

2014

# An Investigation of NURBS-Based Deformable Image Registration

Travis J. Jacobson  
jacobsontj@vcu.edu

Follow this and additional works at: <http://scholarscompass.vcu.edu/etd>

 Part of the [Other Medical Sciences Commons](#), [Other Medicine and Health Sciences Commons](#), [Other Physics Commons](#), and the [Radiology Commons](#)

© The Author

---

Downloaded from

<http://scholarscompass.vcu.edu/etd/3563>

This Thesis is brought to you for free and open access by the Graduate School at VCU Scholars Compass. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of VCU Scholars Compass. For more information, please contact [libcompass@vcu.edu](mailto:libcompass@vcu.edu).

# **An Investigation of NURBS-based Deformable Image Registration**

A thesis dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy at Virginia Commonwealth University

by

Travis J. Jacobson

Bachelor of Science, Tulane University, May 2006

Director: Martin J. Murphy, Ph.D.

Professor, Medical Physics Graduate Program

Department of Radiation Oncology

Virginia Commonwealth University

Richmond, Virginia

July 16, 2014

## Acknowledgements

I would like to first thank my advisor Dr. Martin Murphy for his guidance in this project. Being an effective mentor is a delicate balance between doing too much and doing too little, and I'm glad he let me flounder without there being any real danger of drowning. He allowed me the time to grapple with a lot of conceptual and procedural problems, and I learned a great deal more than I would have otherwise.

I owe a debt of gratitude to the remaining members of my doctoral committee, Dr. Geoff Hugo, Dr. Jeffrey Williamson, and Dr. Alan Docef for providing me with constructive criticisms and insight on this work, as well as suggestions for future work. In addition to that, they were incredibly cooperative and flexible in dealing with the logistical difficulties of scheduling the dissertation defense on short notice as deadlines loomed. I am especially grateful to the graduate program director Dr. Geoff Hugo, who was instrumental in making sure all of the bureaucratic exigencies were fulfilled.

I would also like to thank past professors and mentors, most notably Dr. Jeffrey Siebers, the former graduate program director, who provided me with years of support and advice.

Special thanks go out to my colleagues and to my friends (and the large intersection between the two). There are too many to list here, and that's a good problem to have.

Finally, I would like to thank my parents, Tom and Tammy Jacobson for their endless love and encouragement, their patience and understanding, and their model of rectitude and resilience. This effort is dedicated to them, to my brother Tyler, and to the loving memory of my sister Tatum.

## Table of Contents

List of Tables.....	v
List of Figures.....	vi
List of Abbreviations.....	xviii
Abstract.....	xix
1. Introduction.....	1
1.1 Background and Significance.....	1
1.2 Hypothesis and Goals.....	5
2. B-spline Mathematics.....	6
2.1 B-spline Curves.....	6
2.2 B-spline Tensors.....	9
2.3 Uses of B-splines in DIR.....	9
3. Non-Uniform Rational B-Splines.....	12
3.1 NURBS Curve Fitting.....	17
3.1.1 B-spline Curve Fitting.....	18
3.2 Optimizing Weights for Rational B-Spline Fitting.....	22
3.2.1 Weight Optimization using Gradient Descent Method.....	23
3.2.2 Direct Solution for Control Point Weights.....	33
3.3 Non-uniform Knot Placement in B-Spline Fitting.....	39
3.3.1 Weight Optimization using Gradient Descent Method.....	42
3.3.2 Direct Solution for Control Point Weights.....	47

3.3.3	Force Equilibration Method.....	50
3.3.4	Knot Insertion by Bisection of Error.....	60
3.4	Summary of Investigation.....	66
4.	Non-Uniform B-Spline Tensor Products.....	67
4.1	Non-Uniform Knot Placement for Tensor Product B-Splines.....	67
4.2	DVF Surface Fitting with Non-Uniform B-Splines.....	72
4.3	DVF Volume Fitting with Non-Uniform B-Splines.....	83
4.4	Summary of Investigation.....	87
5.	Non-Uniform B-splines Deformable Image Registration.....	89
5.1	2D DIR.....	90
5.2	3D DIR.....	100
5.2.1	Incremental Knot Insertion DIR Accuracy.....	103
5.2.2	Incremental Knot Insertion DIR Computation Time.....	116
5.2.3	Multiple Knot Insertion DIR.....	119
5.3	Summary of Investigation	
6.	Conclusions and Discussion.....	124
6.1	Conclusions.....	124
6.2	Discussion and Future Work.....	125
7.	References.....	128

## List of Tables

Table 1: Fitting error variation with respect to rank of matrix  $M$

..... 36

## List of Figures

Figure 1: First-degree basis functions, $U = \{0, 0, 0.2, 0.4, 0.6, 0.8, 1.0, 1.0\}$	8
Figure 2: Quadratic basis functions, $U = \{0, 0, 0, 0.2, 0.4, 0.6, 0.8, 1.0, 1.0, 1.0\}$	8
Figure 3: The effect of control points weights.	13
Figure 4: Rational basis functions with $w_3 = 10$	14
Figure 5: Basis functions with knot multiplicity equal to the degree at parameter value $u = \frac{1}{4}$	14
Figure 6: Knot multiplicities allow for discontinuities in the B-spline Curve	15
Figure 7: A uniform B-spline curve (shown in red) is the weighted sum of the B-spline basis functions (shown in black). The spatial extent (i.e, local support) of each B-spline basis function is determined by the spacing of the knots (denoted by tick marks).	16
Figure 8: A non-uniform B-spline curve (shown in red). The non-uniform spacing of knots can increase or decrease the local derivative of the curve.	16
Figure 9: A uniform B-spline (shown in blue) with 8 control points fit to a 1D Gaussian (shown in red).	21
Figure 10: A uniform B-spline (shown in blue) with 13 control points fit to a 1D Gaussian (shown in red).	21
Figure 11: A uniform B-spline (shown in blue) with 19 control points fit to a 1D Gaussian (shown in red).	

.....	22
Figure 12: Control point weights are adjusted via gradient descent. The 5 control point values obtained from solving equation 3.5 are held fixed.	
.....	25
Figure 13: Control point weights are adjusted via gradient descent. The 10 control points values obtained from solving equation 3.5 are held fixed.	
.....	25
Figure 14: Control point weights are adjusted via gradient descent. The 15 control points values obtained from solving equation 3.5 are held fixed.	
.....	26
Figure 15: The rational B-spline can fit the 1D Gaussian with just five control points.	
.....	27
Figure 16: The rational B-spline can fit the 1D Gaussian with just five control points.	
.....	27
Figure 17: Fitting the z-direction 1D DVF with a rational B-spline using the gradient descent method to compute five control point values and their associated weights decreases the fitting error by more than a factor of two compared to the conventional B-spline fit with the same number of control points.	
.....	28
Figure 18: The inclusion of adjustable weights again demonstrates improved fitting accuracy as compared to the conventional B-spline fit using 15 control points.	
.....	29
Figure 19: T The inclusion of adjustable weights again demonstrates improved fitting accuracy as compared to the conventional B-spline fit using 15 control points.	
.....	29
Figure 20: Fitting the x-direction 1D DVF with a rational B-spline using the gradient descent method to compute five control point values and their associated weights decreases the fitting error by more than a factor of five compared to the conventional B-spline fit with the same number of control points.	
.....	30



Figure 21: The inclusion of adjustable weights again demonstrates improved fitting accuracy as compared to the conventional B-spline fit using 10 control points.

.....30

Figure 22: The inclusion of adjustable weights again demonstrates improved fitting accuracy as compared to the conventional B-spline fit using 15 control points.

.....31

Figure 23: Comparison of errors of rational and conventional B-spline when fitted to the 1D DVF in the x direction. Conventional B-splines with twice as many control points exhibit similar fitting accuracy to rational B-splines with half the control points plus their associated weights (same number of total free parameters).

.....32

Figure 24: Comparison of errors of rational and conventional B-spline when fitted to the 1D DVF in the z direction. Conventional B-splines with twice as many control points exhibit similar fitting accuracy to rational B-splines with half the control points plus their associated weights (same number of total free parameters).

.....32

Figure 25: NURBS Surface fit with rank = 98.

.....37

Figure 26: NURBS surface fit with rank = 99.

.....38

Figure 27: An open uniform quadratic B-spline. Below the curve is the basis functions defined on the knots  $k_i$ . Each basis function has an associated control point displayed as a circle of the same color. The knot positions determine the shape of the basis functions, which in turn determine the local support of the control points. The vertical lines labelled  $k_i$  represent the knots on this interval. Here, knots of multiplicity three are denoted by the bunched hash marks at either end of the interval. The knot multiplicity forces the basis function to be 1 and "clamps" the curve to its first and last control point.

.....40

Figure 28: Simplex solution for one free knot, which coincides with the uniform knot distribution.

.....44

Figure 29: Simplex solution for two free knots, which exhibits advantage in asymmetric distributions even on symmetric target functions. The internal duplicate knot is indicated by the red hash mark.

.....44

Figure 30: The solution for three free knots is repeated knot values at the central parameter value. Repeated internal knots are indicated by red hash marks on either side of the true knot value in black. Knots values with multiplicity equal to the degree of the B-spline can produce cusps and discontinuities.

.....45

Figure 31: The solution for four free knots, which again exhibits repeated knot values (in red) and asymmetry.

.....46

Figure 32: The solution for five free knots, which again exhibits repeated knot values (in red) and asymmetry.

.....46

Figure 33: The solution for six free knots. There are no longer any repeated knots, and the spacing of the central knots closely resembles the spacing of the uniform distribution.

.....47

Figure 34: The first three inserted knots are all placed at the same central parameter value. Duplicate knots are displayed in red.

.....48

Figure 35: Figure 35: The discontinuous kink created by the multiple knots reduces the shape control in the surrounding curve segments.

.....49

Figure 36: The KIM method overcompensates by adding too many knots in areas of high local gradient.

.....50

Figure 37: The first two knots are inserted at the maximum error, located at center of the Gaussian.

.....52

Figure 38: Force equilibration redistributes the two internal knots.

.....52

Figure 39: The third knot is placed at the midpoint of the Gaussian.	53
Figure 40: Force equilibration redistributes the three internal knots.	53
Figure 41: Another knot is inserted at parameter value corresponding to the center of the Gaussian.	54
Figure 42: The force equilibration scheme has prevented the knot pile up observed in the KIM method.	54
Figure 43: The largest error continues to reside at the center of the curve.	55
Figure 44: The force equilibration scheme continues to spread the control points away from the center, thereby avoiding a knot distribution that is too narrow.	55
Figure 45: The final knot distribution for the force equilibration method.	56
Figure 46: Comparison of the sum of squared errors normalized to the initial error.	57
Figure 47: The 1D DVF profile in the x-direction fitted with a uniform B-spline.	58
Figure 48: The 1D DVF profile in the x-direction fitted with a non-uniform B-spline computed by the Simplex method.	59
Figure 49: The 1D DVF profile in the x-direction fitted with a non-uniform B-spline computed by knot insertion at location of maximum error followed by knot force equilibration (KIM+FE).	59

Figure 50: The 1D DVF profile in the  $x$ -direction fitted with a non-uniform B-spline computed by knot insertion at location of maximum error. The repeated knot insertion in a small region led to an algorithmic error, motivating a new knot insertion scheme.

