much more complex problems [14]. Furthermore, it is relatively straightforward to implement in the existing creation framework.
4. Pure Bayes estimator
   Very flexible and powerful, but requires prior knowledge and a non-linear solver. Implementation is expected to be time-consuming.
5. Lower spline polynomial degree and/or lower number of simplices
   Easiest solution, but goodness-of-fit might be unacceptably poor.
6. Variable elimination
   More targeted than reducing spline polynomial degree, however, extreme variance inflation causes issues with typical selection criteria [21] [1]. Also, continuity and other constraints likely complicate the process.
7. Increase inter-simplex continuity
   Reducing degrees of freedom requires less uncorrelated data points, but also reduces the goodness-of-fit.

4.4. Ridge Regression

Ridge regression, also known as Tikhonov regularization, can be employed to alleviate the problems caused by collinearity. The underlying idea is shrinking the regression coefficient vector by imposing a penalty on its size, which introduces a small amount of bias but greatly reduces variance [13]. It has

been shown that the RRE can have a lower mean squared error (MSE) compared to the OLS estimator [14], but this is not the primary motivation for using it here.

The well-known OLS estimator, which is the best linear *unbiased* estimator, is:

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{y} \tag{4.2}$$

where $\boldsymbol{X}$ is the data or design matrix, and $\boldsymbol{Y}$ the vector of measurements. Comparing, the *biased* RRE as described in [14] is given by:

$$\hat{\boldsymbol{\beta}}^{\star} = [\boldsymbol{X}'\boldsymbol{X} + k\boldsymbol{I}]^{-1}\boldsymbol{X}'\boldsymbol{y} = \boldsymbol{W}\boldsymbol{X}'\boldsymbol{y} \tag{4.3}$$

where $k \geq 0$ is also known as the ridge parameter. This clearly shows that the RRE works even if $\boldsymbol{X}'\boldsymbol{X}$ itself is rank-deficient due to correlation of the predictor variables. Conceptually, Equation 4.3 is the solution to a constrained least squares problem:

$$RSS\,(k) = (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})'\,(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}) + k\boldsymbol{\beta}'\boldsymbol{\beta} \tag{4.4}$$

This shows that k is simply the Lagrange multiplier of the constraint. However, its optimum value to achieve the minimum residual sum of squares (RSS) is generally unknown and finding it is no trivial task [23]. A simple and therefore popular approach to finding, if not the optimum, at least good candidates is using the so-called ridge trace. This is a plot of estimated regression coefficients as functions of k. It is normally sufficient to consider a range of $0 < k \leq 1$ for the coefficients to stabilize [23].

The corresponding estimator variance can be calculated by [14]:

$$VAR\left[\hat{\boldsymbol{\beta}}^{\star}\right] = \sigma^2\boldsymbol{Z}\,(\boldsymbol{X}'\boldsymbol{X})\,\boldsymbol{Z}' \tag{4.5}$$

where

$$\boldsymbol{Z} = \boldsymbol{I} - k\boldsymbol{W} \tag{4.6}$$

and σ^2 the usual residual variance. This illustrates that increasing k indeed decreases estimator variance.

The aforementioned equations are valid for unconstrained parameter estimation, however, multivariate simplex B-splines can have both inter-simplex continuity requirements and additional constraints. When linear restrictions of the form $\boldsymbol{R}\boldsymbol{\beta} = \boldsymbol{r}$ are to be included, Equation 4.3 becomes [12]:

$$\hat{\boldsymbol{\beta}}_r\,(k) = \hat{\boldsymbol{\beta}}\,(k, \boldsymbol{\beta_0}) - \boldsymbol{S}_k^{-1}\boldsymbol{R}'\left[\boldsymbol{R}\boldsymbol{S}_k^{-1}\boldsymbol{R}'\right]^{-1}\left(\boldsymbol{R}\hat{\boldsymbol{\beta}}\,(k, \boldsymbol{\beta}_0) - \boldsymbol{r}\right), k \geq 0 \tag{4.7}$$

where

$$\boldsymbol{S}_k = \boldsymbol{X}'\boldsymbol{X} + k\boldsymbol{I}_p \tag{4.8}$$

and

$$\hat{\boldsymbol{\beta}}\,(k, \boldsymbol{\beta}_0) = (\boldsymbol{X}'\boldsymbol{X} + k\boldsymbol{I}_p)^{-1}\left(\boldsymbol{X}'\boldsymbol{y} + k\boldsymbol{R}'\,(\boldsymbol{R}\boldsymbol{R}')^{-1}\boldsymbol{r}\right) \tag{4.9}$$

If no additional constraints are present, $\boldsymbol{H}\boldsymbol{c} = \boldsymbol{R}\boldsymbol{\beta} = \boldsymbol{r} = 0$ which simplifies Equation 4.9 to the unconstrained RRE in Equation 4.3. The constrained RRE's covariance matrix is given by:

$$Cov\left(\hat{\boldsymbol{\beta}}_r\,(k)\right) = \sigma^2\boldsymbol{M}_k\boldsymbol{X}'\boldsymbol{X}\boldsymbol{M}_k \tag{4.10}$$

where

$$\boldsymbol{M}_k = \boldsymbol{S}_k^{-1} - \boldsymbol{S}_k^{1}\boldsymbol{R}'\left[\boldsymbol{R}\boldsymbol{S}_k^{-1}\boldsymbol{R}'\right]^{-1}\boldsymbol{R}\boldsymbol{S}_k^{-1} \tag{4.11}$$

When reviewing the literature it appears that most authors standardize the data matrix columns by centring and scaling, so $\boldsymbol{X}'\boldsymbol{X}$ is in correlation form [23] [14] [20]. This is done for a number of reasons, such as reducing unnecessary ill-conditioning due to model structure, and generally a better comparability of coefficient estimates. In the context of ridge regression is typically done because the estimated coefficients are not equivariant under scaling [13]. This is because the RRE estimate size depends on the predictor variable's variance [28]. Standardizing hence avoids dominance of certain coefficients over others.

While these findings advocate standardizing whenever possible, the user can also decide to use the original data matrix instead [28]. There has been, to this author's knowledge, no previous attempt to use the RRE in combination with simplex B-splines. In case of difficulties it might therefore be necessary to resort to the latter approach.

5

Project Definition

This chapter establishes the research project objective, lists and motivates the associated research questions and explains how they are intended to be answered.

5.1. Research Objective

The research objective is investigating how stitched qLPV flight dynamics models can benefit from using multivariate simplex B-splines in place of linearly interpolated lookup tables by applying the procedure to an existing tiltrotor model.

5.2. Research Questions

Below are the four research questions which have been selected, each followed by a brief motivation:

1. How does varying the number of simplices and polynomial degree affect model accuracy?

   The absence of efficient algorithms makes finding optimal triangulations a very challenging task, especially in higher dimensions. It is therefore the model designer's responsibility to find one which offers sufficient performance. An investigation will be performed as to how accurate different combinations of triangulations and polynomial degrees are.

2. What is the effect of skipping data manipulation and pre-processing steps on model accuracy?

   Lookup tables require data to be spread out on strictly rectangular grids. However, data generated by simulations or flight tests is often scattered and/or limited in coverage. For example, the XV-15 can only operate inside the non-rectangular conversion corridor. Trim state and control settings, system and input matrices are hence unavailable for certain nacelle angle-airspeed pairs. This necessitates data pre-processing and manipulation. Firstly, to fill the grid outside the flight envelope with synthetic data. Secondly, to obtain gridded data from scattered input. Thirdly, to artificially increase grid density. In the stitching architecture this is typically done by relatively basic spline curve fitting methods, which do not offer much control over the process. On the contrary, multivariate simplex B-splines use flexible triangulations and are able to fit *any* scattered data set. It is therefore expected to have a fit of similar or better quality, in addition to a simplified model building process.

3. How much more memory efficient is the spline-based model for similar accuracy?

   Linearly interpolated lookup tables require increased trim point resolution for improved accuracy. This comes at the cost of computer memory, which can become unacceptable in higher dimensions. Splines, on the contrary, are continuous functions and only require the storage of B-coefficients and triangulation vertices. The question is therefore not if, but how much more efficient the spline-based model is.

4. How does memory usage compare as the number of scheduling parameters is increased?

   Increasing the number of scheduling parameters leads to a broader flight envelope and is therefore desirable. However, lookup table memory usage does not scale well with dimension due to

their fundamental structure. Additionally, it is not only usable data taking up space. Synthetic data is required to fill otherwise empty grid cells, but adds no value to the model itself. Splines have no such issues and are therefore expected to perform significantly better.

5.3. Methodology

To answer the first three research questions it is necessary to create two branches of spline-based stitched qLPV models. The first branch is to investigate accuracy (Q1 & Q2), the second memory complexity (Q3). Initially only two scheduling parameters are used, but the final goal is three. Below are the steps necessary for this endeavour. The first, preparation, is only required once while the other two need be repeated for each model.

1. Preparation

   The initial step is familiarization with the provided Matlab/Simulink model and accompanying support scripts. It needs to be understood how the model stitching architecture has been implemented and what additional blocks are present. A focus will be put on model inputs, outputs, states and signal format. Next, the lookup table and interpolation blocks will be analysed. Questions include, but are not limited to: In what format is the data stored? How are the parameters sorted? How are the breakpoints retrieved? How is the interpolation scheme implemented?

   Once the model has been understood sufficiently, attention will shift to the provided data package. Firstly, the data needs to be checked for completeness. It needs to be ensured that all relevant model parameters available as functions of the scheduling parameters. Also, the data might contain components such as Coriolis terms. These are added later in the quasi-non-linear model and hence need to be removed. Secondly, the data distribution throughout the domain is analysed. This is necessary to determine promising triangulation options and identify gaps. Thirdly, additional data is obtained to fill the gaps and either improve the model or be used in the verification process.
2. Curve-fitting

   This step can be separated into setup and parameter estimation. Setup means deciding what triangulation, spline degree and continuity are to be used. This is based on the previous data analysis and external requirements, such as a minimum degree of continuity. Next, the B-coefficients are estimated with the help of an already existing Matlab toolbox. Developed by Coen de Visser in recent years, this toolbox contains functions to facilitate the entire process of creating multivariate simplex B-spline models. It might therefore not only be used for parameter estimation, but also automated triangulation in dimensions beyond two.
3. Verification and Validation

   These two steps are crucial, as they create confidence in model quality. While often used interchangeably, verification is done to determine if a simulation model accurately represents the chosen physical model and validation determines if simulation results accurately represent the physical model [24].

   Verification will focus on the spline function residuals and model parameters, as has been outlined in Chapter 3. Due to the large number of individual models (over 8000 for the system matrix alone) a high degree of automation is desirable. When a model shows deficiencies the user is alerted and directed to its location for further manual inspection.

   Next to judging individual spline quality it is necessary to evaluate their impact on model output. In case of the system and control matrices these are the *aerodynamic* rigid body state accelerations. Trim, as shown in Figure 2.3, is used to calculate state and input perturbations but also influences the aerodynamic trim force. The strategy is to compare each contribution to the original LT model by using equal inputs, which are recorded from a number of reference cases. Additionally, the combined aerodynamic accelerations are investigated.

   Finally, validation will be carried out by running the entire model using reference case inputs. These inputs and selected state time histories are available in [26]. Since the model is open-loop, however, it is possible that the model diverges and generates unsatisfactory results. In this case the V&V effort will be limited to verification only.

Answering the research questions warrants a number of different techniques, which are listed in Table 5.1. The first two require the verification and validation techniques described above. Question three is concerned with memory complexity. For a fair comparison it is therefore necessary to not use the most accurate spline model, but the simplest one which matches LT accuracy. A comparison with the high-accuracy models is included for the sake of completeness. Question four is a fundamental one, independent of the underlying data. It is therefore be answered by a theoretical analysis rather than studying a certain model.

#	answered by
1	Verification & Validation techniques
2	Verification & Validation techniques
3	Practical memory demand analysis
4	Theoretical memory complexity analysis

Table 5.1: Research question answering methods

5.4. Project planning

Presented below are the thesis project milestones, as required by the MSc Thesis Kick-off Form (AE 2). For a detailed view of work packages and due dates see the Gantt chart.

Milestone	(Expected) Date
Start MSc project	26/02/2020
Kick-off meeting	18/03/2020
Hand in literature study	22/04/2022
Submit draft thesis	10/06/2022
Green light review	16/06/2022
Request examination	16/06/2022
Thesis hand-in	01/07/2022
Defence	14/07/2022

Table 5.2: Thesis milestones

6

Appendix

This chapter establishes the adjusted research project objective, lists and motivates the associated research questions.