.....60

Figure 51: The 1D DVF profile in the  $x$ -direction fitted with a non-uniform B-spline computed by the KIB method.

.....61

Figure 52: The 1D DVF profile in the  $x$ -direction fitted with a non-uniform B-spline computed by KIB+FE.

.....62

Figure 53: Comparison of the sum of squared errors of each algorithm fitting the 1D DVF profile in the  $x$ -direction.

.....62

Figure 54: Uniform B-spline fit to 1D DVF profile in the  $z$ -direction.

.....63

Figure 55: Non-uniform B-spline fit using Simplex method.

.....64

Figure 56: Non-uniform B-spline fit using KIB method.

.....64

Figure 57: Non-uniform B-spline fit using KIB+FE method.

.....65

Figure 58: Non-uniform B-spline fit using KIM+FE method.

.....65

Figure 59: Comparison of the sum of squared errors of each algorithm fitting the 1D DVF profile in the  $z$ -direction.

.....66

Figure 60: Inserting a knot (in black) to the Cartesian point  $(a,b)$  requires adding the values  $a$  and  $b$  to the  $x$  and  $y$  knot vectors, respectively.

.....69

Figure 61: The first free knot in each direction is placed such that it bisects the integral error of the projected 1D error profile.

.....70

Figure 62: The target surface is re-fit with the new knot vector. The fitting error is updated, and the knot is allowed to move according to the force equation. Here the initial knot position is represented by the unfilled circle and the new knot position is represented by the solid circle.

.....71

Figure 63: New knots (in red) are again inserted according to the error bisection rule.

.....71

Figure 64: The target surface is once again re-fit, and the knots are allowed to move along the updated error profiles according to the force equation.

.....72

Figure 65: Contour plot of the target surface (top panel) generated from a coronal slice with a qualitative comparison of the four B-spline fitting methods using 100 control points.

.....74

Figure 66: Contour plots of the surfaces generated by the four B-spline fitting methods applied to a coronal slice of the DVF data.

.....75

Figure 67: Comparison of the sum of squared differences between each B-spline fitting method and the coronal DVF slice. The errors are normalized to the fit with 16 control points (0 free knots).

.....76

Figure 68: Contour plot of the target surface (top panel) generated from an axial slice of the DVF data with a comparison of the four B-spline fitting methods using 100 control points.

.....77

Figure 69: Contour plots of the surfaces generated by the four B-spline fitting methods applied to an axial slice of the DVF data using 256 control points.

.....78

Figure 70: Comparison of fitting the error (sum of squared differences) to the axial surface for each B-spline fitting method for a given number of control points. The errors are normalized to the fit with 16 control points (0 free knots).	79
Figure 71: Contour plot of the target surface (top panel) generated from a sagittal slice of the DVF data with a comparison of the four B-spline fitting methods using 100 control points.	80
Figure 72: Contour plots of the surfaces generated by the four B-spline fitting methods applied to a sagittal slice of the DVF data using 256 control points.	81
Figure 73: Comparison of the sum of squared differences between each B-spline fitting method and the sagittal DVF slice. The errors are normalized to the fit with 16 control points (0 free knots).	82
Figure 74: Contour plot of the average coronal slice.	83
Figure 75: Contour plot of the average sagittal slice.	84
Figure 76: Contour plot of the average axial slice.	84
Figure 77: Comparison of the sum of squared differences between each B-spline fitting method and the z-component DVF volume. The errors are normalized to the fit with 16 control points (0 free knots).	86
Figure 78: Comparison of the sum of squared differences of a uniform B-spline with equal knot spacing in each axis and a uniform B-spline with an equal number of knots per axis. Equal knot spacing is conventional method, and it is seen here to be superior to the knot distribution with an equal number of knots per axis.	87
Figure 79: Flowchart of the non-uniform refinement algorithm.	93

Figure 80: Comparison of the non-uniform (upper) and uniform (lower) B-spline registration errors in the lung with an 8 by 8 control point grid. .....	95
Figure 81: Comparison of the non-uniform (upper) and uniform (lower) B-spline registration errors with a 10 by 10 control point grid. .....	96
Figure 82: Comparison of the non-uniform (upper) and uniform (lower) B-spline registration errors with a 12 by 12 control point grid. .....	96
Figure 83: Comparison of the non-uniform (upper) and uniform (lower) B-spline registration errors with a 15 by 15 control point grid. .....	97
Figure 84: Non-uniform refinement results in a reduction of local errors in the ROI. .....	98
Figure 85: Non-uniform refinement results in an overall reduction of registration error. .....	99
Figure 86: Axial error map with 1D projections and the resultant knot vectors. .....	101
Figure 87: Sagittal error map with 1D projections and the resultant knot vectors. .....	102
Figure 88: Coronal error map with 1D projections and the resultant knot vectors. .....	103
Figure 89: Error comparison of the three incremental B-spline DIR methods. Error is plotted as a function of the log of the number of control points. .....	104
Figure 90: Difference images in a coronal slice of the CT pair before registration (top panel), and after registration with the three methods (uniform, KIB+FE, KIB). The frames on the right overlay the knot distributions with 6 free knots. The uniform method had a 10 by 10 by 10 control point grid, while both non-uniform methods selected a control point grid of 11 by 10 by 9. .....	106

Figure 91: Coronal difference images resulting from registration using uniform, KIB+FE, and KIB methods with approximately 10,000 control points.

.....107

Figure 92: Contour plots of a coronal plane containing the z-component of the DVF. The left panel is the DVF generated by the uniform B-spline method with 1,000 control points, the middle panel is the DVF generated by the non-uniform B-spline method with 990 control points, and the right panel is the target DVF generated by the Demon's algorithm.

.....108

Figure 93: Contour plots of a coronal plane containing the z-component of the DVF. The left panel is the DVF generated by the uniform B-spline method with 10,000 control points, the middle panel is the DVF generated by the non-uniform B-spline method with 990 control points, and the right panel is the target DVF generated by the Demon's algorithm.

.....109

Figure 94: Difference images in a sagittal slice of the CT pair before registration (top panel), and after registration with the three methods (uniform, KIB+FE, KIB). The frames on the right overlay the knot distribution with 6 free knots. The uniform method had a 10 by 10 by 10 control point grid, while both non-uniform methods selected a control point grid with dimensions of 11 by 10 by 9.

.....110

Figure 95: Sagittal difference images resulting from registration using uniform, KIB+FE, and KIB methods with approximately 10,000 control points.

.....111

Figure 96: Contour plots of a sagittal plane containing the z-component of the DVF. The left panel is the DVF generated by the uniform B-spline method with 1,000 control points, the middle panel is the DVF generated by the non-uniform B-spline method with 990 control points, and the right panel is the target DVF generated by the Demon's algorithm.

.....112

Figure 97: Contour plots of a sagittal plane containing the z-component of the DVF. The left panel is the DVF generated by the uniform B-spline method with 10,000 control points, the middle panel is the DVF generated by the non-uniform B-spline method with 990 control points, and the right panel is the target DVF generated by the Demon's algorithm.

.....112



Figure 98: Difference images in a sagittal slice of the CT pair before registration (top panel), and after registration with the three methods (uniform, KIB+FE, KIB). The frames on the right overlay the knot distribution with 6 free knots. The uniform method had a 10 by 10 by 10 control point grid, while both non-uniform methods selected 11 by 10 by 9 control point grid.

..... 113

Figure 99: Sagittal difference images resulting from registration using uniform, KIB+FE, and KIB methods with approximately 10,000 control points.

..... 114

Figure 100: Contour plots of an axial plane containing the z-component of the DVF. The left panel is the DVF generated by the uniform B-spline method with 10,000 control points, the middle panel is the DVF generated by the non-uniform B-spline method with 990 control points, and the right panel is the target DVF generated by the Demon's algorithm.

..... 115

Figure 101: Contour plots of an axial plane containing the z-component of the DVF. The left panel is the DVF generated by the uniform B-spline method with 10,000 control points, the middle panel is the DVF generated by the non-uniform B-spline method with 990 control points, and the right panel is the target DVF generated by the Demon's algorithm.

..... 115

Figure 102: Adding a knot to the B-spline curve will only affect  $p+1$  basis functions. Because of this, it is only necessary to compute  $p+1$  new control point values

..... 117

Figure 103: Comparison of the total time required to compute the three B-spline methods. The uniform B-spline method requires nearly twice as long to compute as the KIB non-uniform method. Note that from Figure 99, the error upon completion of the registration with 10,000 control points produces very nearly the same error.

..... 118

Figure 104: Comparison of the mean squared intensity difference after each of the three multilevel B-spline DIR methods. The non-uniform method slightly out-performs the uniform methods until after 100,000 control points.

..... 120

Figure 105: Comparison of the total time required to compute the three B-spline multilevel methods as a function of the number of control points.

..... 121

Figure 106: Comparison of the times to compute each multilevel registration, averaged over the 9 breathing phases.

.....122

Figure 107: Comparison of the errors versus computation time, averaged over the 9 breathing phases.

.....123

## List of Abbreviations

BFGS	Broyden-Fletcher-Goldfarb-Shanno
CAD	Computer Aided Design
CAE	Computer Aided Engineering
CT	Computed Tomography
DIR	Deformable Image Registration
DVF	Displacement Vector Field
FE	Force Equilibration
GPU	Graphics Processing Unit
GN	Gauss-Newton
KIB	Knot Insertion by Bisection
KIM	Knot Insertion at Maximum
L-BFGS	Limited-memory Broyden-Fletcher-Goldfarb-Shanno
LM	Levenberg-Marquardt
MRI	Magnetic Resonance Imaging
NUBS	Non-Uniform B-Spline
NURBS	Non-Uniform Rational B-Spline
POPI	Point-validated Pixel-based
ROI	Region of Interest
SSD	Sum of Squared Differences

## **Abstract**

### AN INVESTIGATION OF NURBS-BASED DEFORMABLE IMAGE REGISTRATION

By Travis James Jacobson, B.S.

A dissertation submitted in partial fulfillment of the requirements for the degree Doctor of Philosophy at Virginia Commonwealth University.

Virginia Commonwealth University, 2014

Major Director: Martin J. Murphy, Ph.D.

Professor, Medical Physics Graduate Program

Department of Radiation Oncology

Deformable image registration (DIR) is an essential tool in medical image processing. It provides a means to combine image datasets, allowing for intra-subject, inter-subject, multi-modality, and multi-instance analysis, as well as motion detection and compensation. One of the most popular DIR algorithms models the displacement vector field (DVF) as B-splines, a sum of piecewise polynomials with coefficients that enable local shape control. B-splines have many advantageous properties in the context of DIR, but they often struggle to adequately model steep local gradients and discontinuities. This dissertation addresses that limitation by proposing the replacement of conventional B-splines with a generalized formulation known as a Non-Uniform Rational B-Splines (NURBS). Beginning with the 1D fitting, heuristic rules are developed to determine the values of the additional free parameters introduced by NURBS. These rules are subsequently modified and extended to the 2D and 3D fitting of anonymized and publicly available patient DVFs. Based on the lessons learned from these increasingly complex test cases, a 2D DIR scheme is developed and tested on slices from a thoracic computed tomography (CT) scan. Finally, an automatic, non-uniform scheme is presented, and its registration performance is compared to the conventional uniform methods.

# 1. Introduction

## 1.1 Background and Significance

Over the last two decades, the use of advanced diagnostic imaging techniques has proliferated in the field of medicine, with evidence suggesting an acceleration of the trend [Smith-Bindman et al, 2012]. The ability to visualize 3D anatomical and functional information can improve the quality and accuracy of clinical treatments and interventions [Brock, 2003]. To fully exploit the additional information provided by these images requires methods to analyze the digital data. While this can be time consuming for physicians, modern computational systems make it both possible and preferable to develop automatic methods for image analysis. One tool essential to accomplishing this goal is image registration, which provides a means of combining and comparing image datasets [Hill. 2001].

This thesis presents a novel approach to image registration in the context of radiation therapy. Specifically, it focuses on adapting the well-known B-spline registration algorithm to a more general mathematical formulation through the use of Non-uniform Rational B-splines (NURBS). The motivation for this generalization is described with regards to the inherent geometric characteristics of B-splines and the displacement typically observed in thoracic computed tomography (CT) images.

Image registration is the process of finding an optimal correspondence function mapping homologous points in an image pair [Brown, 1992]. Applications in the field of medicine include intra-subject, inter-subject, multi-modality, and multi-instance analysis, as well as motion detection and compensation. The on-going trend of incorporating multi-modality (e.g. fusing

data from CT and MRI) and multi-instance imaging in treatment planning for external beam radiotherapy has underlined the need for fast and accurate deformable image registration techniques. The majority of currently available commercial image registration methods provide rigid registration techniques, which are adequate when aligning anatomical sites in which not much motion is expected, such as the brain, but insufficient in general [Brock, 2010]. Because of this, non-rigid registration techniques, also referred to as deformable image registration (DIR), are the subject of on-going research and development [Sarrut, 2006].

The fusion of information obtained from multiple imaging modalities and multiple instances throughout the radiation therapy treatment process enables a more conformal dose delivery by reducing uncertainty in the tumor's extent, motion, and position [Jaffray et al, 2007]. The increased conformality of the dose delivery allows higher doses to the tumor and lower doses to the surrounding normal tissues, thereby reducing toxicities and increasing tumor control probability. In order for this information to be properly integrated, the individual image data sets must be registered within a common geometry. Deformable image registration accomplishes this by producing a voxel-by-voxel displacement vector field (DVF) that maps points in a source image to the homologous points in a target image. The DVF is then used to map or transfer information (such as doses or contours) from one image to another.

Deformable image registration models generally fall into two categories. The first class of techniques involves modeling the deformation as a physical process, solving partial differential equations with respect to some energy minimization or other constraint. Elastic registration, fluid registration, and other mechanical models are included in this class. These methods have the appealing property of being physically intuitive, but suffer from an assortment of drawbacks. Elastic registration, for example, is limited by the inability to adequately model

highly localized deformations [Rueckert R, 2001]. Furthermore, these techniques rely on *a priori* knowledge about the expected deformations, such as elasticity and viscosity parameters [Schreibmann et al, 2006]. The second class, often referred to as parametric techniques, uses a linear combination of basis functions to create a free-form DVF to describe the image deformations. However, the complete free-form DIR problem is ill-conditioned--there are more degrees of freedom (displacement vectors) than constraining data. To make the problem tractable requires a model that regularizes (smoothes and interpolates) the DVF. Unfortunately, smoothing also reduces sensitivity to locally sharp deformations. If these local features are poorly modeled by the DVF, then there will be potentially large local errors in dose mapping and contour propagation [Kashani et al 2008].

Kybic and Unser [Kybic and Unser, 2003] present an argument for cubic B-splines as the most adequate set of basis functions based on their reduced interdependency between coefficients, local support, and calculation speed. B-spline models are effective for regularizing the deformation model and thus reducing the number of degrees of freedom. In the conventional implementation of B-spline DIR, a uniform spatial grid of free parameters (control points) is used to model the DVF. This grid regularizes the DVF uniformly throughout the image. However, the spatial pattern of anatomical deformation is often irregular, with regions of slowly varying deformation combined with regions of sharp or even discontinuous deformation. This is commonly seen in the thorax, where breathing movement (i.e., deformation) becomes much greater near the diaphragm and abruptly terminates at the ribs [D'Angelo et al, 2004; Xie et al, 2011; Demirovic et al, 2009; Wu et al, 2008]. A uniform control point grid over-smoothes the local irregularities, which reduces sensitivity to local details in the deformation. This causes large local errors in the registration, and can only be mitigated by the introduction of additional

control points. With a uniform grid, this requires an increase in the global number of control points. Global refinement to reduce large local errors is an inefficient use of free parameters, and increases computation time.

Multiple studies have addressed the issue of adapting uniform B-splines to non-uniform DVF features. One technique is to segment the anatomy into subvolumes or motion masks and register them independently; however this requires additional regularization to ensure that the registered subvolumes match at their boundaries. Refinement schemes based on radial basis functions (rather than B-spline basis functions) and hierarchical B-splines have also been introduced [Rohde et al, 2003; Forsey et al, 1995; Schnabel et al, 2001; Xie and Farin, 2004]. Recent work by Shusharina and Sharp has explored the possibility of adapting radial basis functions to local image features [Shusharina and Sharp, 2012]. These approaches produce non-uniform control point grids, but still rely on partitioning the registration into multiple smaller registrations.

It would be advantageous if truly non-uniform B-spline models in a single global geometry could be used for DIR, in the manner of the Non-Uniform Rational B-Splines (NURBS) that are routinely used to model complex surfaces in computer graphics [Piegel and Tiller, 1997]. This could be done by using a non-uniform control point grid that is adapted to local irregularities in the deformation. A non-uniform B-spline DIR method should be able to automatically arrange the control point grid so that more control points are present near locally sharp discontinuities, and fewer control points are used in comparatively smooth regions. The purpose of this project has been to develop such a method.



## 1.2 Hypothesis and Goals

In this dissertation, it is hypothesized that replacing B-splines with Non-Uniform Rational B-splines (NURBS) in the deformable image registration process will result in greater registration accuracy. The dissertation is divided into four specific aims.

- *Specific Aim 1: To develop an automatic NURBS fitting routine for known functions*
- *Specific Aim 2: To demonstrate that NURBS provide a better fit to known functions than B-Splines*
- *Specific Aim 3: To substitute NURBS for B-splines in the modeling of displacement vector fields.*
- *Specific Aim 4: To show that the incorporation of NURBS in deformable image registration algorithms leads to fewer registration errors than the B-splines algorithms*

The mathematics of conventional B-splines is presented in Chapter 2. Chapter 3 defines NURBS and develops methods for NURBS fitting to 1D functions that are consistent with the DIR paradigm. Chapter 4 extends these fitting methods to NURBS 2- and 3D tensor products. Chapter 5 incorporates the newly created NURBS fitting methods to 2- and 3D DIR. Finally, Chapter 6 provides an overview of the novel contributions within this work.

## 2. B-spline Mathematics

### 2.1 1D B-spline Functions

B-splines are piecewise continuous functions generated by an interpolation scheme predicated by the basis functions. The basis functions are defined such that they span the vector space of all piecewise polynomial functions of a desired degree and continuity [Piegl and Tiller, 1997]. The first recursive definition for numerical calculation of the B-spline basis functions was developed by Cox and de Boor [Rogers, 2000]. Given  $n+1$  control points, each basis function  $b_i$  of degree  $p$  can be defined as:

$$b_{i,0}(x) = \begin{cases} 1 & \text{if } k_i \leq x < k_{i+1} \\ 0 & \text{otherwise} \end{cases} \quad (2.1)$$

$$b_{i,p}(x) = \frac{x - k_i}{k_{i+p} - k_i} b_{i,p-1}(x) + \frac{k_{i+p+1} - x}{k_{i+p+1} - k_{i+1}} b_{i+1,p-1}(x) \quad (2.2)$$

The  $p$ th-degree B-spline function is then defined:

$$u(x) = \sum_{i=0}^n c_i b_{i,p}(x) \quad (2.3)$$

where the coefficients  $c_i$  are the scalar valued control points, and the  $p$ th-degree B-spline basis functions are defined on a non-decreasing, uniform knot vector  $U$  of  $m+1$  knots. Distinct knot values  $k_i$  represent breakpoints such that the half-open interval between two knot values  $[k_i, k_{i+1})$  is called the  $i$ th knot span (or knot segment). Non-zero knot spans define individual polynomial segments. One can see from the recursion relation defined above that the chosen knot values

significantly impact the B-spline basis functions, and, by extension, the shape of the resultant B-spline.

The B-spline basis functions confer several properties onto B-spline curves and surfaces which make them desirable for geometrical modeling. The most notable of these is the property of local support. Given a B-spline curve of degree  $p$  and a knot vector  $U = \{k_0, \dots, k_m\}$ , the basis function  $b_{i,p}(x) = 0$  if  $x$  is outside the interval  $[k_i, k_{i+p+1})$ . Similarly, in any given knot span, at most  $p+1$  basis functions are nonzero. This means that changes to the resultant B-spline curve can be made locally without affecting the global shape. This is done through manipulation of the control points  $c_i$ , which are effectively turned “on” and “off” by the basis functions. The local support property of the basis functions is clearly observed by plotting their values with respect to  $x$ . Consider the first-degree basis function  $b_{2,1}(x)$ , shown in Figure 1. It is only non-zero on the interval  $[k_2, k_4)$ , which corresponds to the parameter values from 0.0 to 0.4. Figure 2 shows the non-zero basis functions for degree 2. Notice that higher degree basis functions are non-zero over a wider range of parameter values, and therefore have a larger area of influence.