6.1. Adjusted Research Objective

The research objective is to curve-fit a collinear tiltrotor stability derivative by a multivariate simplex B-spline.

6.2. Adjusted Research Questions

1. How can multivariate simplex B-splines be used effectively and reliably to fit collinear aerodynamic data sets?

   A good fit requires small residuals and low-variance coefficient estimates of reasonable magnitude for prediction/interpolation. Collinearity can prevent simultaneous achievement of these goals.

   (a) How can regularization be integrated in the existing B-coefficient estimation framework?
   This, apparently, has not be done before.
   (b) How well does it work against regression matrix ill-conditioning?
   This is what the RRE has been originally designed to do.
   (c) How can good tuning parameter candidates be determined?
   It would be very convenient if simple, established techniques could be applied.
   (d) What is the simplex counterpart to the multivariate tensor product P-spline?
   While sharing the concept of scaled basis functions, simplex B-splines have a local coordinate system and continuity conditions.
   (e) How does it compare to ridge regression in terms of ill-conditioning reduction, goodness-of-fit and smoothing properties?
   It is useful to know when to use each type.

References

[1] David A. Belsley, Edwin Kuh, and Roy E. Welsch. *Regression diagnostics*. Wiley, 1980. ISBN: 0471058564.

[2] Samprit Chatterjee and Ali Hadi. *Regression Analysis by Example*. English. Hoboken, 2012. URL: http://site.ebrary.com/id/11034369%20http://www.myilibrary.com?id=769933%20http://rbdigital.oneclickdigital.com%20http://pmt-eu.hosted.exlibrisgroup.com/openurl/44OPN_INST/44OPN_services_page?u.ignore_date_coverage=true%5C&rft.mms_id=9952509021802316%20https://archi.

[3] F. Colombo et al. "A comprehensive aeroservoelastic approach to detect and prevent rotorcraft-pilot coupling phenomena in tiltrotors". In: *American Helicopter Society's 74th Annual Forum and Technology Display*. 2018.

[4] Carl de Boor. "B-form basics". In: *Geometric modeling: algorithms and new trends* (1987), pp. 21–28.

[5] Agostino De Marco, Eugene L. Duke, and Jon S. Berndt. "A general solution to the aircraft trim problem". In: *Collection of Technical Papers - 2007 AIAA Modeling and Simulation Technologies Conference*. Vol. 2. American Institute of Aeronautics and Astronautics Inc., 2007, pp. 792–831. ISBN: 1563479060. DOI: 10.2514/6.2007-6703.

[6] C. C. de Visser, Q. P. Chu, and J. A. Mulder. "A new approach to linear regression with multivariate splines". In: *Automatica* 45.12 (Dec. 2009), pp. 2903–2909. ISSN: 00051098. DOI: 10.1016/j.automatica.2009.09.017.

[7] C. C. de Visser, Q. P. Chu, and J. A. Mulder. "Differential constraints for bounded recursive identification with multivariate splines". In: *Automatica* 47.9 (Sept. 2011), pp. 2059–2066. ISSN: 00051098. DOI: 10.1016/j.automatica.2011.06.011.

[8] C. C. de Visser et al. *Global Nonlinear Model Identification*. 2011. ISBN: 9789085707707.

[9] Paul H.C. Eilers and Brian D. Marx. "Flexible smoothing with B-splines and penalties". In: *Statistical Science* 11.2 (1996), pp. 89–102. ISSN: 08834237. DOI: 10.1214/ss/1038425655.

[10] Samuel W. Ferguson. *A Mathematical Model for Real Time Flight Simulation of a Generic Tilt-Rotor Aircraft*. Tech. rep. NASA, 1988.

[11] Richard Franke. "A critical comparison of some methods for interpolation of scattered data. Naval Postgraduate school". In: *Technology Report NPS-53-79-003* (1979).

[12] Jürgen Groß. "Restricted ridge estimation". In: *Statistics and Probability Letters* 65.1 (2003), pp. 57–64. ISSN: 01677152. DOI: 10.1016/j.spl.2003.07.005.

[13] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The elements of statistical learning*. English. New York, 2009. URL: http://site.ebrary.com/id/10289757%20https://search.ebscohost.com/login.aspx?direct=true%5C&scope=site%5C&db=nlebk%5C&db=nlabk%5C&AN=277008%20http://www.myilibrary.com?id=212674%20https://doi.org/10.1007/978-0-387-84858-7%20https://doi.org/10.1007/b94608%20http://web.stanford..

[14] Arthur E. Hoerl and Robert W. Kennard. "Ridge Regression : Biased Estimation for Nonorthogonal Problems". In: *Technometrics* 12.1 (1970), pp. 55–67.

[15] John Johnston. *Econometric methods*. English. 2nd ed. New York: McGraw-Hill, 1972. ISBN: 0070326797.

[16] Ben Lawrence, Carlos A Malpica, and Colin R Theodore. "The development of a large civil tiltrotor simulation for hover and low-speed handling qualities investigations". In: *36th European Rotorcraft Forum, ERF 2010*. Vol. 1. 2010, pp. 361–372. ISBN: 9781510854765. URL: www.aeronautics.nasa.gov/fap.

[17] Marco Lovera, Marco Bergamasco, and Francesco Casella. “LPV modelling and identification: An overview”. In: *Lecture Notes in Control and Information Sciences*. Vol. 437 LNCIS. Springer, Berlin, Heidelberg, 2013, pp. 3–24. ISBN: 9783642361098. DOI: 10.1007/978-3-642-36110-4_1.

[18] M. D. Maisel, D. C. Borgman, and D. D. Few. *Tilt Rotor Research Aircraft Familiarization Document*. Tech. rep. January. 1975, pp. 1–105. URL: http://hdl.handle.net/2060/19750016648.

[19] Andrés Marcos and Gary J. Balas. “Development of Linear-Parameter-Varying Models for Aircraft”. In: *Journal of Guidance, Control, and Dynamics* 27.2 (May 2004), pp. 218–228. ISSN: 07315090. DOI: 10.2514/1.9165.

[20] Donald W Marquardt. “Generalized Inverses, Ridge Regression, Biased Linear Estimation, and Nonlinear Estimation”. In: *American Statistical Association, American Society for Quality, Taylor & Francis* 12.3 (2018), pp. 591–612.

[21] Donald W Marquardt and Ronald D Snee. “Ridge Regression in Practice”. In: 29.1 (1975), pp. 3–20.

[22] Pierangelo Masarati, Vincenzo Muscarello, and Giuseppe Quaranta. “Linearized aeroservoelastic analysis of rotary-wing aircraft”. In: *36th European Rotorcraft Forum, ERF 2010*. Vol. 1. 2010, pp. 493–502. ISBN: 9781510854765.

[23] Gary C. McDonald. “Ridge regression”. In: *Wiley Interdisciplinary Reviews: Computational Statistics* 1.1 (2009), pp. 93–100. ISSN: 19395108. DOI: 10.1002/wics.14.

[24] Erwin Moij. *Lecture Notes AE3205*.

[25] Hafiz Noor Nabi and Giuseppe Quaranta. “A Quasi-Linear Parameter Varying (QLPV) modeling approach for real time piloted simulation of tiltrotor”. In: *45th European Rotorcraft Forum 2019, ERF 2019*. Vol. 2. 2019, pp. 989–1000. ISBN: 9781713805922.

[26] Hafiz Noor Nabi et al. “Development of a quasi-linear parameter varying model for a tiltrotor aircraft”. In: *CEAS Aeronautical Journal 2021* 1 (Sept. 2021), pp. 1–16. ISSN: 1869-5590. DOI: 10.1007/S13272-021-00539-1. URL: https://link.springer.com/article/10.1007/s13272-021-00539-1.

[27] Mark B. Tischler and Eric L. Tobias. *A Model Stitching Architecture for Continuous Full Flight-Envelope Simulation of Fixed-Wing Aircraft and Rotorcraft from Discrete Point Linear Models*. Tech. rep. April. 2016. URL: http://www.dtic.mil/docs/citations/AD1008448.

[28] Wessel N. van Wieringen. *Lecture notes on ridge regression*. 2015. arXiv: 1509.09169. URL: http://arxiv.org/abs/1509.09169.

[29] T. Visser, C. C. de Visser, and E. Van Kampen. “System Identification using the Multivariate Simplotope B-Spline”. In: *Papers 12th Pegasus-AIAA student conference*. Valencia, Spain, 2016.

Part II

Scientific Paper

Curve Fitting of Collinear Data using Multivariate Simplex P-Splines

L. F. Steiner
Ir. N. Nabi
Dr. Ir. C. C. de Visser
Control and Simulation Department
Delft University of Technology
Delft, The Netherlands
l.f.steiner@student.tudelft.nl

***Abstract*—Data collinearity is a serious problem when applying curve fitting techniques for aerodynamic model identification. As a remedy, Tikhonov regularization is routinely applied to simple polynomials, univariate B-splines and tensor product B-splines. However, if the data set contains significant local and global nonlinearities, the higher approximation power of multivariate simplex B-splines may be required. No regularization scheme is known to exist yet for this type of spline function, which limits their application to non- or only mildly collinear data sets. It is therefore proposed to integrate two variants of Tikhonov regularization: ridge regression and a spline-specific coefficient difference penalty. Both are able to overcome data collinearity issues, which is being demonstrated on a real system identification problem. Furthermore, they show promise for local data smoothing applications.**

***Index Terms*—multivariate splines, system identification, collinearity, Tikhonov regularization, P-spline**

I. Introduction

Modern aircraft are complex systems featuring large flight envelopes. It is not always possible or cost-effective to build aerodynamic models from flight tests alone. Instead, data is obtained from wind tunnel experiments or generated by computer programs. This allows for selected experimental parameters to be fixed while others are varied in predefined increments. The result is often a rigid grid-like data structure.

A popular method for fitting linear models is least-square regression, which relies on matrix inversion. If fitting data is collinear, i.e. large portions are coplanar, the regression matrix becomes ill-conditioned or singular. The estimator's ability to fit the identification data is generally not affected, but its prediction power becomes severely degraded. This is because low estimation precision due to ill-conditioning leads to large potentially correlated estimator error and inflated variance. Various methods such as VIF (variance inflation factors) or VDP (variance-decomposition proportions) can be applied to pinpoint the cause and assess the damage. For more insight in this particular field, the reader is referred to the book by Belsley, Kuh, and Welsch [2].

A number of solutions are known to overcome or, at the very least, to alleviate the issue for general linear models. New, non-coplanar and consistent data can be introduced. The model structure can be adapted to yield a well-conditioned regression matrix. The generalized inverse is able to find solutions of ill-conditioned problems. Small random increments can be added to coplanar data.

If none of these straightforward approaches are viable, Tikhonov regularization is often used. This method augments the basic least-square estimator with tunable penalties to enforce a favourable solution. While it had originally been applied to simple polynomial models, it was later extended to univariate and multivariate (tensor product) B-splines [15, 9, 17].

These are useful and proven function approximators, but aerodynamic data sets frequently contain significant local and global nonlinearities. Multivariate simplex B-splines possess the necessary approximation power [7, 20] but lack a dedicated method to prevent negative effects caused by data collinearity. Differential constraints, as presented in [6], can potentially be used as a fix. However, the need for expert knowledge and manual tuning limits this particular application to small, low-complexity models.

The main contribution of this paper is, therefore, a Tikhonov regularization scheme for multivariate simplex B-splines, with the goal of removing numerical issues caused by strongly collinear data sets. Being inspired by the work of Eilers and Marx [9], the result is named the multivariate simplex P-spline.

Further contributions are the adaptation of coefficient difference penalties to multivariate simplex splines and the evaluation of two established tuning parameter selection criteria. The novel simplex P-splines are demonstrated on a highly collinear stability derivative of the tiltrotor aircraft flight dynamics model presented in [18].

II. Preliminaries on Multivariate Simplex B-Splines

This section aims to offer a broad overview of multivariate simplex B-splines. It is based on the work of de Visser et al., which presents the use of simplex splines in a system identification context and beyond [5, 7, 6].

Multivariate simplex B-splines are piecewise polynomials of predefined continuity, which exist on geometric structures known as simplices. Per definition, a simplex provides a minimal span of n-dimensional space using $n+1$ non-degenerate

vertices. These can be used to define the local barycentric coordinate system. Multiple simplices can be combined to form a triangulation, such as the simple one shown in Figure 1.