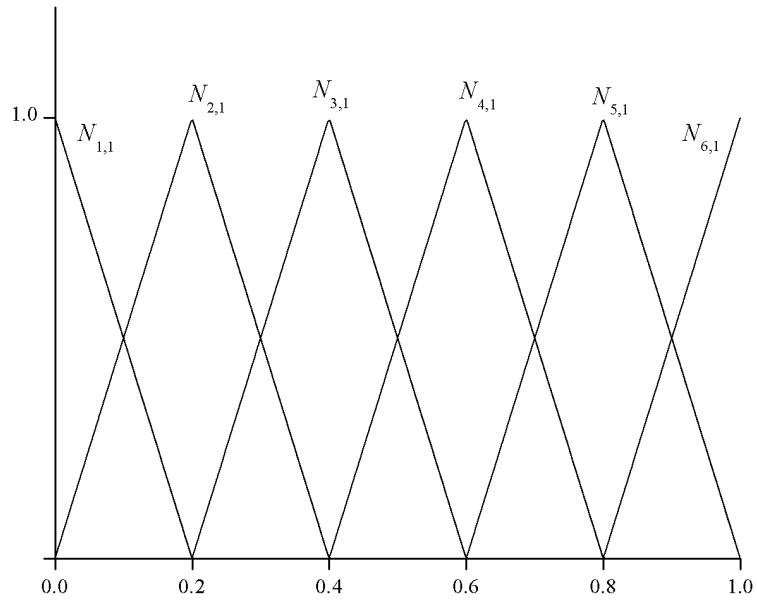


Figure 1: First-degree basis functions,  $U = \{0, 0, 0.2, 0.4, 0.6, 0.8, 1.0, 1.0\}$ .

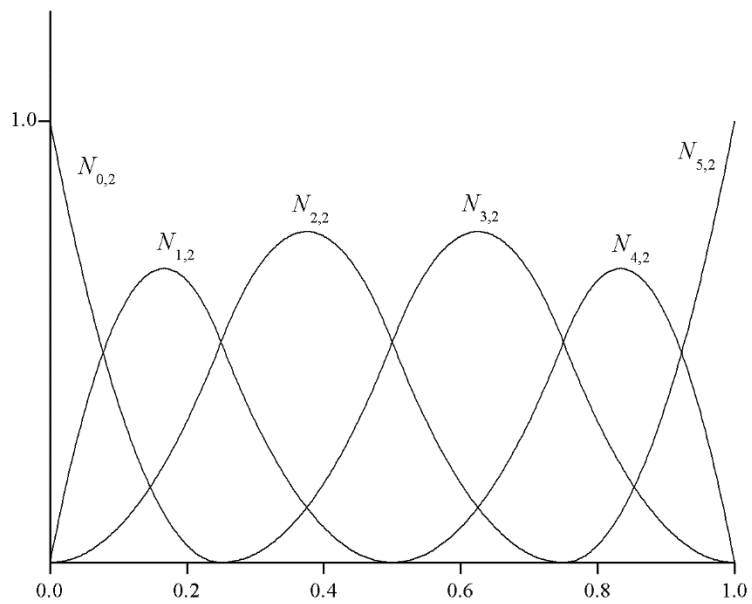


Figure 2: Quadratic basis functions,  $U = \{0, 0, 0, 0.2, 0.4, 0.6, 0.8, 1.0, 1.0, 1.0\}$ .

## 2.2 B-spline Tensors

Tensor products of two 1D B-spline functions with independent variables  $x$  and  $y$  can be used to define surfaces.

$$S(x, y) = \sum_{i=0}^n \sum_{j=0}^m c_{ij} b_{i,p}(x) b_{j,p}(y) \quad (2.3)$$

Similarly, a B-spline volume is defined as a tensor product of three 1D B-spline functions with independent variables  $x$ ,  $y$ , and  $z$ .

$$V(x, y, z) = \sum_{i=0}^n \sum_{j=0}^m \sum_{k=0}^l c_{ijk} b_{i,p}(x) b_{j,p}(y) b_{k,l} \quad (2.5)$$

The B-spline basis functions are again defined on non-decreasing, uniform knot vectors which are independent of one another. These knot vectors subdivide the B-spline into rectilinear grids which determine the local support of the control points  $c_{ijk}$ . The effect of the knot spacing is discussed in detail in Chapter 3.

## 2.3 Uses of B-splines in DIR

B-Splines have been used extensively in the field of computer aided geometric design and computer graphics [Piegel and Tiller, 1997]. More recently, they have also been employed to model deformations in non-rigid image registration [Reukert et al, 1999; Szeliski and Coughlan, 1995]. The image registration problem, as framed by Szeliski et al, consists of two images locally displaced by horizontal and vertical displacement fields,  $u_t(x, y)$  and  $v_t(x, y)$ , such that:

$$I_t(x + u_t, y + v_t) = I(x, y) \quad (2.6)$$

By minimizing the squared error over the set of pixels in the image, one arrives at the sum of squared differences formula:

$$E(\{u_i, v_i\}) = \sum_i [I_1(x_i + u_i, y_i + v_i) - I_0(x_i, y_i)]^2 \quad (2.7)$$

In 2D-2D B-spline registration models, the displacement vector field (DVF)  $u(x, y)$  and  $v(x, y)$  are represented as two-dimensional B-splines which lie on a grid of control points,  $c_{ij}$ . Optimization of the control point values to minimize the sum of squared differences formula proceeds via an iterative method. Choices for the iterative optimization typically include gradient methods such as steepest descent and the more complex Levenberg-Marquardt non-linear minimization technique [Press, 1992], but use of non-gradient methods such as the Nelder-Mead downhill simplex has also been reported [Nelder and Mead, 1965].

B-splines exhibit an advantage over other spline representations such as thin plate splines in that their basis functions have local support. This feature allows for local adjustments with no effect on global functional values. However, in order to fully realize this advantage, the control point grid spacing must be decreased. In uniform grid spacing, this can greatly increase the number of free parameters. In addition, uniform grid spacing may place control points in locations with no image intensity gradients. In these cases, the local support acts as a disadvantage because the control points are essentially unused free parameters.

The inherent smoothness of B-splines becomes problematic in DIR when attempting to model steep gradients and discontinuous motion in the DVF. This is observed, for example, with lung tissue sliding along the boundary with the lung wall. Because the basis functions of degree greater than 1 are defined to have continuous first derivatives across knot spans of non-zero

length, a DVF computed with the conventional representation of B-splines is constrained to be continuous everywhere. The spline knots are distributed at equal spacing throughout the registered volume, resulting in a uniform spatial resolution. To resolve locally sharp features in the DVF, one must distribute a large number of control points throughout the image. This is inefficient and partially defeats the B-spline advantage of local support, and thus has motivated research into B-spline configurations that can be adapted to local structure in the DVF.

An analogue of this limitation has been addressed in the field of computer graphics. A technique known as recursive subdivision, which is based on knot insertion, has been employed to generate arbitrary surfaces. Sederberg et al [Sederberg et al, 1998] have demonstrated the extra flexibility gained from the use of unequal knot intervals in a recursive subdivision scheme. Similarly, image registration techniques have incorporated subdivision via pre-segmentation of sites expected to exhibit discontinuities in the displacement vector field. However, this is an unattractive solution to the problem as it decreases automation. To solve this problem, this project proposes the use of NURBS, combined with a method to automatically assign the non-uniform control point grids.

### 3. Non-Uniform Rational B-splines

A more general approach to parametric representation of functions is provided by non-uniform rational B-splines (NURBS). Because NURBS are a generalization of B-splines, they carry forward nearly all the geometric and analytic properties of B-splines, while simultaneously providing additional degrees of freedom. A  $p$ th-degree NURBS curve is defined by

$$u(x) = \frac{\sum_{i=0}^n c_i b_{i,p}(x) w_i}{\sum_{i=0}^n b_{i,p}(x) w_i} \quad (2.8)$$

where the  $c_i$  and  $b_{i,p}(x)$  represent the control points, and  $p$ th-degree B-spline basis functions, respectively. The additional free parameters are introduced by the control point weights,  $w_i$ , and the non-uniform knot vector  $\bar{U}$  on which the  $b_{i,p}(x)$  are defined. The control point weights have the effect of multiplying the B-spline curve by a constant in the interval in which the associated control point is active. For example increasing  $w_3$  on a quadratic NURBS curve would increase the influence of the control point  $c_3$  in the interval  $[k_3, k_6)$ . This is shown in Figure 3. If all other weights are held to unity, changing a control point's weight  $w_i$  effectively scales the basis function  $b_{i,p}$  by the same factor (see Figure 4). Notice that because the sum of the B-spline basis functions for any value  $x$  must by definition equal 1 (i.e. the basis functions are defined as a partition of unity), increasing  $b_{i,p}$  decreases the remaining non-zero basis functions evaluated at  $x$ . The knots also alter the B-spline curve through changes to the basis functions. Multiple knots reduce the range for which a given basis function is non-zero. Knots of multiplicity equal to the degree of the B-spline reduce the non-zero knot span to a single parameter value, at which the basis function is equal to 1, as shown in Figure 5. This has the effect of creating a cusp or corner



in the resultant B-spline curve (see Figure 6), and therefore enables NURBS to model discontinuities.

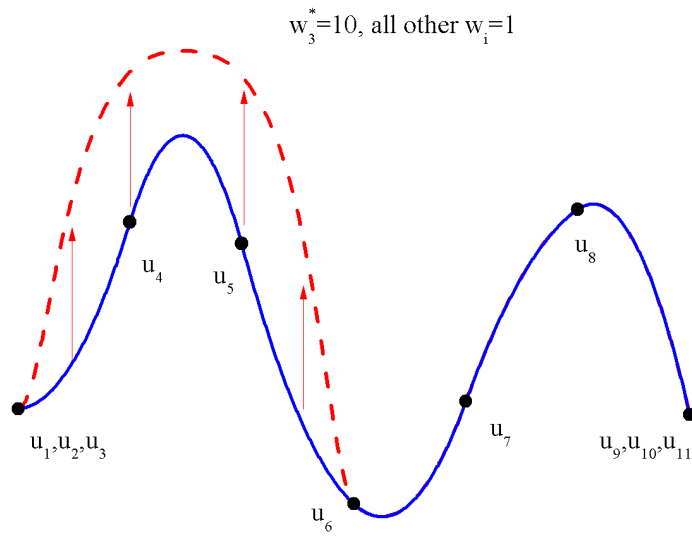


Figure 3: The effect of control points weights

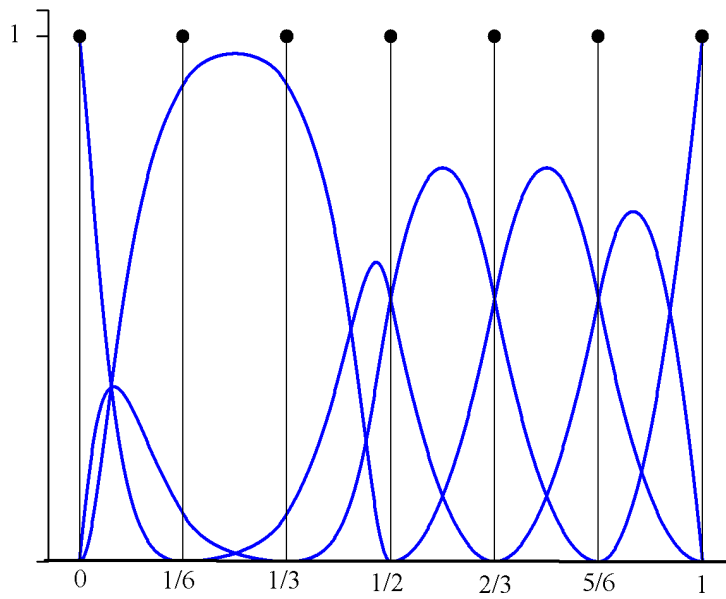


Figure 4: Rational basis functions with  $w_3=10$

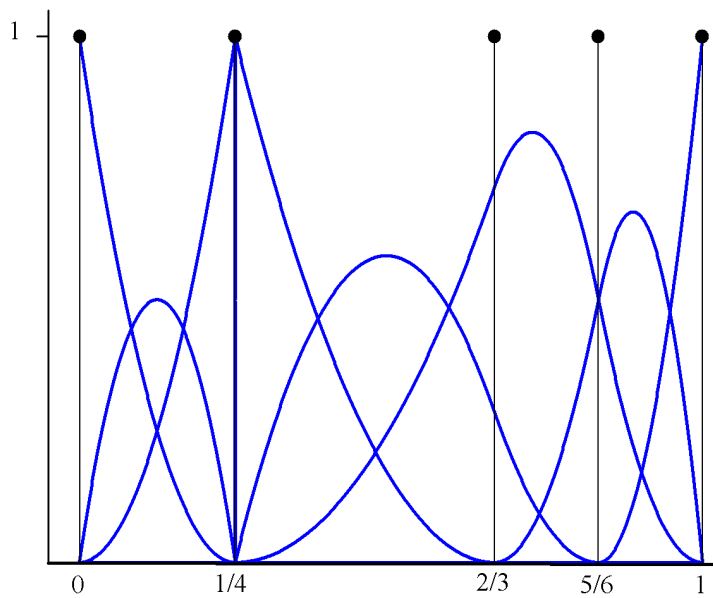


Figure 5: Basis functions with knot multiplicity equal to the degree at parameter value  $u = 1/4$

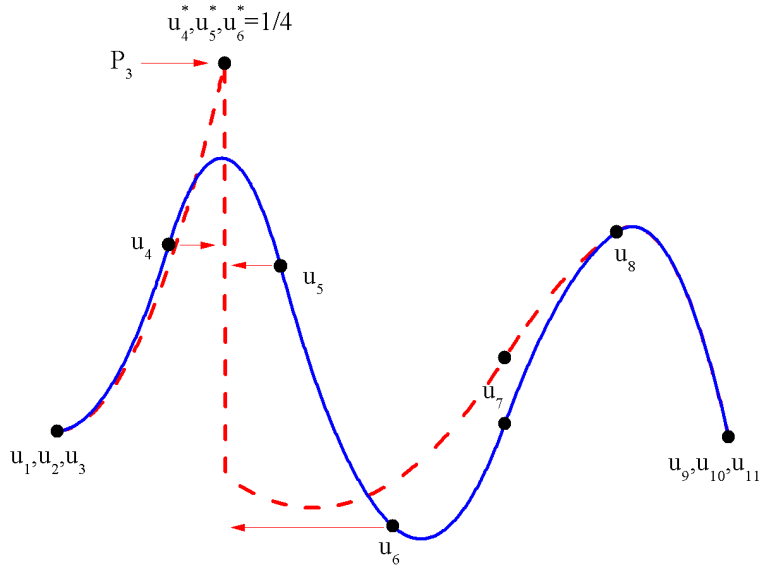
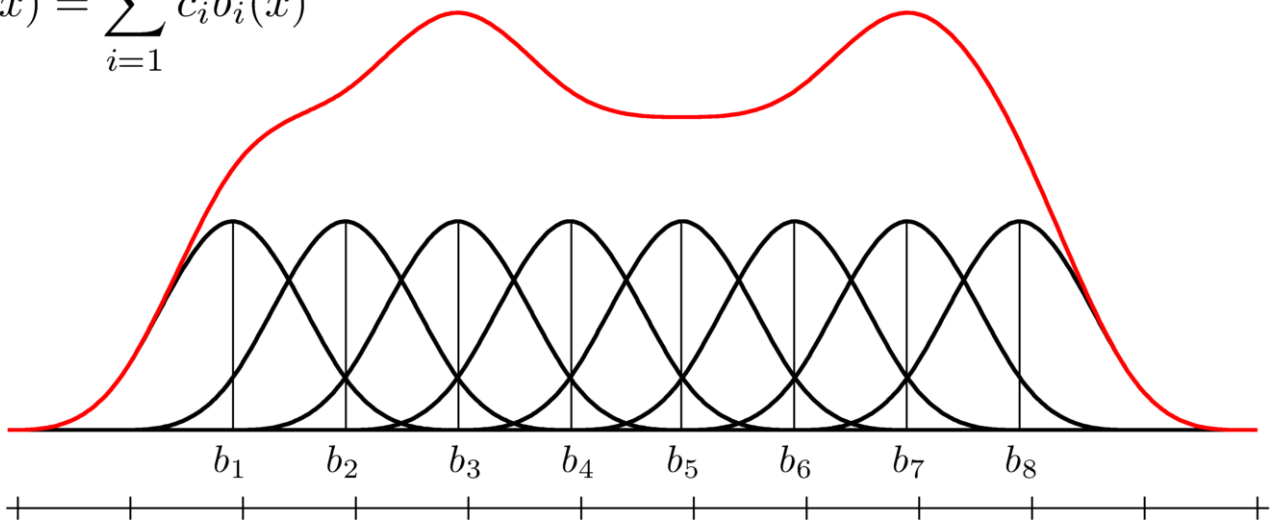


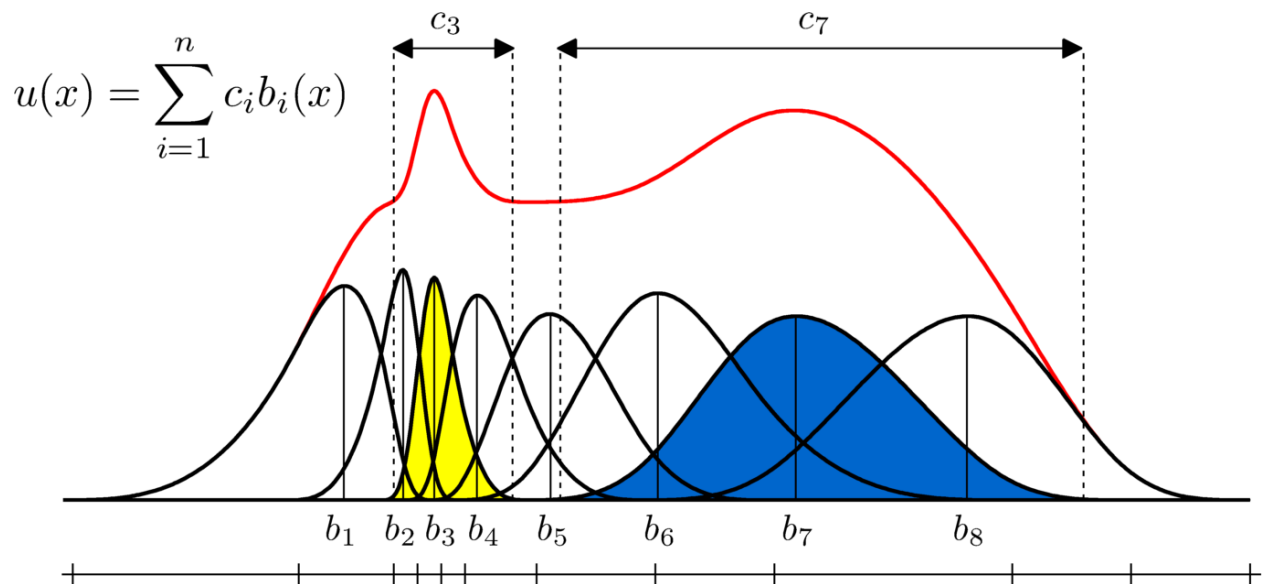
Figure 6: Knot multiplicities allow for discontinuities in the B-spline Curve

By definition, each basis function  $b_i$  is non-zero from knot  $i$  to knot  $i+p+1$ , where  $p$  is the degree of the B-spline. This confers local support to the B-spline coefficient  $c_i$  (i.e., the control point) associated with each  $b_i$ . The knot spacing therefore constrains the level of local curvature the B-spline is able to represent. If the knot spacing is decreased, the local derivative of the curve can be increased, and sharper features can be represented. This is conventionally accomplished by increasing the number of knots (and corresponding control points) uniformly across the parameter space. However, one could influence the local derivative of the curve by spacing the knots non-uniformly. Decreasing the knot spacing in a region reduces the range of the basis functions in that region and allows the representation of higher curvature. This is demonstrated in figures 7 and 8 by assigning equal values to  $c_3$  and  $c_7$  and noting the resultant difference in the curve  $u(x)$  in the parameter space corresponding to the associated basis functions  $b_3$  (yellow) and  $b_7$  (blue).

$$u(x) = \sum_{i=1}^n c_i b_i(x)$$



**Figure 7:** A uniform B-spline curve (shown in red) is the weighted sum of the B-spline basis functions (shown in black). The spatial extent (i.e., local support) of each B-spline basis function is determined by the spacing of the knots (denoted by tick marks).



**Figure 8:** A non-uniform B-spline curve (shown in red). The non-uniform spacing of knots can increase or decrease the local derivative of the curve.

### 3.1 NURBS Curve Fitting

NURBS are one of the most common fitting models in computer aided design because of their stability, flexibility, and local modification properties [Leal et al, 2010]. The applications of NURBS approximation range from reconstruction of digitized data from 3D measurements of manufactured parts for the purpose of reverse engineering to simplification (i.e. reduction of space required in computer memory) of pre-existing mathematical models. The shape of a NURBS 1D or 2D function is determined by the order, the number of control points and the values of the control points, the control point weights, and the knot vector. The NURBS fitting process seeks to find values for these parameters which minimize an error at discrete data points, typically measured by the  $l_2$  norm (least-squares fitting). As it is not known in advance how many control points are required to obtain an accurate fit, NURBS fitting is an iterative process, with each iteration adding one or more control points (or, equivalently, knots). A crucial step in all NURBS fitting techniques is the computation of control point values. This is discussed in Section 3.2.1. As is the case with the control point values, the knots and weights can be solved separately. Section 3.2.2 discusses techniques for optimizing the control point weights and Section 3.2.3 discusses optimization of the knot vector. In practice, there is little advantage to higher-order NURBS. While higher order polynomial terms add more flexibility, they also introduce more uncontrollably undulations. It has been observed in computer aided design (CAD) that cubic (degree 3) and quadratic (degree 2) NURBS offer the best trade-off between flexibility and fairness. For these reasons, this work assumes NURBS with a fixed degree  $p = 3$  unless otherwise stated.