While this offers great flexibility, there is no general approach to choosing an optimal triangulation for a given data set. A popular choice are subdivided (hyper-) rectangles, due to their simplicity. It is always possible to create a rectangular triangulation and remove simplices without data content afterwards, however, flexible alternatives such as the Delaunay triangulation exist.

The well-known multinomial theorem for a polynomial of degree d in barycentric coordinates b is:

$$\begin{aligned}(b_0 + b_1 + \ldots + b_n)^d &= \sum_{|\kappa|=d} \frac{d!}{\kappa_0!\kappa_1! \ldots \kappa_n!} b_0^{\kappa_0} b_1^{\kappa_1} \ldots b_n^{\kappa_n} \\ &= \sum_{|\kappa|=d} \frac{d!}{\kappa!} b^{\kappa} = \sum_{|\kappa|=d} B_{\kappa}^{d}(b) = 1\end{aligned} \tag{1}$$

$\kappa = (\kappa_0, \kappa_1, \ldots, \kappa_n)$ is a multi-index introduced for convenience. It has the following properties:

$$\begin{aligned}\kappa! &= \kappa_0!\kappa_1! \ldots \kappa_n! \\ |\kappa| &= \kappa_0 + \kappa_1 + \ldots + \kappa_n\end{aligned} \tag{2}$$

The total number of valid permutations is:

$$\hat{d} = \frac{(d+n)!}{n!d!} \tag{3}$$

All individual basis functions $B_{\kappa}^{d}(b)$ add up to 1 at every point in the simplex, which means the basis is stable. This is a considerable advantage over common global polynomials. Expanding on this insight, de Boor showed in [4] that *any* polynomial p can be expressed in the so-called B-form by multiplying each basis functions with a constant c_{κ} :

$$p(b) = \sum_{|\kappa|=d} c_{\kappa} B_{\kappa}^{d}(b) \tag{4}$$

It is possible to express (4) in vector notation, which is very useful in practical applications [5]. $\boldsymbol{c}$ is the vector of B-coefficients, and t_j is an index which refers to individual simplices j in a triangulation t:

$$p(b) = \boldsymbol{B}_{t_j}^{d}(b) \cdot \boldsymbol{c}^{t_j} \tag{5}$$

A spline function's directional derivative can be calculated using its original B-coefficients [6].

$$D_u^m p(b) = \frac{d!}{(d-m)!} \boldsymbol{B}^{d-m}(b) \boldsymbol{A}^{d,d-m}(a) \cdot \boldsymbol{c}^{t} \tag{6}$$

where m is the derivative order, $\boldsymbol{A}$ the de Casteljau matrix, and a the coordinate of derivative direction u.

Another important basis polynomial feature is the unique spatial distribution of B-coefficients, which forms the so-called B-net. An example of polynomial degree $d = 4$ is shown in Figure 1. Note that coefficients are overlapping on the shared

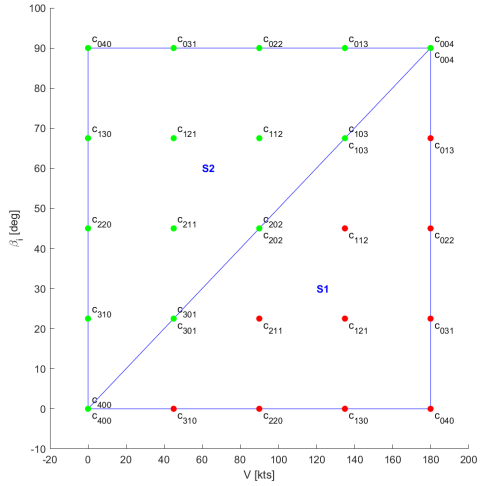


Fig. 1: B-Coefficient Spatial Distribution (B-Net): 2 Simplices, $d = 4$, $n = 2$

simplex edge. Their barycentric coordinates can be calculated from its multi-index by:

$$b(c_k) = \frac{\kappa}{d} \tag{7}$$

Inter-simplex continuity is enforced by relating coefficients of neighbouring simplices in the following way:

$$c_{\kappa_0,m,\kappa_1}^{t_j} = \sum_{|\gamma|=m} c_{(\kappa_0,0,\kappa_1)+\gamma}^{t_k} B_{\gamma}^{m}(v_*), 0 \leq m \leq r \tag{8}$$

where γ is a multi-index independent of κ, r is the continuity degree and v_* the out-of-edge vertex of t_j. This requires the coefficients to be sorted following the B-net rule: high multi-index coefficients at vertices of low index and low multi-index coefficients at vertices of high index.

The set of all continuity conditions is collected in the smoothness matrix $\boldsymbol{H}$. Solving the ordinary (optionally, weighted or general) least-square problem to determine the optimal coefficient vector estimate $\hat{\boldsymbol{c}}$ leads to constrained minimization:

$$\hat{\boldsymbol{c}} = \arg\min_{\boldsymbol{c}} \|\boldsymbol{Y} - \boldsymbol{B}\boldsymbol{c}\|_2^2 \tag{9}$$

subject to:

$$\boldsymbol{H}\boldsymbol{c} = 0 \tag{10}$$

$\boldsymbol{Y}$ is the observation vector of length N and $\boldsymbol{B}$ the regression matrix.

$$\boldsymbol{B} = \begin{bmatrix} \boldsymbol{B}_{t_1} & 0 & 0 & 0 \\ 0 & \boldsymbol{B}_{t_2} & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & \boldsymbol{B}_{t_J} \end{bmatrix} \in R^{N \times J \cdot \hat{d}} \tag{11}$$

where each block consists of the B-form polynomial evaluated at each of the n points which lay inside the respective simplex:

$$\boldsymbol{B}_{t_j} = \begin{bmatrix} \boldsymbol{B}_{t_j}^d(\boldsymbol{b}_1) \\ \boldsymbol{B}_{t_j}^d(\boldsymbol{b}_2) \\ \vdots \\ \boldsymbol{B}_{t_j}^d(\boldsymbol{b}_n) \end{bmatrix} \tag{12}$$

A variety of methods exist to solve the constrained optimization problem; however, a popular choice is Lagrange multipliers for which efficient iterative solvers exist:

$$\mathcal{L}(\boldsymbol{c}, \boldsymbol{\mu}) = \frac{1}{2}(\boldsymbol{Y} - \boldsymbol{B}\boldsymbol{c})'(\boldsymbol{Y} - \boldsymbol{B}\boldsymbol{c}) + \boldsymbol{\mu}'\boldsymbol{H}\boldsymbol{c} \tag{13}$$

where $\boldsymbol{\mu}$ is the constraint multiplier. Taking partial derivatives, this leads to the following equations:

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \boldsymbol{c}} &= -(\boldsymbol{Y} - \boldsymbol{B}\boldsymbol{c})'\boldsymbol{B} + \boldsymbol{\mu}'\boldsymbol{H} \\ &= \boldsymbol{B}'\boldsymbol{B}\boldsymbol{c} + \boldsymbol{H}'\boldsymbol{\mu} - \boldsymbol{B}'\boldsymbol{Y} = 0 \end{aligned} \tag{14}$$

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\mu}} = \boldsymbol{H}\boldsymbol{c} = 0 \tag{15}$$

Setting up the Karush-Kuhn-Tucker (KKT) matrix.

$$\begin{bmatrix} \boldsymbol{B}'\boldsymbol{B} & \boldsymbol{H}' \\ \boldsymbol{H} & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{c} \\ \boldsymbol{\mu} \end{bmatrix} = \begin{bmatrix} \boldsymbol{B}'\boldsymbol{Y} \\ 0 \end{bmatrix} \tag{16}$$

$$\begin{bmatrix} \hat{\boldsymbol{c}} \\ \hat{\boldsymbol{\mu}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{B}'\boldsymbol{B} & \boldsymbol{H}' \\ \boldsymbol{H} & 0 \end{bmatrix}^+ \begin{bmatrix} \boldsymbol{B}'\boldsymbol{Y} \\ 0 \end{bmatrix} = \begin{bmatrix} \boldsymbol{C}_1 & \boldsymbol{C}_2 \\ \boldsymbol{C}_3 & \boldsymbol{C}_4 \end{bmatrix} \begin{bmatrix} \boldsymbol{B}'\boldsymbol{Y} \\ 0 \end{bmatrix} \tag{17}$$

B-coefficient covariance is given by [7]:

$$Cov(\hat{\boldsymbol{c}}) = \boldsymbol{C}_1 \tag{18}$$

The Lagrangian estimator requires both smoothness matrix $\boldsymbol{H}$ and dispersion matrix $\boldsymbol{Q} = \boldsymbol{B}'\boldsymbol{B}$ to be of full rank [5]. This requirement translates to a minimum number of $\hat{d}$ non-coplanar data points to be available. Otherwise, $\boldsymbol{B}$ and, consequently, $\boldsymbol{Q}$ become ill-conditioned or singular.

III. Penalized Splines

Tikhonov regularization is frequently used to prevent ill-conditioning without changing model structure or data. In its simplest form it is the well-known ordinary least-square estimator augmented with a penalty term $\lambda\boldsymbol{P}$ [15]:

$$\hat{\boldsymbol{\beta}}_t = (\boldsymbol{X}'\boldsymbol{X} + \lambda\boldsymbol{P})^{-1}\boldsymbol{X}'\boldsymbol{Y}, \lambda \geq 0 \tag{19}$$

$\hat{\boldsymbol{\beta}}_t$ is the parameter vector estimate, $\boldsymbol{X}$ is the regression matrix, and λ is the tuning parameter and $\boldsymbol{P}$ is the penalty matrix. It is defined as:

$$\boldsymbol{P} = \boldsymbol{\Gamma}'\boldsymbol{\Gamma} \tag{20}$$

where $\boldsymbol{\Gamma}$ is the Tikhonov matrix. It is important to note that (19) does not contain additional constraints, which are required for spline inter-simplex continuity. The penalized estimator is therefore derived using the Lagrange multiplier technique presented earlier. In [21], it is explained how a penalty can be considered an additional constraint to the least-square minimization problem (9). For generic penalties, this constraint is:

$$\|\boldsymbol{\Gamma}\boldsymbol{c}\|_2^2 \leq t \tag{21}$$

for some suitable $t > 0$. Expanding the first term:

$$\|\boldsymbol{\Gamma}\boldsymbol{c}\|_2^2 = \boldsymbol{c}'\boldsymbol{\Gamma}'\boldsymbol{\Gamma}\boldsymbol{c} = \boldsymbol{c}'\boldsymbol{P}\boldsymbol{c} \tag{22}$$

Making use of (22), the least-square minimization problem including both constraints can be written as Lagrangian:

$$\begin{aligned} \mathcal{L}(\boldsymbol{c}, \boldsymbol{\mu}, \boldsymbol{\lambda}) = &\frac{1}{2}(\boldsymbol{Y} - \boldsymbol{B}\boldsymbol{c})'(\boldsymbol{Y} - \boldsymbol{B}\boldsymbol{c}) + \\ &\boldsymbol{\mu}'\boldsymbol{H}\boldsymbol{c} + \frac{1}{2}\boldsymbol{\lambda}(\boldsymbol{c}'\boldsymbol{P}\boldsymbol{c} - t) \end{aligned} \tag{23}$$

where $\boldsymbol{\mu}$ and $\boldsymbol{\lambda}$ are the constraint multipliers. Taking partial derivatives:

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \boldsymbol{c}} &= -(\boldsymbol{Y} - \boldsymbol{B}\boldsymbol{c})'\boldsymbol{B} + \boldsymbol{\mu}'\boldsymbol{H} + \boldsymbol{\lambda}\boldsymbol{c}'\boldsymbol{P} \\ &= [\boldsymbol{B}'\boldsymbol{B} + \boldsymbol{\lambda}\boldsymbol{P}]\boldsymbol{c} + \boldsymbol{H}'\boldsymbol{\mu} - \boldsymbol{B}'\boldsymbol{Y} = 0 \end{aligned} \tag{24}$$

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\mu}} = \boldsymbol{H}\boldsymbol{c} = 0 \tag{25}$$

It is not necessary to compute the partial derivative with respect to $\boldsymbol{\lambda}$ as it is our tuning parameter and remains in (24). Setting up the KKT matrix leads to the following augmented equations.