### 3.1.1 B-spline Fitting of 1D Functions

Before going into the details of NURBS fitting, it is useful to first discuss the simpler case of B-spline fitting. B-spline fitting is the process of determining control points  $c_{ij}$  which minimize the sum of the squared distances at each point  $x$  in a function  $f(x)$ . In computer-aided design, the control points of a B-spline curve are vector-valued points in  $\mathbf{R}^2$ . This representation intuitively reflects the relationship between control points and curve shape: moving a control point in the 2D plane has the ability to stretch and twist a parameterized curve into loops and closed curves. For the purposes of this work, control points will be regarded as scalar coefficients to the basis functions. The mathematics remains the same.

Given a function  $f(x)$  evaluated at  $m+1$  data points,  $n$  unknown control points  $c_j$ , and assuming an order has been assigned and a knot vector has been computed, the least-squares error function between the resultant B-spline curve and the data points is

$$E(\mathbf{c}) = \frac{1}{2} \sum_{h=0}^m \left[ \sum_{j=0}^n c_j b_{j,p}(x_h) - f(x_h) \right]^2 \quad (3.1)$$

Equation 3.1 measures the squared distance between each point  $f(x_h)$  and the B-spline evaluated at  $x_h$ . The function  $E$  is quadratic in the control points and therefore has a parabolic shape whose vertex is a global minimum. This global minimum can be solved by setting the first-order partial derivatives to zero:

$$\frac{\partial E(\mathbf{c})}{\partial c_i} = \sum_{h=0}^m \left[ \sum_{j=0}^n c_j b_{j,p}(x_h) - f(x_h) \right] b_{i,p}(x_h) \quad (3.2)$$

$$= \sum_{h=0}^m \sum_{j=0}^n c_j b_{j,p}(x_h) b_{i,p}(x_h) - \sum_{h=0}^m b_{i,p}(x_h) f(x_h) \quad (3.3)$$

Equation 3.3 can be written in equivalent matrix notation as

$$A^T A \mathbf{c} - A^T \mathbf{f} = 0 \quad (3.4)$$

Minimization is therefore achieved by solving the linear system of equations above through direct inversion of the matrix  $A^T A$ :

$$\mathbf{c} = (A^T A)^{-1} A^T \mathbf{f} \quad (3.5)$$

The Schoenberg-Whitney condition [Piegel and Tiller, 1997] states that for a solution to be guaranteed existence and uniqueness, each evaluation point  $x$  must satisfy:

$$\begin{aligned} \forall k_h \in \mathbf{k}, 0 \leq j < n \\ k_h \leq x_h \leq k_{h+p+1} \end{aligned} \quad (3.6)$$

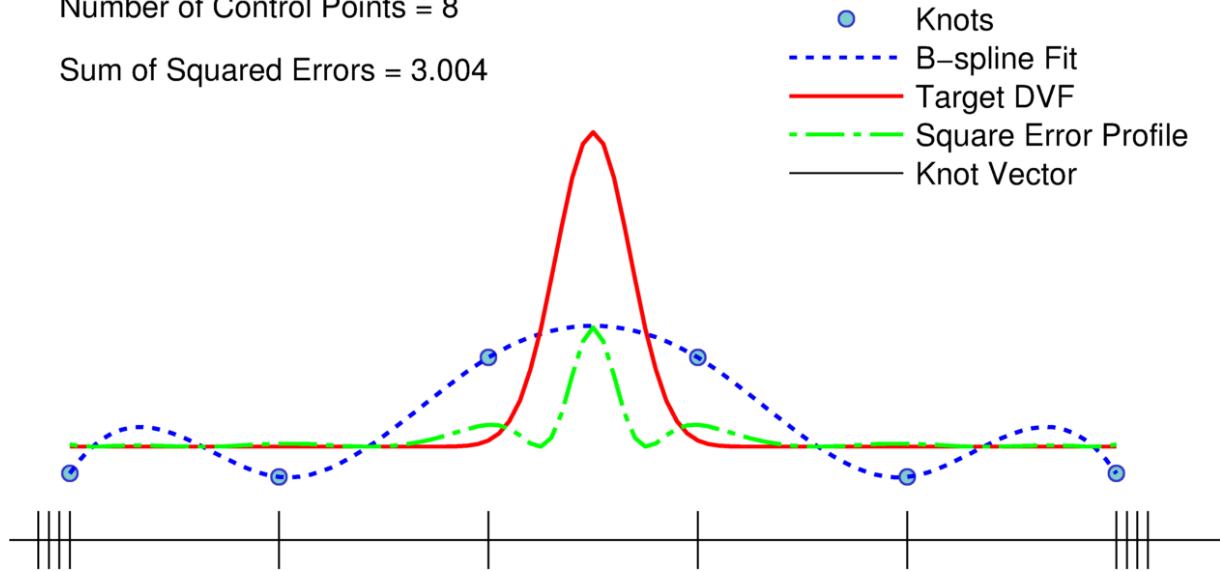
As Park [Park, 2004] notes, equation 3.5 provides reasonably good solutions when the number of parameter values  $m$  is equal to or much greater than the number of control points  $n$ . However, when the two are approximately equal, the Schoenberg-Whitney condition is not guaranteed, which results in unstable solutions sensitive to the parameterization. In addition, when  $m$  is less than  $n$ , the linear system of equations is under-specified. These situations will not be addressed because in deformable image registration the number of data points (voxels) is much larger than the number of control points used. Hence, this method produces an exact solution to least squares curve fitting with a uniform B-spline.

B-spline curves can be described as a weighted sum of B-spline basis functions. The shape of the basis functions is determined by the knot vector and the order of the spline. Therefore, shape control of a uniform B-spline curve is determined solely by the values assigned

to the uniformly distributed control points. B-splines are piecewise continuous polynomials partitioned by the knot vector. In order to increase local curvature, the length of the spline segments must be decreased. In the case of uniform B-splines, this can only be achieved by increasing the number of control points. Figures 9-11 demonstrate this point. In each figure, a uniform cubic B-spline is fitted to a 1D Gaussian. This synthetic DVF simulates a situation in which motion is observed locally and abruptly. It is an instructive example in that it clearly demonstrates the inefficient use of the B-spline free parameters (the control points), and the advantage in shape control gained through the use of non-uniform knot placement. As the number of control points is increased, the uniform spacing is decreased, which in turn decreases the range of local support and allows the B-spline to better represent local complexities and steep gradients. Notice, however, that with uniform spacing many of the internal knots are redundant. In this example, the knots located on the tails of the Gaussian would be more beneficial to the fitting problem if they instead supported smaller segments of the B-spline near the center of the Gaussian. Methods to achieve a more optimal distribution are discussed in the next section.

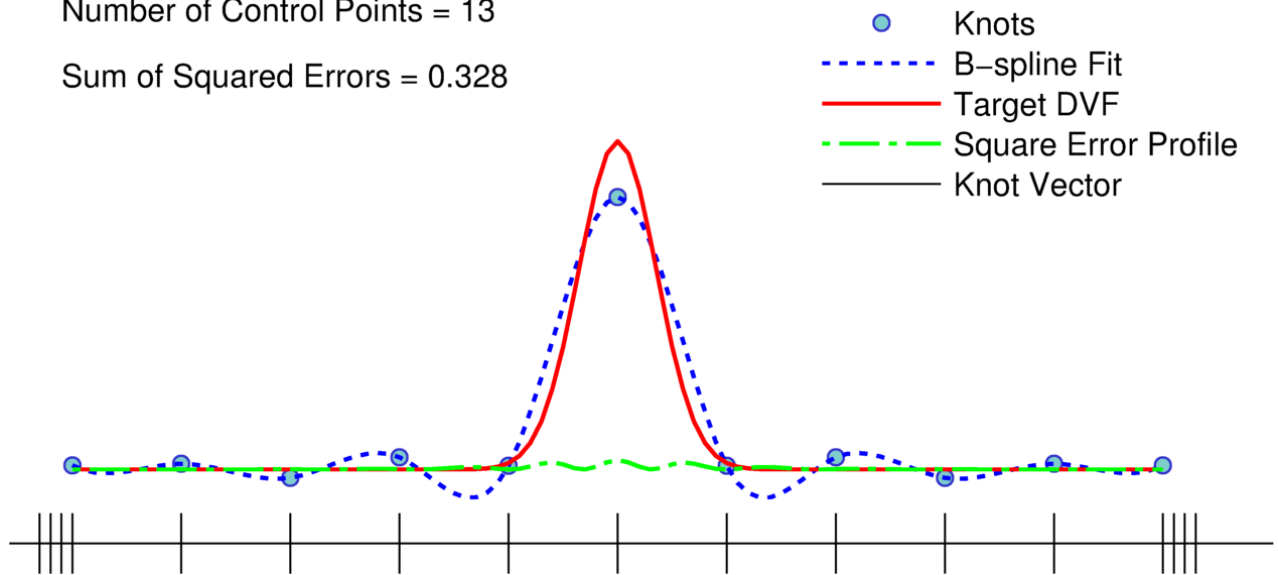


Number of Control Points = 8  
Sum of Squared Errors = 3.004

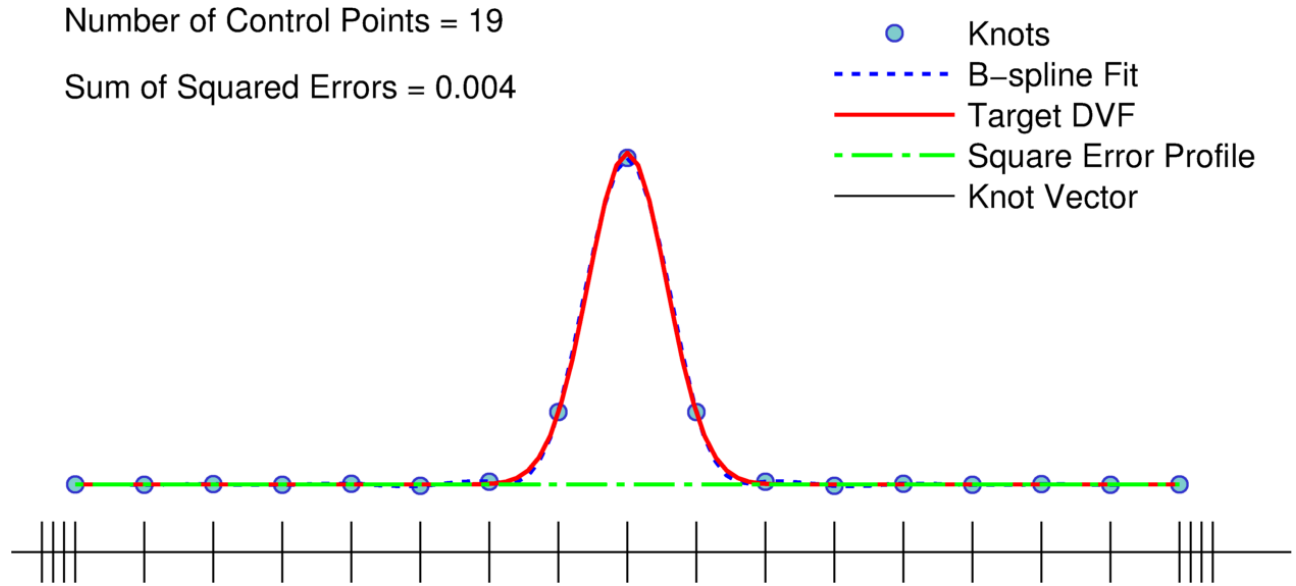


**Figure 9:** A uniform B-spline (shown in blue) with 8 control points fit to a 1D Gaussian (shown in red).

Number of Control Points = 13  
Sum of Squared Errors = 0.328



**Figure 10:** A uniform B-spline (shown in blue) with 13 control points fit to a 1D Gaussian (shown in red).



**Figure 11:** A uniform B-spline (shown in blue) with 19 control points fit to a 1D Gaussian (shown in red).

### 3.2 Optimizing Weights for Rational B-Spline Fitting

As rational functions, NURBS are algebraic fractions whose numerator and denominator are polynomials. A 1D NURBS is defined as

$$u(x) = \frac{\sum_{i=0}^n c_i b_{i,p}(x) w_i}{\sum_{i=0}^n b_{i,p}(x) w_i} \quad (3.7)$$

where the  $c_i$  and  $b_{i,p}(x)$  represent the control points, and  $p$ th-degree B-spline basis functions defined on the non-uniform knot vector  $\bar{U}$ . The normalization factor in the denominator makes NURBS fitting a nonlinear problem. Carlson presents a nonlinear least squares fitting of NURBS to data using the Gauss-Newton method [Carlson, 2009]. Alternatively, Ma and Kruth have developed a two-step algebraic solution which separates the weight calculations from the control point calculations [Ma and Kruth, 1998]. These methods are discussed in sections 3.2.1 and 3.2.2, respectively.

### 3.2.1 Weight Optimization using Gradient Descent Method

There are many well-known gradient-based algorithms for solving non-linear least squares. Some of the more commonly reported methods employed in NURBS fitting include Levenberg-Marquardt (LM), Broyden-Fletcher-Goldfarb-Shanno (BFGS), and Gauss-Newton (GN). Gradient-based methods are popular in B-spline fitting because the first derivatives are well-defined and can be determined without great computational burden. Minimizing the sum of squared distances between a set of points and the NURBS approximation leads to the objective function

$$\min \sum_{h=0}^m [u(x_h) - f(x_h)]^2 \quad (3.8)$$

where  $u(x)$  is defined as in equation 3.7 and  $f(x)$  is the target function.

In order to use a gradient-base optimization for NURBS curves, it is necessary to compute the derivatives of equation 3.7 with respect to the control points  $P$  and weights  $w$ . The partial derivative of the 1D NURBS  $u(x)$  with respect to the control point  $c_i$  is simply the rational basis

$$\frac{\partial u(x)}{\partial c_i} = R_i(x) = \frac{b_{i,p}(x)w_i}{\sum_{j=0}^n b_{j,p}(x)w_j} \quad (3.9)$$

The partial derivative with respect to  $w_i$  is slightly more involved because the term appears in both the numerator and denominator. It is simplest to separate the terms, find their derivatives, and re-combine. The numerator and denominator written as functions of  $w_i$  are

$$N(\mathbf{w}) = \sum_{i=0}^n b_{i,p}(x) w_i c_i \quad (3.10)$$

$$M(\mathbf{w}) = \sum_{i=0}^n b_{i,p}(x) w_i \quad (3.11)$$

And their derivatives are

$$N'(w_i) = b_{i,p}(x) c_i \quad (3.12)$$

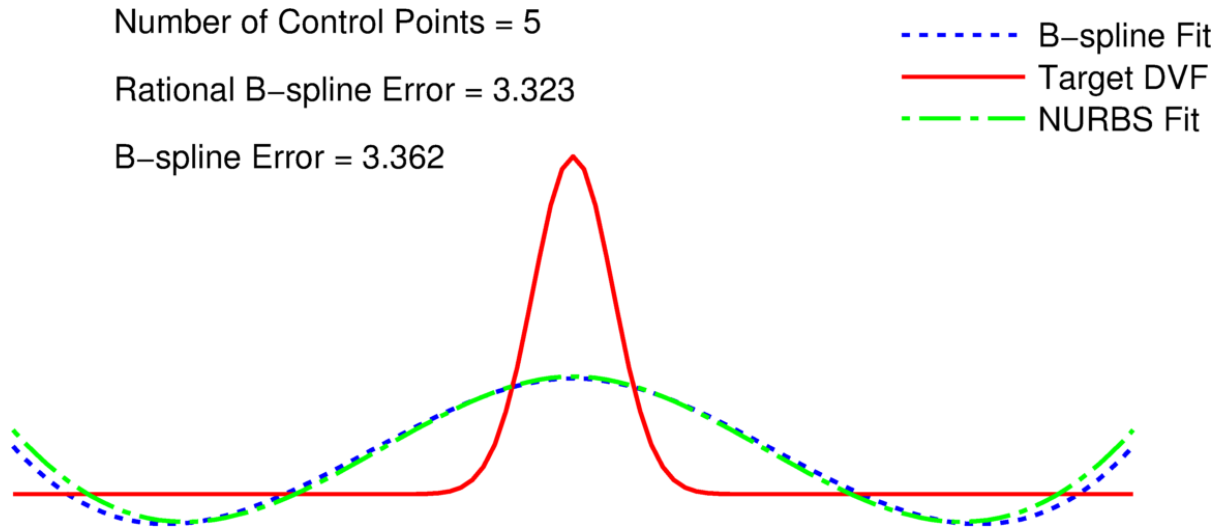
$$M'(w_i) = b_{i,p}(x) \quad (3.13)$$

Re-combing the terms, the derivative with respect to  $w_i$  is

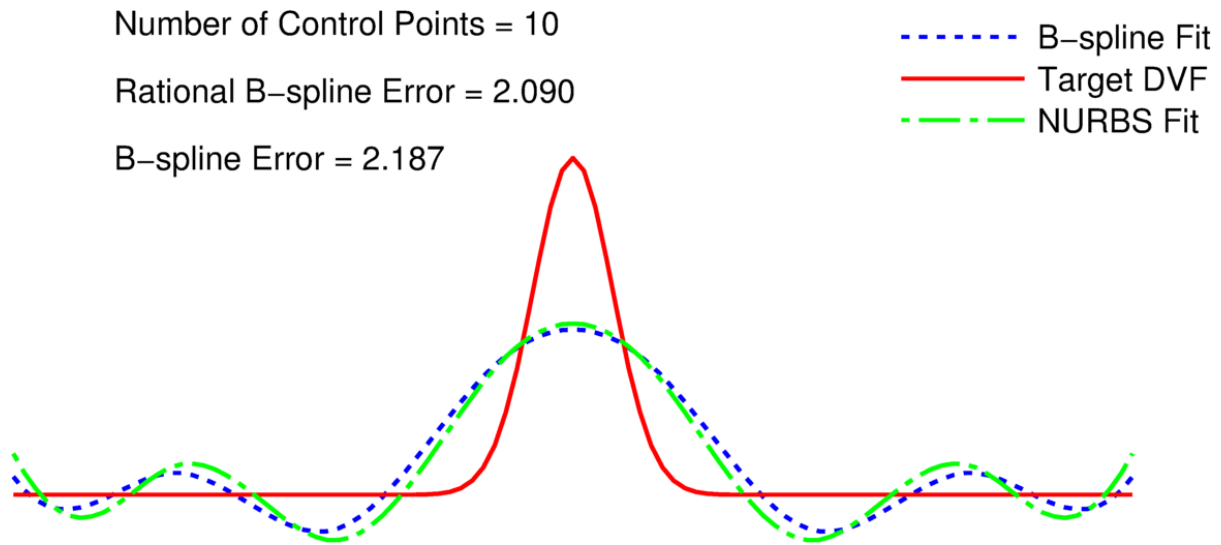
$$\frac{\partial u(x)}{\partial w_i} = \frac{M(\mathbf{w})N'(w_i) - M'(w_i)N(\mathbf{w})}{M(\mathbf{w})^2} \quad (3.14)$$

This paper uses an implementation of the BFGS algorithm.

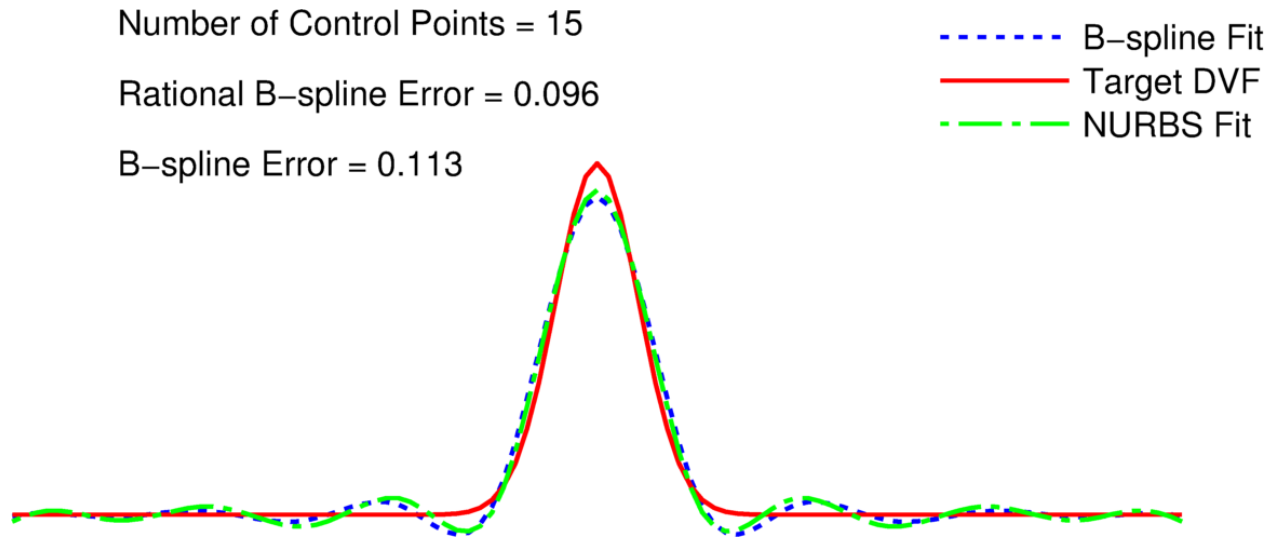
The performance of an optimization scheme is dependent upon a good choice of initial parameter values. One possible approach is to first fit the target curve with all the weights set to 1, which corresponds to a conventional B-spline fit. Optimization of the weights can then proceed with the control point values being held fixed. This approach was applied to fitting the 1D Gaussian for 5, 10, and 15 control points. Figures 12-14 show that adjusting the weights while leaving the control points values fixed to their conventional B-spline least squares solutions decreases the fitting error slightly.



**Figure 12:** Control point weights are adjusted via gradient descent. The 5 control point values obtained from solving equation 3.5 are held fixed.