$$\begin{bmatrix} \boldsymbol{B}'\boldsymbol{B} + \boldsymbol{\lambda}\boldsymbol{P} & \boldsymbol{H}' \\ \boldsymbol{H} & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{c} \\ \boldsymbol{\mu} \end{bmatrix} = \begin{bmatrix} \boldsymbol{B}'\boldsymbol{Y} \\ 0 \end{bmatrix} \tag{26}$$

$$\begin{aligned} \begin{bmatrix} \hat{\boldsymbol{c}} \\ \hat{\boldsymbol{\mu}} \end{bmatrix} &= \begin{bmatrix} \boldsymbol{B}'\boldsymbol{B} + \boldsymbol{\lambda}\boldsymbol{P} & \boldsymbol{H}' \\ \boldsymbol{H} & 0 \end{bmatrix}^+ \begin{bmatrix} \boldsymbol{B}'\boldsymbol{Y} \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} \boldsymbol{C}_1 & \boldsymbol{C}_2 \\ \boldsymbol{C}_3 & \boldsymbol{C}_4 \end{bmatrix} \begin{bmatrix} \boldsymbol{B}'\boldsymbol{Y} \\ 0 \end{bmatrix} \end{aligned} \tag{27}$$

The covariance matrix of the spline coefficients, adapted from [16] for the ordinary least-square (OLS) case and utilizing previously established notation, is given by:

$$\boldsymbol{C}_a = \sigma^2(\boldsymbol{B}'\boldsymbol{B} + \lambda\boldsymbol{P})^{-1} \tag{28}$$

Retracing the steps of de Visser, Chu, and Mulder in [5], who use insights about the generalized inverse from Rao [19], yields the familiar result:

$$Cov(\hat{\boldsymbol{c}}) = \boldsymbol{C}_1 \tag{29}$$

Ridge Penalty (RP)

The ridge penalty, named after its characteristic structure, is by far the most popular choice for $\boldsymbol{P}$. The reason is its effectiveness and extreme simplicity:

$$\boldsymbol{P} = \boldsymbol{I} \tag{30}$$

where $\boldsymbol{I}$ is the identity matrix of appropriate size. This allows for very easy and flexible integration into various

models. While expressions for the special case of constrained least-squares are readily available [13], the so-called ridge regression has, at least to this author's knowledge, not yet been applied to multivariate simplex B-splines.

Implementation in the inverse KKT matrix (27) is straightforward, especially if λ is chosen to be a constant. One is, however, free to choose a value per simplex:

$$\boldsymbol{\lambda P} = diag(\lambda_1, \lambda_2, ..., \lambda_n)\boldsymbol{I} \tag{31}$$

This allows for precise local application, continuity conditions notwithstanding. The effect is B-coefficient shrinkage, as its Euclidean norm is penalized. Variance will decrease as penalty influence on the fit grows in relation to the residual influence. Both coefficient vector magnitude and variance approach 0 as $\lambda \to \infty$. Ill-conditioning is reduced because small $\boldsymbol{B'B} + \boldsymbol{\lambda P}$ eigenvalues increase [15]. The effect is very visible in the regression matrix condition number, which for normal matrices is the ratio of the largest and the smallest eigenvalue.

In statistical literature, the columns of $\boldsymbol{B'B}$ are typically standardized, i.e. centred and then scaled by their variance, before the ridge penalty is applied. This procedure can be helpful as covariates might be on a very different scale. Also, the ridge regression estimator has an inherent "preference" for covariates of high variance [21]. It is, however, not essential to achieve good results and has in fact been disregarded for simplex B-spline applications. The reasoning is as follows:

1) Standardization requires coefficient rescaling after the estimation, which in combination with smoothness constraints is not straightforward
2) Using barycentric coordinates involves data centring and scaling, reducing the need for additional pre-processing

P-Splines using this penalty type are referred to as RP-splines for the remainder of this article.

Difference Penalty (DP)

The difference penalty is inspired by the work of Eilers and Marx, who applied a similar approach for their univariate and tensor product P-splines [9, 17]. These are, in the following, referred to as conventional. The fundamental idea is to use the coefficients' spatial distribution and penalize differences between them. It has been shown that curve roughness, defined by (32), is reduced for conventional P-splines.

$$R = \int_l^u \left[f''(x)\right]^2 dx \tag{32}$$

where l and u are the domain bounds, and $f''(x)$ a (directional) derivative. A secondary, for fitting of collinear data crucial effect is the reduction of regression matrix ill-conditioning. Enforcing similarity between adjacent coefficients reduces both variance and relative magnitude, which will be demonstrated in the result section.

Construction of $\boldsymbol{P}$ for the simplex DP-spline is more involved than it is for the simplex RP-spline because knowledge about relative coefficient positions is required. The basic building block is the forward difference equation (33), adapted from [1]. Backward and central differences were considered as well, however, the forward direction proved to be most beneficial in terms of geometric interpretation. The forward difference of order k calculated at B-coefficient c_i in the direction u is given below:

$$\begin{aligned}\Delta_{i,u}^k &= \Delta_{i+1,u}^{k-1} - \Delta_{i,u}^{k-1} \\ &= \sum_{j=0}^{k} (-1)^j \binom{k}{j} c_{\kappa_i + (k-j)\mu_u} \\ &= \boldsymbol{D}_u^k \cdot c_{\kappa_i + (k-j)\mu_u}\end{aligned} \tag{33}$$

where $\boldsymbol{D}_u^k$ is the difference relation vector in a single direction. The independent multi-index $\boldsymbol{\mu}$ is assembled from all unit index offset vector $\boldsymbol{\omega}$ permutations and an additional row of zeros.

$$\boldsymbol{\omega} = \begin{bmatrix} 1 & -1 & \boldsymbol{z} \end{bmatrix} \tag{34}$$

where $\boldsymbol{z}$ is a zero-vector of $n-1$ elements.

The addition of $(k-j)\,\mu_u$ to κ_i results in a unit step in the direction u. If the step leads out of the simplex, i.e. $\kappa_i + (k-j)\,\mu_u \notin \kappa$, the difference relation is dropped. In total, $(n+1)!$ permutations of $\boldsymbol{\omega}$ are possible. Two step options are available in each direction, one leading forward and the other backward. Of these, one is redundant because the procedure is applied to all coefficients $\boldsymbol{c}_\kappa$ in the B-net. Forward differences of one c_{κ_i} are backward differences of others.

The per-coefficient difference matrix consists of one difference relation per direction:

$$\boldsymbol{D}_{c_i} = \begin{bmatrix} \boldsymbol{D}_1^k & \boldsymbol{D}_2^k & \dots & \boldsymbol{D}_{n+1}^k \end{bmatrix}^\top \tag{35}$$

It is used as the building block for the per-simplex difference matrix:

$$\boldsymbol{D}_{t_j} = \begin{bmatrix} \boldsymbol{D}_{c_1} & \boldsymbol{D}_{c_2} & \dots & \boldsymbol{D}_{c_{\hat{d}}} \end{bmatrix}^\top \tag{36}$$

The global difference matrix comprising all J simplices can then be assembled:

$$\boldsymbol{D}_g = diag\left(\begin{bmatrix} \boldsymbol{D}_{t_1} & \boldsymbol{D}_{t_2} & \dots & \boldsymbol{D}_{t_J} \end{bmatrix}\right) \tag{37}$$

The penalty matrix is the squared global difference matrix:

$$\boldsymbol{P} = \boldsymbol{D}_g \boldsymbol{D}_g \tag{38}$$

Consider the following example of a two-dimensional, fourth-order simplex polynomial and first-order difference penalty. The corresponding unit index offset vector has three elements:

$$\boldsymbol{\omega} = \begin{bmatrix} 1 & -1 & 0 \end{bmatrix} \tag{39}$$

This leads to a total of $3! = 6$ permutations. Figure 2 shows how they relate a single coefficient $c_1 = c_{\kappa_1} = c_{211}$ to its 6 surrounding neighbours. For the purpose of constructing $\boldsymbol{D}_{c_1}$, only half are required. The index offset vectors corresponding to the 3 green links in Figure 2 are shown in (40). However, the red ones would have been an equally valid choice.

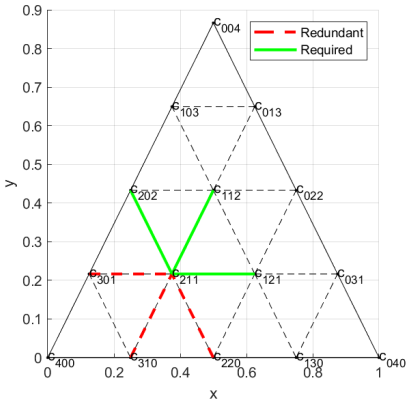


Fig. 2: Visualization of $\boldsymbol{D}_{c_1}$, $k = 1$, $d = 4$

$$\boldsymbol{\mu} = \begin{bmatrix} \mu_0 \\ \mu_1 \\ \mu_2 \\ \mu_3 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -1 & 1 \\ -1 & 0 & 1 \\ -1 & 1 & 0 \end{bmatrix} \tag{40}$$

Since $k - j = 1$, the coefficient matrix is composed as follows:

$$\boldsymbol{c}_{211+\mu_u} = \begin{bmatrix} c_{211} & c_{202} & c_{112} & c_{121} \end{bmatrix}^\top \tag{41}$$

Multiplication with the coefficient difference matrix shown below yields the forward differences used for penalization.

$$\boldsymbol{D}_{c_1} = \begin{bmatrix} \boldsymbol{D}_1^1 \\ \boldsymbol{D}_2^1 \\ \boldsymbol{D}_3^1 \end{bmatrix} = \begin{bmatrix} -1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{bmatrix} \tag{42}$$

$$\boldsymbol{\Delta}_1^1 = \boldsymbol{D}_{c_1} \cdot \boldsymbol{c}_{211+\mu_u} \tag{43}$$

The procedure is repeated for every coefficients in the B-net until all $\boldsymbol{D}_{t_j}$ and, eventually, $\boldsymbol{D}_g$ are filled.

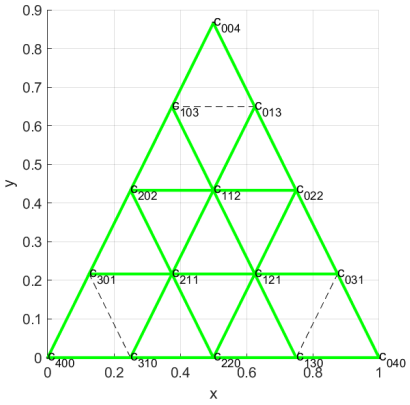


Fig. 3: Visualization of $\boldsymbol{D}_{t_j}$, $k = 2$, $d = 4$

If second order differences were to be used, a $\boldsymbol{D}_{t_j}$ matrix can be visualized as in Figure 3. Close to the simplex vertices an insufficient amount of coefficients is available to form difference relations in all directions, leading to gaps. Preliminary investigation has shown no negative effects; however, in the following sections, only first order differences penalties are used as a precaution.

IV. Selection of Tuning Parameter λ

The difficulty in constructing a penalty matrix ranges from trivial to relatively straightforward. Therefore, the actual challenge when using Tikhonov regularization is selecting an optimal tuning parameter λ. This is the one minimizing estimator error.

A multitude of methods, or information criteria, have been developed for this purpose. However, none of these appear to be universally applicable. Furthermore, multiple estimation cycles are required to compare performance. A recommendation often found in literature is examining the ridge trace, a plot of coefficients values as function of λ. Residual behaviour needs to be monitored simultaneously. Covariates in statistical problems often have a physical interpretation, so expert knowledge can be used to determine at which point they assume expected values.

A possible metric for simplex B-splines, due to their stable basis property, is the coefficient maximum value or fraction of coefficients within the data range. There are, however, no universal guidelines of what good values might be.

Instead, it has been decided to follow Eilers, Marx, and Durb in their choice of information criteria for conventional P-splines [8].

Generalized Cross-Validation

Developed by Golub, Heath, and Wahba, generalized cross-validation (GCV) is a method of choosing an optimum tuning parameter [12]. Originally intended for ridge regression, it has been successfully applied to penalized splines before [9]. For a particular value of λ the equation is:

$$GCV\left(\lambda\right) = \sum_{i=1}^{n} \left[\frac{y_i - \hat{y}_i}{n - ED} \right]^2 \tag{44}$$

where n is the number of measurements, y_i an observation and $\hat{y}_i$ its prediction. ED is the effective model dimension:

$$ED = \mathrm{tr}\left(\boldsymbol{G}\right) \tag{45}$$

$\boldsymbol{G}$ is commonly known as the "hat" matrix as it projects the vector of measurements to its prediction:

$$\boldsymbol{G} = \boldsymbol{B}\boldsymbol{C}_1\boldsymbol{B}' \tag{46}$$

$$\hat{\boldsymbol{Y}} = \boldsymbol{G}\boldsymbol{Y} = \boldsymbol{B}\hat{\boldsymbol{c}} \tag{47}$$

The fundamental idea behind leave-one-out cross-validation is that the best estimator is the one minimizing prediction error:

$$\epsilon = \sum_{i=1}^{n} \left(y_i - \hat{y}_i\right)^2 \tag{48}$$

The index i indicates that this particular data point has been left out of the estimation. It is, however, not necessary to solve n least-square problems to calculate ϵ as all relevant information is contained in $\boldsymbol{G}$.

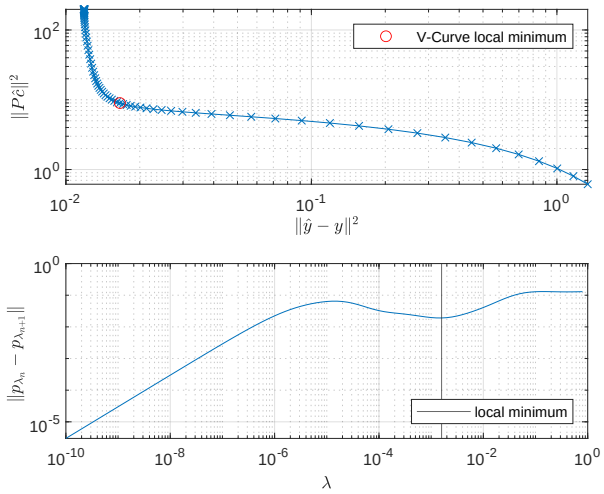


Fig. 4: Typical RP-Spline L-Curve (top) and V-Curve (bottom)

Next to being rotation-invariant, the GCV method does not require a noise covariance matrix σ^2 estimate. Since collinearity can cause serious difficulty creating even a preliminary model, this is a very useful property. Noise has to be uncorrelated for an unbiased estimate of prediction error [11, 3].

L- and V-Curve

The L-curve was developed by Hansen as an alternative way of determining a good ridge parameter [14]. It is a parametric plot, typically on a log-log scale, which contains two essential metrics for any Tikhonov regularization effort: penalty effect and residual behaviour.

The top half of Figure 4 shows the L-curve for an exemplary RP-spline estimation. On the vertical axis is the penalty effect. Its interpretation depends on the selected penalty type. For ridge regression, it is the squared coefficient vector magnitude:

$$\|\boldsymbol{P}\hat{\boldsymbol{c}}\|^2 = \|\boldsymbol{I}\hat{\boldsymbol{c}}\|^2 = \|\hat{\boldsymbol{c}}\|^2 \tag{49}$$

On the horizontal axis is the residual vector squared magnitude. Each point in the parametric plot corresponds to a value of tuning parameter λ. Here, it has been iterated over a range of 10^{-10} to 10^{0}.

The effect on both coefficient vector length and residuals is initially very small but grows larger as λ increases. What gives the L-curve its name and makes it useful is the distinct corner in the middle. It was found that the point of maximum curvature makes for a very good choice of λ. In fact, it outperforms GCV for correlated residuals [14]. However, no mathematical proof of its good performance is available.

The calculation of curvature to find the maximum is not straightforward, which is why simplified alternatives have been developed. Eilers and Marx recognized that L-curve point density around the corner is higher than elsewhere. They therefore proposed the minimum of Euclidean distance between points as an equivalent choice for λ and called the plot V-curve [10].

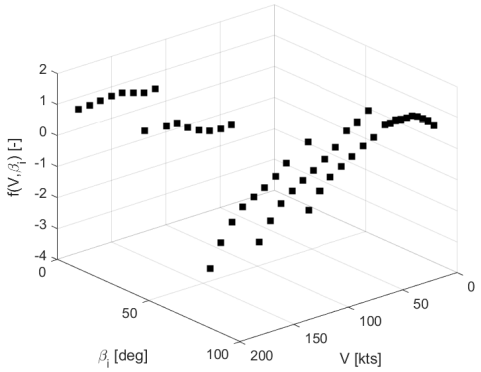


Fig. 5: Identification Data Set

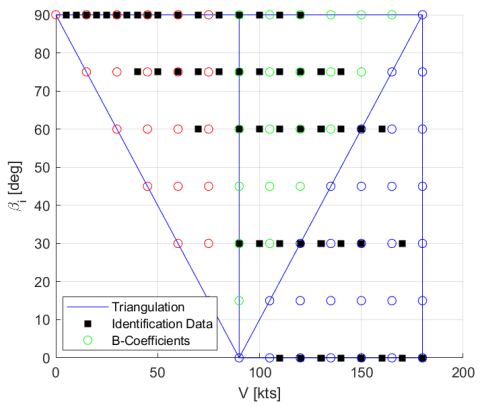


Fig. 6: Triangulation, Identification Data and B-net for $d = 6$ (coefficient colour based on simplex membership)

The bottom half of Figure 4 is the V-curve based on the L-curve above. The L-curve corner point is approximated well by the red circle, which corresponds to the λ determined by V-curve minimization. This minimum, however, is a local one. It is therefore necessary to restrict the range of λ in some convenient way. Simplex B-splines can, for example, have coefficient magnitude or variance thresholds based on the identification data.

V. Results

In this section multivariate simplex P-splines using both penalty types are applied to a suitable curve fitting problem.

Identification Data and Spline Setup

Figure 5 presents the identification data, which is a single representative state space system matrix coefficient f in the flight dynamics model of a tiltrotor aircraft. See [18] for a description of the model itself.

The coefficient is sampled over a range of rotor nacelle incidence angles β_i and airspeeds V. All 53 data points are arranged along $\beta_i = [0, 30, 60, 75, 90]$, which results in strong collinearity. Also, the coefficients' non-linearity is evident. A polynomial degree of $d = 6$ was therefore selected to ensure a sufficiently good fit.

Figure 6 shows a planar view of the same data. Gaps exist where outliers were removed. The plot furthermore

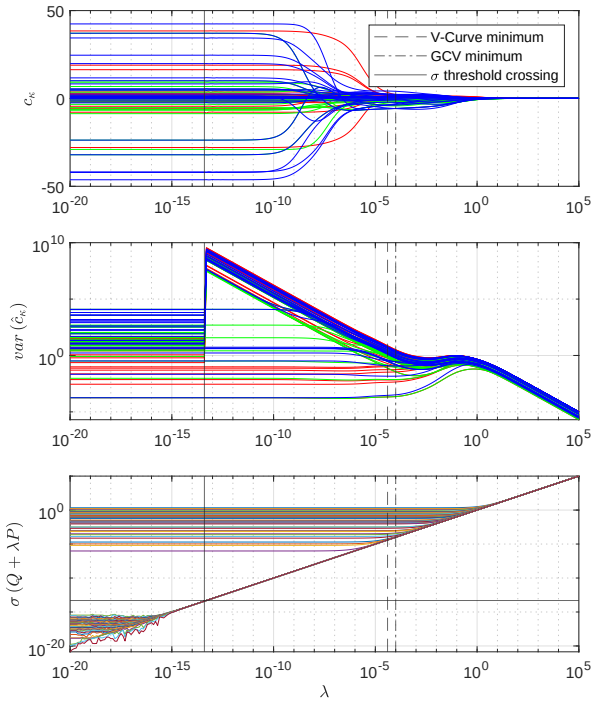
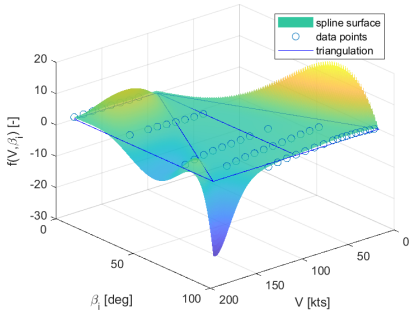
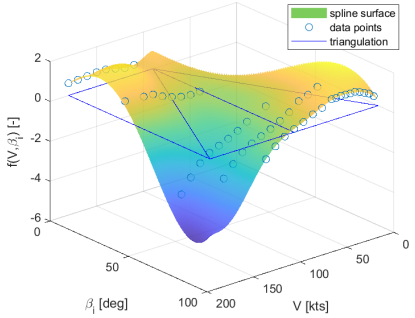


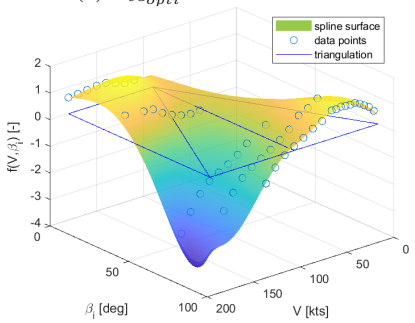
Fig. 7: RP-spline Tuning Parameter Effects (colours of $\hat{c}$ and $var\,(\hat{c})$ consistent with Figure 6, colours of σ random)



(a) $\lambda_0 = 1.00 \times 10^{-20}$ (Effectively Unpenalized)



(b) $\lambda_{vc_{opti}} = 3.98 \times 10^{-5}$



(c) $\lambda_{gcv_{opti}} = 2.00 \times 10^{-3}$

Fig. 8: RP-Spline Surfaces at Selected Tuning Parameter Values

includes a triangulation, which was chosen based on simplicity and shape. The coloured circles indicate the position of B-coefficients. Each simplex contains $\hat{d} = 28$ B-coefficients for polynomial degree $d = 6$. The total of $3 \times 28 = 84$ is more than the available 53 data points; however, adding additional coplanar data would not have helped against ill-conditioning. Inter-simplex continuity was set to C1 (first order) to ensure sufficient smoothness for future flight-controller applications.

Ridge Penalty Effect

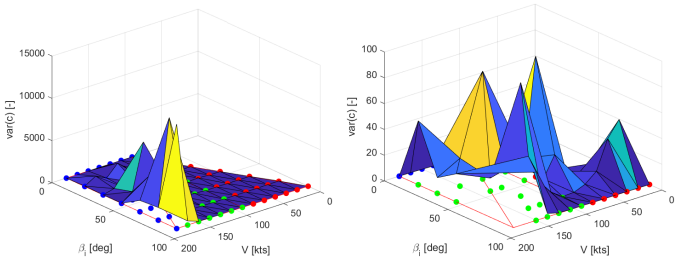
The effect of a tunable ridge penalty on B-coefficient estimates is shown in Figure 7. On top are the coefficients $\hat{c}_\kappa$, followed by their variance and augmented dispersion matrix singular values $\sigma\,(\boldsymbol{Q} + \lambda \boldsymbol{P})$. All sub-plots are functions of tuning parameter λ and include vertical lines to indicate values of interest. The two dashed lines are settings of λ suggested by information criteria, the solid line is related to singular value behaviour. In total, 251 values of λ are available to ensure sufficient resolution for information criterion optimization.

For small tuning parameter values $10^{-20} \leq \lambda \leq 4 \times 10^{-14}$, penalty effects on coefficient estimates are negligible. These are initially constant and inflated due to collinearity. The inflation can be verified in Figure 8a, which shows large peaks and valleys inside and outside the identification data envelope. Increasing the tuning parameter shrinks the coefficient estimates to zero as $\lambda \to \infty$.

The two information criteria optimal values are situated between those limits. The V-curve optimum $\lambda_{vc_{opti}}$, presented in Figure 8b, is lower than the GCV optimum $\lambda_{gcv_{opti}}$ in Figure 8c. Both offer much improved prediction, especially outside the data envelope, and the residuals are not visibly increased. Peaks are more pronounced in the V-curve solution, but the positive influence is comparable.

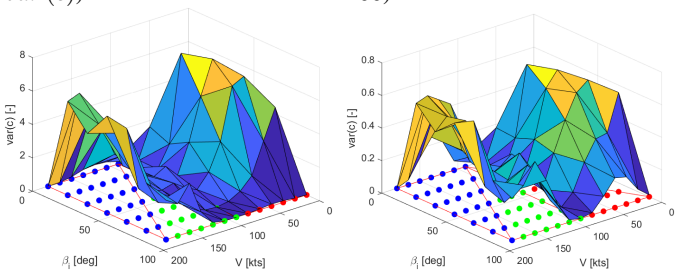
Variance behaviour is characterized by a sudden increase at $\lambda \approx 4 \times 10^{-14}$ which affects 60 out of 84 B-coefficients. This behaviour is caused by using the generalized inverse: most implementations have a threshold under which small singular values are considered to be 0. This avoids numerical problems. MATLAB's default tolerance is indicated in the σ-plot of Figure 7 by a solid horizontal line. The penalty causes the small singular values, i.e. those initially below 10^{-15}, to converge and jointly increase. Crossing the threshold clearly has an adverse effect on the generalized inverse solution. The jump, however, is followed by a steady decrease and convergence phase, and later a further simultaneous decrease toward 0.

Two points are important to note: firstly, the penalty enforces a lower limit on singular values. Those above 10^{-15} are hence initially not affected and later merge with the small ones. The condition number $\kappa = \sigma_{\max}/\sigma_{\min}$, consequently,



(a) $\lambda_0 = 1.00 \times 10^{-20}$ (All $var\,(\hat{c})$)

(b) $\lambda_0 = 1.00 \times 10^{-20}$ ($var\,(\hat{c}) \leq 100$)



(c) $\lambda_{vc_{opti}} = 3.98 \times 10^{-5}$

(d) $\lambda_{gcv_{opti}} = 2.00 \times 10^{-3}$

Fig. 9: RP-Spline Variance Surfaces (B-coefficient colours consistent with Figure 6)

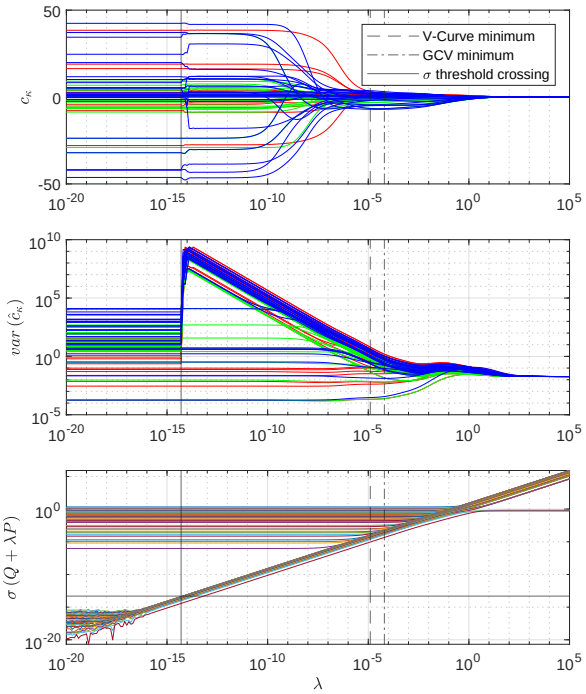


Fig. 10: DP-spline Tuning Parameter Effects

decreases until $\boldsymbol{Q} + \lambda \boldsymbol{P}$ is well-conditioned. Secondly, variances at the information criteria optima are several orders of magnitude lower than their initial values despite the intermittent jump.

Figure 9 shows the B-coefficient variance surfaces corresponding to Figure 8. Without effective penalization, shown in Figure 9a, variances are greatly inflated due to collinearity. The highest peaks occur outside the data envelope, however, the interior area is affected as well. This is visible in Figure 9b, which is filtered to only display coefficient variances ≤ 100. Figures 9c and 9d show that penalty application considerably reduces maximum variance, which is best observable by the change in vertical axis scale. Variance inside the data envelope is once again lower than outside, however, the difference is much less distinct compared to λ_0.

Difference Penalty Effect

B-coefficient difference penalties have positive effects similar to the ridge type. These, and the subtle differences in penalty behaviour, are shown in Figure 10.

The initial DP-spline coefficient estimate is equal to the initial RP-spline's. Once the singular value threshold is reached, however, coefficients start to vary and some even change sign. This is a result of small singular values not converging to a single line, but increasing in parallel. As can be seen in the bottom part of Figure 10, the threshold is not crossed at a single value of λ but at multiple values. Large jumps in coefficient variance occur similar to the RP-spline, but the peaks are spread out.

Coefficient values stabilize and shrink shortly after the variance upset. In the limit, zero is approached, which is similar to the ridge penalty. Here, however, the cause is not $\boldsymbol{P}$ but the identification data being standardized to have mean 0. Eilers and Marx in [9] have shown that the conventional P-spline becomes a polynomial of order $k-1$ in the limit. Brief investigation indicated similar simplex P-spline behaviour, but a formal proof has to be left for future research.

A further observation is variance not decreasing towards 0 but becoming constant as $\lambda \to \infty$. Singular values start diverging from approximately $\lambda = 10^0$. This is not of immediate concern as $\lambda_{vc_{opti}}$ and $\lambda_{gcv_{opti}}$ are several orders of magnitude lower; however, the rise in condition number is worth noting.

Figure 11 shows the spline and coefficient variances surfaces corresponding to the information criterion optima. Prediction quality is improved and coefficient variance reduced for both V-curve and GCV suggestions, as has been the case for the RP-spline. Performance of both penalty types is very much on par.

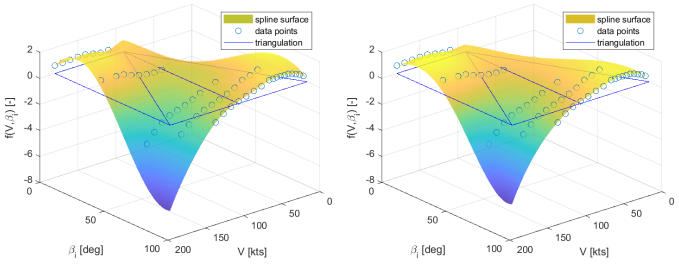
Continuity Impact

P-spline intra- and inter-simplex continuity is maintained despite the application of penalties. Table I shows that the vector norms of $H \cdot \hat{c}$ are small and the continuity conditions hence met. In fact, the P-spline's norms are smaller than the unmodified B-spline's.

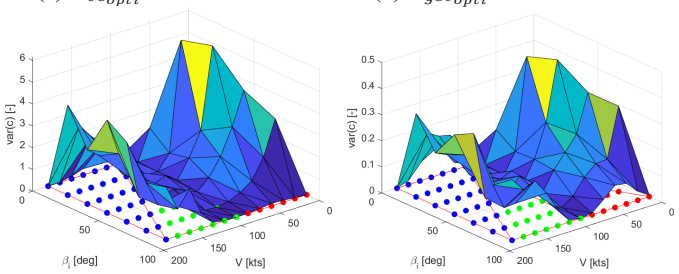
TABLE I: Continuity Condition Verification

	$\lambda = \lambda_{vc_{opti}}$	$\|H \cdot \hat{c}\|$
B-Spline $d = 6$	-	6.97×10^{-9}
RP-Spline $d = 6$	3.98×10^{-5}	3.85×10^{-12}
DP-Spline $d = 6$	1.26×10^{-5}	3.39×10^{-12}

For additional verification consider Figure 12. It shows the 1st order directional derivate splines as calculated by (6).



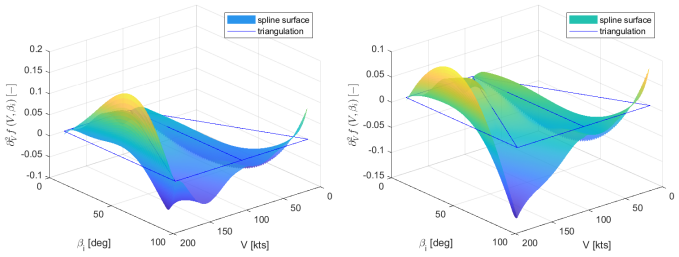
(a) $\lambda_{vc_{opti}} = 1.26 \times 10^{-5}$ (b) $\lambda_{gcv_{opti}} = 5.01 \times 10^{-4}$



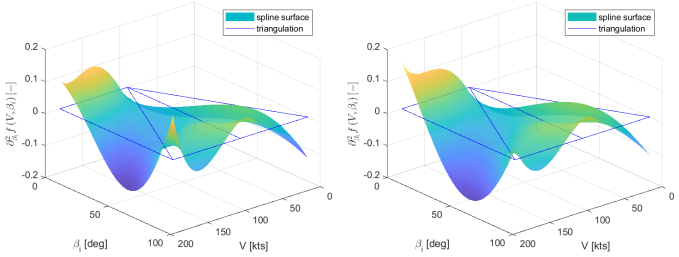
(c) $\lambda_{vc_{opti}} = 1.26 \times 10^{-5}$ (d) $\lambda_{gcv_{opti}} = 5.01 \times 10^{-4}$

Fig. 11: DP-Spline Surfaces and Variance Surfaces at Selected Tuning Parameter Values

No discontinuities are present in either direction, and C1 continuity therefore ensured.



(a) RP, V-Direction (b) DP, V-Direction



(c) RP, β_i-Direction (d) DP, β_i-Direction

Fig. 12: 1st Order Directional Derivative Surfaces of RP- and DP-Splines, $\lambda = \lambda_{vc_{opti}}$

Residuals and Smoothing Behaviour

A trade-off between estimator variance and bias has to be made when applying penalties. Reduction of coefficient variance has been successfully demonstrated; however, it comes at the expense of increased residuals. Root-mean-square error (RMSE) behaviour of the previously estimated P-splines is shown in Figure 13. It includes RMSE values of unregularized B-splines for comparison. The first is $d = 6$, on which the P-splines are based. The second is $d = 4$, the highest polynomial degree for which the KKT matrix is still well-conditioned.

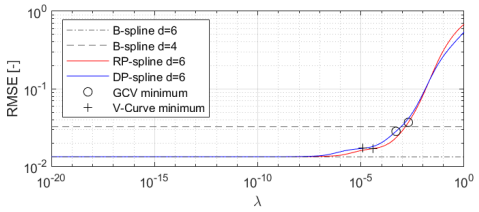


Fig. 13: P-Spline and B-Spline RMSE

Both RP- and DP-spline residuals rise as λ increases. This is to be expected, as the unaugmented OLS estimator used for B-spline gives the best linear and unbiased estimate. Any deviation caused by penalties therefore deteriorates the identification data fit. Eventually, the P-spline's error exceeds the $d = 4$ B-spline's. See Table II for error numerical values and their relative increase compared to the baseline $d = 6$ B-spline:

TABLE II: P-Spline and B-Spline RMSE Comparison

All RMSE $\times 10^{-2}$	Unregularized	V-Curve	GCV
B-Spline d=6	1.33	-	-
B-Spline d=4	3.26 (+145%)	-	-
RP-Spline d=6	-	1.69 (+28%)	3.69 (+178%)
DP-Spline d=6	-	1.71 (+29%)	2.82 (+112%)

Both DP- and RP-splines based on V-curve optima have a relatively moderate increase of approximately 30 % compared to the $d = 6$ B-spline. GCV accepts larger errors, which are close to or beyond the conventional, well-behaved $d = 4$ B-spline. Table III shows the maximum absolute errors::

TABLE III: P-Spline and B-Spline Maximum Absolute Error Comparison

All Errors $\times 10^{-2}$	Unregularized	V-Curve	GCV
B-Spline d=6	5.49	-	-
B-Spline d=4	8.30 (+51.2%)	-	-
RP-Spline d=6	-	5.40 (-1.6%)	13.71 (+149.9%)
DP-Spline d=6	-	5.47 (-0.4%)	6.91 (+25.8%)

The V-curve-tuned RP- and DP-splines have maximum absolute errors which are slightly lower than the baseline $d = 6$ B-spline. GCV, by contrast, produces significantly larger errors. Noteworthy is the large difference between RP-spline (+149.9% error) and DP-spline (+25.8% error). This is due to the ridge penalty having a more pronounced smoothing effect at larger tuning parameter values. See Figure 14, where curve roughness as measured by (32) is plotted for all B-splines and P-splines.

Both penalty types decrease R, although not monotonously. An important observation is that at *all* information criteria locations, it is less than the $d = 4$ B-spline. Given their superior residual performance it can therefore be argued that either P-spline type using V-curve should be chosen over a simple degree reduction.

VI. Conclusion

It has been shown that Tikhonov regularization of multivariate simplex B-splines reliably removes numerical issues

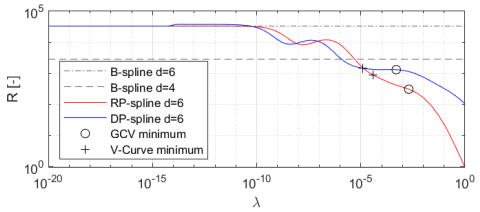


Fig. 14: P-Spline and B-Spline Curve Roughness in V-Direction

caused by strongly collinear data sets while maintaining continuity. Simplex RP-splines and DP-splines are equally suited for this application. The tuning parameter allows for a trade-off between coefficient variance and identification data residuals. V-curve and GCV proved to be useful for selecting a good compromise. However, a high λ-resolution is required to calculate them with precision.

A brief investigation confirmed that P-splines reduce curve roughness, but more research on the topic is necessary. The effect of higher-order difference penalties and their relation to differential constraints deserve particular attention. Local application of penalties has not been assessed yet and a direct comparison between the novel simplex and conventional tensor product P-splines is highly recommended. Furthermore, penalty effect demonstrations for dimensions $n > 2$ are still pending.

Tikhonov regularization and multivariate simplex splines have independently proven to be very powerful and flexible tools. It is hoped that, jointly, they enable fitting of aerodynamic models which was not previously possible.

References

[1] Milton Abramowitz and Irene A. Stegun. *Handbook of mathematical functions*. English. Washington, DC.: United States Government Printing Office, 1964, p. 877.

[2] David A. Belsley, Edwin Kuh, and Roy E. Welsch. *Regression diagnostics*. Wiley, 1980. ISBN: 0471058564.

[3] Lucas Monteiro Chaves et al. "Explaining the Generalized Cross-Validation on Linear Models". In: *Journal of Mathematics and Statistics* 15.1 (2019), pp. 298–307. ISSN: 1549-3644. DOI: 10.3844/jmssp.2019.298.307.

[4] Carl de Boor. "B-form basics". In: *Geometric modeling: algorithms and new trends* (1987), pp. 21–28.

[5] C. C. de Visser, Q. P. Chu, and J. A. Mulder. "A new approach to linear regression with multivariate splines". In: *Automatica* 45.12 (Dec. 2009), pp. 2903–2909. ISSN: 00051098. DOI: 10.1016/j.automatica.2009.09.017.

[6] C. C. de Visser, Q. P. Chu, and J. A. Mulder. "Differential constraints for bounded recursive identification with multivariate splines". In: *Automatica* 47.9 (Sept. 2011), pp. 2059–2066. ISSN: 00051098. DOI: 10.1016/j.automatica.2011.06.011.

[7] C. C. de Visser et al. *Global Nonlinear Model Identification*. 2011. ISBN: 9789085707707.

[8] Paul H. C. Eilers, Brian D. Marx, and Maria Durb. "Twenty years of P-splines". In: *Statistics and Operations Research Transactions* 39.January (2015), pp. 149–186.

[9] Paul H.C. Eilers and Brian D. Marx. "Flexible smoothing with B-splines and penalties". In: *Statistical Science* 11.2 (1996), pp. 89–102. ISSN: 08834237. DOI: 10.1214/ss/1038425655.

[10] Paul H.C. Eilers and Brian D. Marx. "Splines, knots, and penalties". In: *Wiley Interdisciplinary Reviews: Computational Statistics* 2.6 (2010), pp. 637–653. ISSN: 19390068. DOI: 10.1002/wics.125.

[11] Gianluca Frasso and Paul H.C. Eilers. "L- and V-curves for optimal smoothing". In: *Statistical Modelling* 15.1 (2015), pp. 91–111. ISSN: 14770342. DOI: 10.1177/1471082X14549288.

[12] Gene H. Golub, Michael Heath, and Grace Wahba. "Generalized cross-validation as a method for choosing a good ridge parameter". In: *Technometrics* 21.2 (1979), pp. 215–223. ISSN: 15372723. DOI: 10.1080/00401706.1979.10489751.

[13] Jürgen Groß. "Restricted ridge estimation". In: *Statistics and Probability Letters* 65.1 (2003), pp. 57–64. ISSN: 01677152. DOI: 10.1016/j.spl.2003.07.005.

[14] Christian Hansen. "Analysis of Discrete Ill-Posed Problems by Means of the L-Curve". In: *SIAM Review* 34.4 (2010), pp. 561–580.

[15] Arthur E. Hoerl and Robert W. Kennard. "Ridge Regression : Biased Estimation for Nonorthogonal Problems". In: *Technometrics* 12.1 (1970), pp. 55–67.

[16] Kathrin Kagerer. "A hat matrix for monotonicity constrained B-spline and P-spline regression". 2015. URL: https://epub.uni-regensburg.de/31450/.

[17] Brian D. Marx and Paul H.C. Eilers. "Multidimensional penalized signal regression". In: *Technometrics* 47.1 (2005), pp. 13–22. ISSN: 00401706. DOI: 10.1198/004017004000000626.

[18] Hafiz Noor Nabi and Giuseppe Quaranta. "A Quasi-Linear Parameter Varying (QLPV) modeling approach for real time piloted simulation of tiltrotor". In: *45th European Rotorcraft Forum 2019, ERF 2019*. Vol. 2. 2019, pp. 989–1000. ISBN: 9781713805922.

[19] C. Radhakrishna Rao. *Linear statistical inference and its applications*. English. 2nd ed. New York: Wiley, 2002. ISBN: 0471218758 9780471218753.

[20] H. J. Tol et al. "Nonlinear multivariate spline-based control allocation for high-performance aircraft". In: *Journal of Guidance, Control, and Dynamics* 37.6 (Nov. 2014), pp. 1840–1862. ISSN: 15333884. DOI: 10.2514/1.G000065.

[21] Wessel N. van Wieringen. *Lecture notes on ridge regression*. 2015. arXiv: 1509.09169. URL: http://arxiv.org/abs/1509.09169.

Part III

Discussion

7

Conclusion

What started off as an effort to convert a look-up table based tiltrotor qLPV flight dynamics model to multivariate simplex B-splines ended with the fitting of a single stability derivative. Despite this, the project is considered a success.

First of all, a large amount of experience was gained in project management. The difficulties in fitting a seemingly simple data set were initially severely underestimated. This resulted in the creation of a system identification pipeline (see Appendix B), which was eventually unused. Also, a disproportionate amount of time and effort was spent on futilely fixing what was in hindsight a very common and well-researched problem. Therefore, literature should have been consulted much earlier in the process thus applying proven solutions instead of a trial-and-error approach.

That being said, everything came together beautifully once the correct search terms were established and the relevant literature had been collected. Tikhonov regularization can be integrated conveniently into least-squares B-coefficient estimators, and tuned on a per-simplex basis. The ridge penalty is trivial to implement, yet a powerful tool against ill-conditioning. Coefficient difference penalties have a similarly positive effect, but are more customizable at the expense of increased complexity.

The observations presented in this report give reason to believe that the multivariate simplex P-spline is indeed a useful enhancement to classic simplex B-splines. It can be used to fit collinear data and therefore achieve the research objective. In addition, further applications are within in reach. Firstly, ridge regression is under certain conditions known to have lower estimator error as compared to basic least-squares. Secondly, tensor product P-splines have been applied successfully for multidimensional smoothing purposes. It appears that their simplex counterpart is equally capable.

One of the P-spline's advantages is the tunable penalty. Unfortunately, this is also its main drawback because optimization of information criteria requires numerous fits of the same data set. Computation time scales exponentially with a model's dimension and spline's polynomial degree. This can become an issue for large or complex data sets. Consequently, increasing the fitting process' efficiency is one of the strongest recommendations in Chapter 8.

Finally, it is expected that the multivariate simplex P-spline is a very useful new tool against data collinearity. It can also be used for a wider range of applications. However, wielding it efficiently will take time and practice.

8

Recommendations

The recommendations have been split into two groups due to the project's history:

1. Spline-based tiltrotor flight-dynamics model

   (a) Replace look-up tables by multivariate simplex P-splines. The results should also be compared to unpenalized B-splines of a polynomial degree low enough to avoid collinearity issues.
   (b) Investigate as to whether a *single* tuning parameter value works for fitting simplex P-splines to *all* stability & control derivatives and trim points. If not, find an efficient way for individual optimization of λ.
   (c) Obtain consistent data for intermediate settings of nacelle incidence angle. This would avoid ill-conditioned regression matrices in the first place.

2. Further simplex P-spline research

   (a) Compare the simplex P-spline (coefficient difference penalty version) to the tensor product P-spline which it is based on. There is reason to believe that the simplex variant inherited all advantages of regularization, while having none of the disadvantages of tensor product B-splines (inability to fit scatter data, cumbersome computations in higher dimensions etc.)
   (b) Investigate the relation between difference penalties and differential constraints. This would contribute to a stronger mathematical foundation and possibly new penalty options.
   (c) Examine the effect of second and higher-order difference penalties, especially what happens close to simplex vertices. Such vertices where not enough coefficients are available to construct difference constraints in all directions.
   (d) Investigate if difference penalties can be expanded across simplex borders. Perhaps an integration with the smoothness matrix is possible.
   (e) Search the literature for alternative information criteria. There might be ones which are more robust or otherwise superior to the V-curve or GCV.
   (f) Find a way to automatically adjust tuning parameter search range and resolution. The current approach of wide search range and fixed λ-increments is very inefficient.
   (g) Use simplex P-splines to smooth data sets featuring correlated noise.
   (h) Try local application of simplex P-splines. Efficient optimization of not one but multiple λ simultaneously is certainly a challenge.
   (i) Try simplex P-splines in dimensions higher than two. The penalties are flexible enough to support this.

Part IV

Appendix

A

Additional Demonstration Cases

A.1. Overview

This section presents three artificial data sets, which are fit using P-splines and compared to conventional B-splines. The sets are based on Franke's bivariate test function, shown in Figure A.1. Hence, unlike the tiltrotor set, verification data is readily available to evaluate root-mean-square error performance. The first set contains coplanar data similar to a tiltrotor stability derivative, the second requires extrapolation and the third requires smoothing. See Table A.1 for an overview.

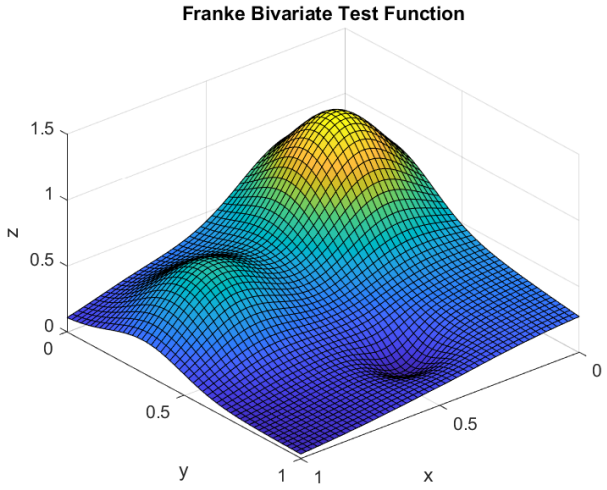


Figure A.1: The Function used for Identification and Verification Data, see [11] for Details

A.2. Case 1: Collinear Set

The top part of Figure A.2 shows a comparison between spline surfaces generated with and without regularization. On the left is the unregularized B-spline. It features a peak which is several orders of magnitude larger than the identification data. The data therefore appears flat, even though it has been sampled from the test function in Figure A.1 (with the addition of some noise).

On the right is the regularized DP-spline, where λ has been determined by locally minimizing the V-curve (bottom left corner). Predictive performance is much improved, which can be verified in the RMSE plot (bottom right corner). Initially, the difference between original test function and estimated spline on the triangulation is very large. The error becomes smaller as the tuning parameter increases and has a minimum at approximately 10^{-2}. Beyond this point RMSE increases due to the penalty becoming dominant.

The V-curve plot highlights the importance of selecting the right local minimum. This can be done by, for example, setting a threshold for coefficient estimate maxima or the condition number. The given

Table A.1: Demonstration Case Overview

	Case 1	Case 2	Case 3
Data Set			
- Base Function	Franke	Franke	Franke
- Data Point #	150	300	300
- Structure	coplanar	random	random
- Noise Magnitude	+/- 0.5 %	+/- 0.5 %	+/- 2.5 %
Triangulation			
- Simplex #	2	2	2
- Shape	non-uniform	extended square	tight square
- Regularized	all	all	all
Spline			
- Dimension	2	2	2
- Poly. Degree	6	6	6
- Continuity	C1	C1	C1
Penalty			
- Type	DP	DP	DP
- Order	1st	1st	1st
Information Criterion			
- Type	V-curve	V-curve	GCV

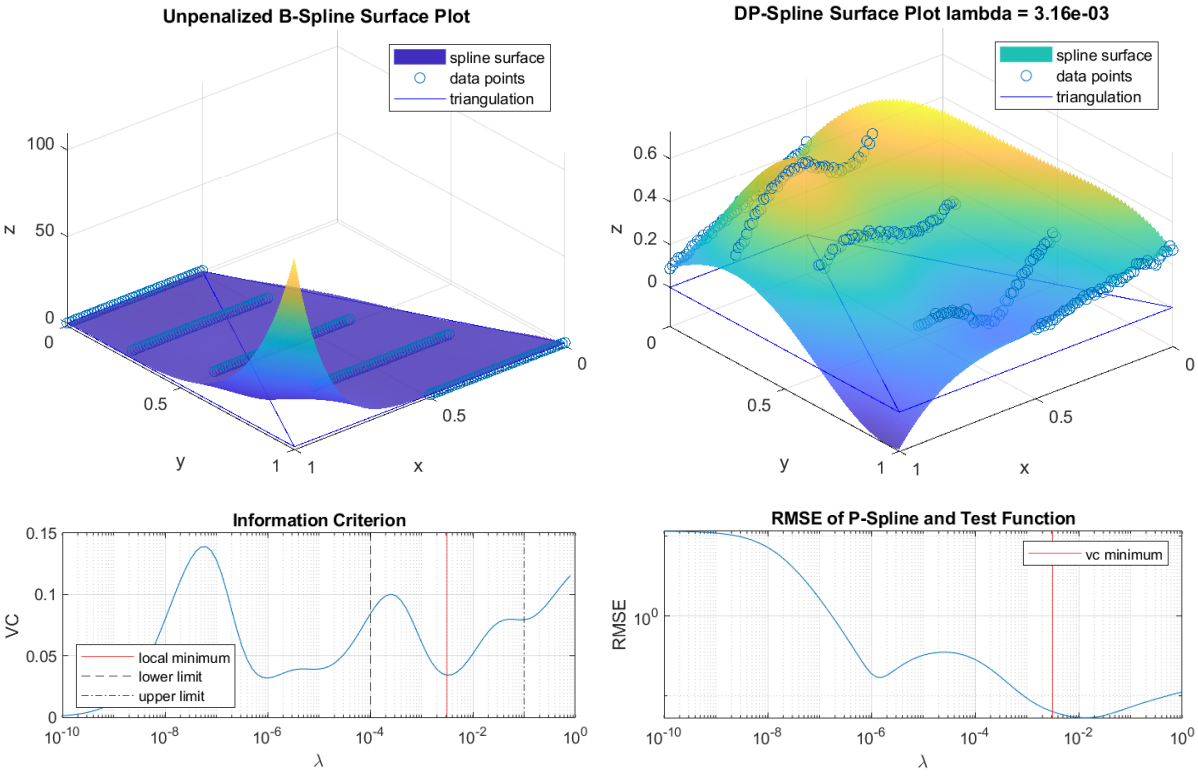


Figure A.2: Demonstration Case 1

example yields very good results, as the RMSE value is close to its minimum.

A.3. Case 2: Extrapolation Set

The second set is shown in Figure A.3. A large number of identification data points is available and organized in a random fashion, however, only 60 % of the triangulation surface is covered. This leads to

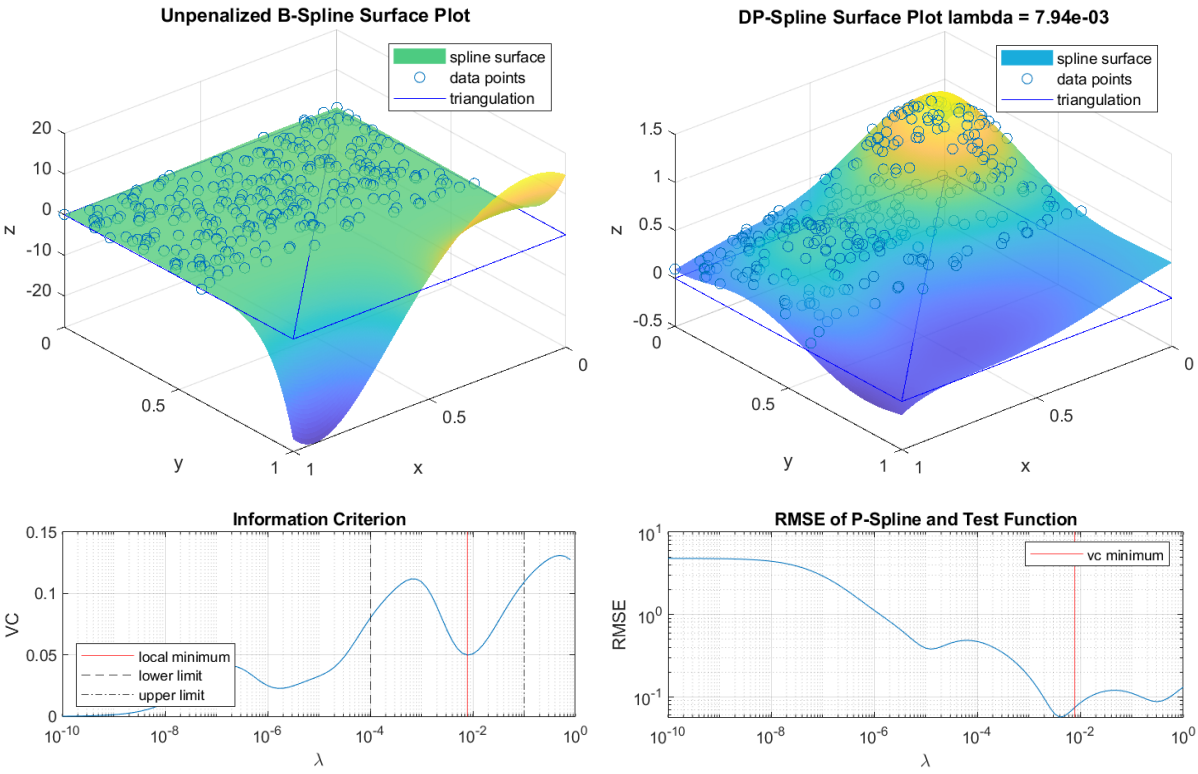


Figure A.3: Demonstration Case 2

large coefficient magnitudes outside the identification data domain and consequently bad verification data RMSE. Regularization using a coefficient difference penalty solves this issue. Once again, the V-curve information criterion is able to find a point close to the RMSE minimum.

A.4. Case 3: Noisy Set

The third and last example is shown in Figure A.4. Here, the issue is neither the data's structure or distribution, but overfitting due to high spline polynomial degree. Using a coefficient difference penalty reduces the problem, but does not eliminate it entirely. Nevertheless, the GCV information criterion provides a better solution in terms of RMSE (V-curve lead to underfitting in this particular case).

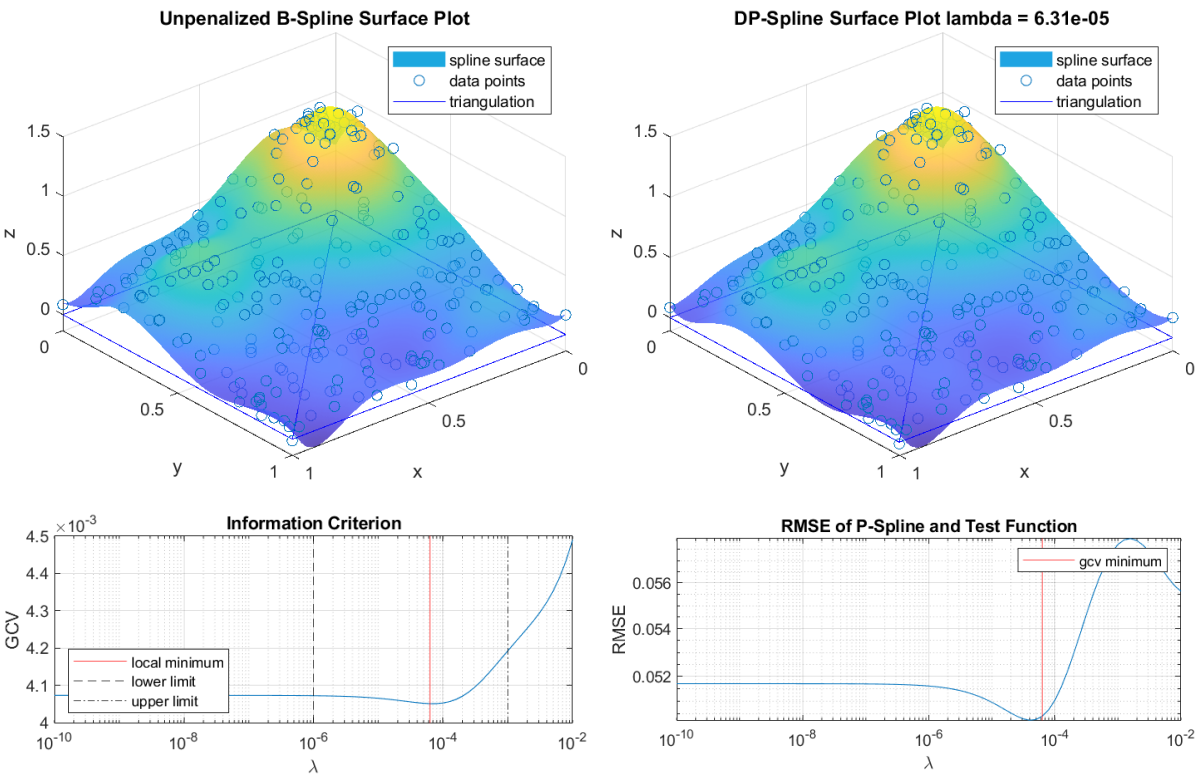


Figure A.4: Demonstration Case 3

B

Spline Model Development Code Structure

The code which was intended for creating, verifying and validating the spline-based qLPV tiltrotor flight dynamics model is structured as shown in Figure B.1.

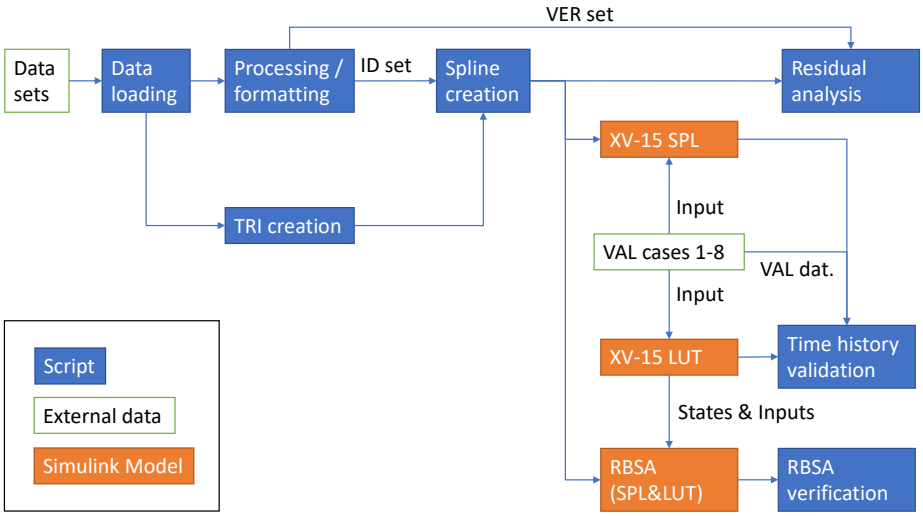


Figure B.1: Code Structure Diagram

Data is loaded and pre-processed, e.g. to normalize values and remove outliers, and used to create a suitable triangulation TRI. It is then split into a part for identification (ID) and a part for verification (VER). The ID data is used to fit multivariate simplex splines, which are compared to the VER data in terms of residuals. The splines (SPL) are also directly integrated in the XV-15 qLPV Simulink model in place of look-up tables (LUT). Using time series of known control inputs allows comparisons of model output and validation (VAL) data.

An additional verification method is the analysis of rigid body state accelerations (RBSA). It is used as a back-up in case the model ”blows up” after a small number of time steps.