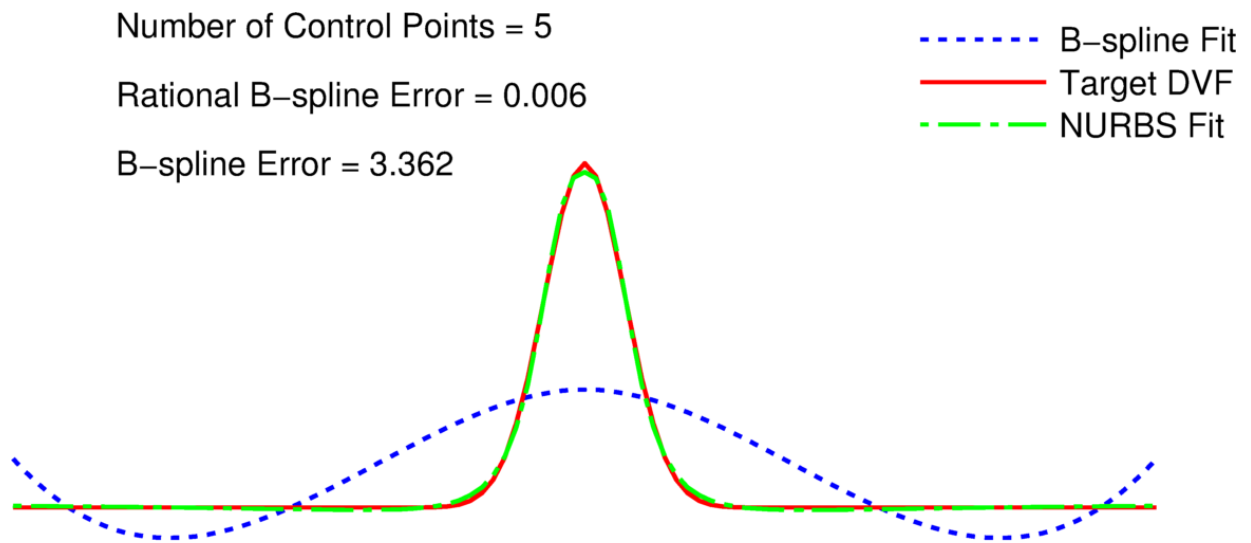
**Figure 13:** Control point weights are adjusted via gradient descent. The 10 control points values obtained from solving equation 3.5 are held fixed



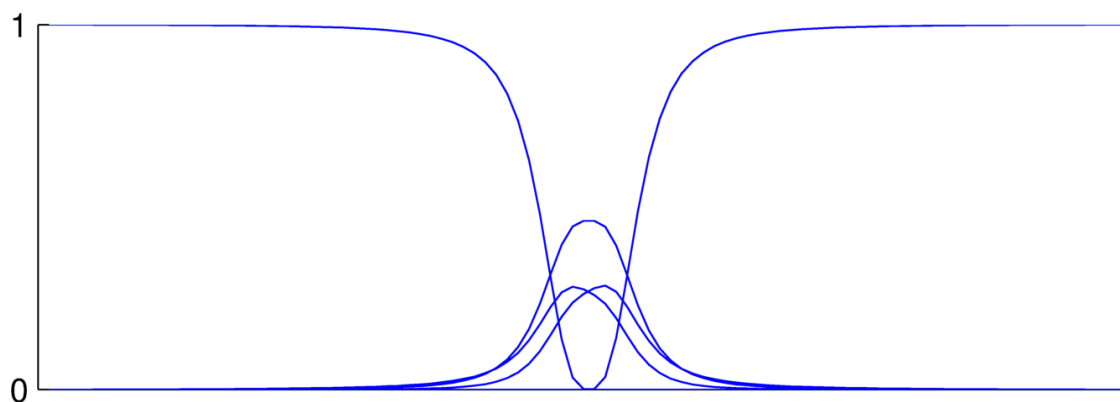
**Figure 14:** Control point weights are adjusted via gradient descent. The 15 control points values obtained from solving equation 3.5 are held fixed.

The small variation observed in the previous test can be attributed to the fact that the vector of the fixed control point values is a stationary point of the cost function in the case that the weights are all approximately equal. One can expect to see more variation when the control point values and their weights are adapted in tandem. Figure 15 confirms that suspicion. Initializing 5 control points to 0 and their weights to 1, the rational B-spline is able to fit the Gaussian target function to within 0.01. This result is surprising until one considers the shape of the of the rational basis functions, shown in Figure 16. Recall that the b-spline basis functions are defined so that they exhibit a partition of unity—that is, for any parameter value, the basis functions must add to 1. Assigning large weights to a particular basis function has the effect of increasing its influence in the knot segments for which it is non-zero. This is apparent in the first and last basis functions displayed in Figure 16. The combination of comparatively large weights with control points values set to zero allows for exact duplication of the Gaussian’s tails. Setting large weight values for the first and last basis functions also reduces the influence of the

remaining non-zero basis functions in the same interval. This squeezes the internal basis functions towards the middle, and the linear combination of these polynomials can easily model the Gaussian's curves. It turns out that this is a trivial test case for the rational B-spline, which was historically developed as an improvement over B-splines for representing conic sections and other analytic functions [Piegl and Tiller, 1997].



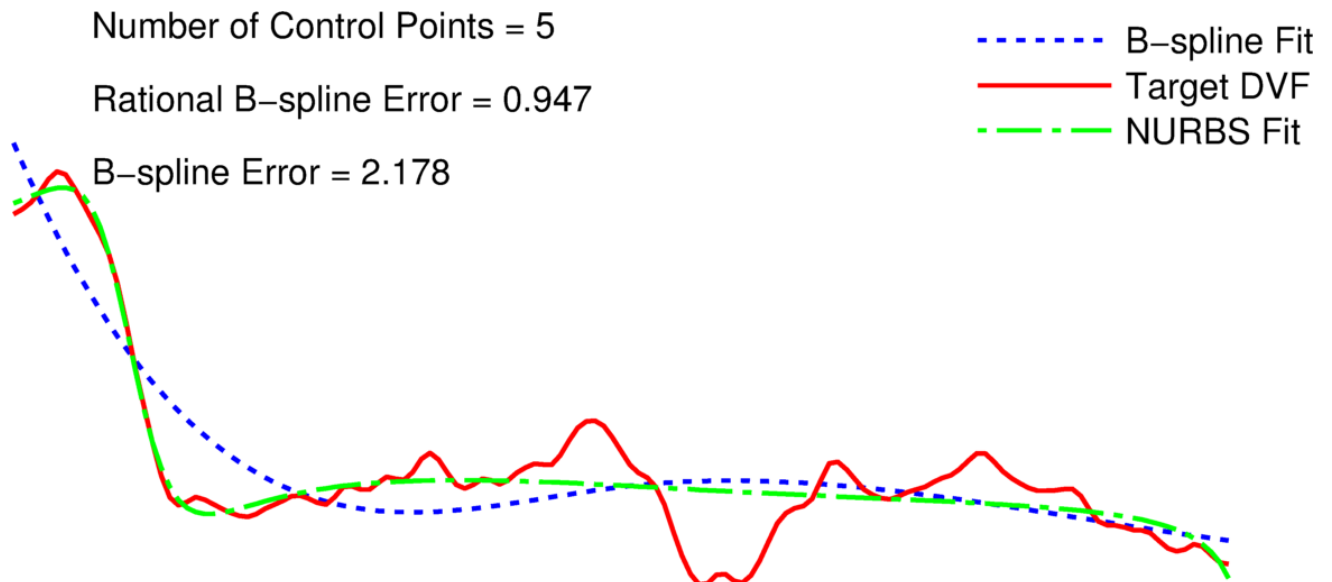
**Figure 15:** The rational B-spline can fit the 1D Gaussian with just five control points.



**Figure 16:** The rational B-spline can fit the 1D Gaussian with just five control points.

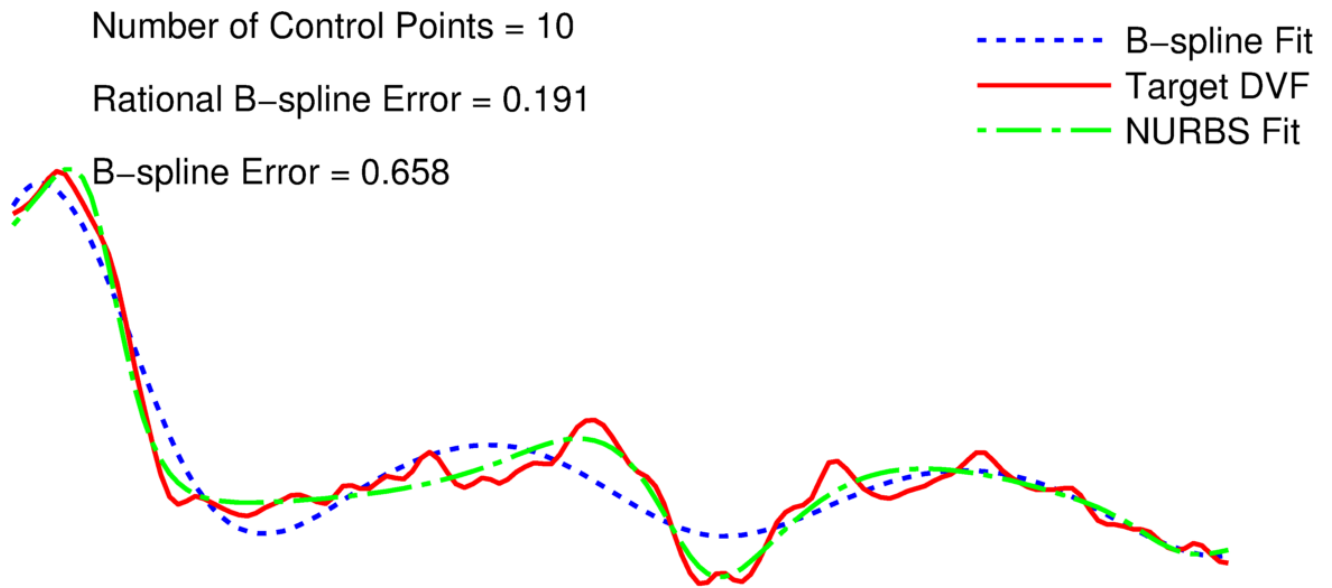
It is useful to test the algorithms on datasets that are more representative of real anatomical motion. For this purpose, two one-dimensional, single-component DVFs were generated from an existing 3D DVF computed by the Demon's algorithm. The DVF profiles were obtained from the POPI dataset, which contains two thoracic CT volume images and the DVF describing the anatomical motion between them. A coronal slice was selected, and the z-component of the motion was recorded for each voxel along a line in the x and z directions.

The non-analytic DVFs provide a better gauge of the possible comparative benefits of optimizing weighted control points in curve fitting. Figures 17-24 compare the rational and conventional B-spline fits to the 1D DVF profiles in the z-direction and x-directions, using 5, 10, and 15 control points, respectively.

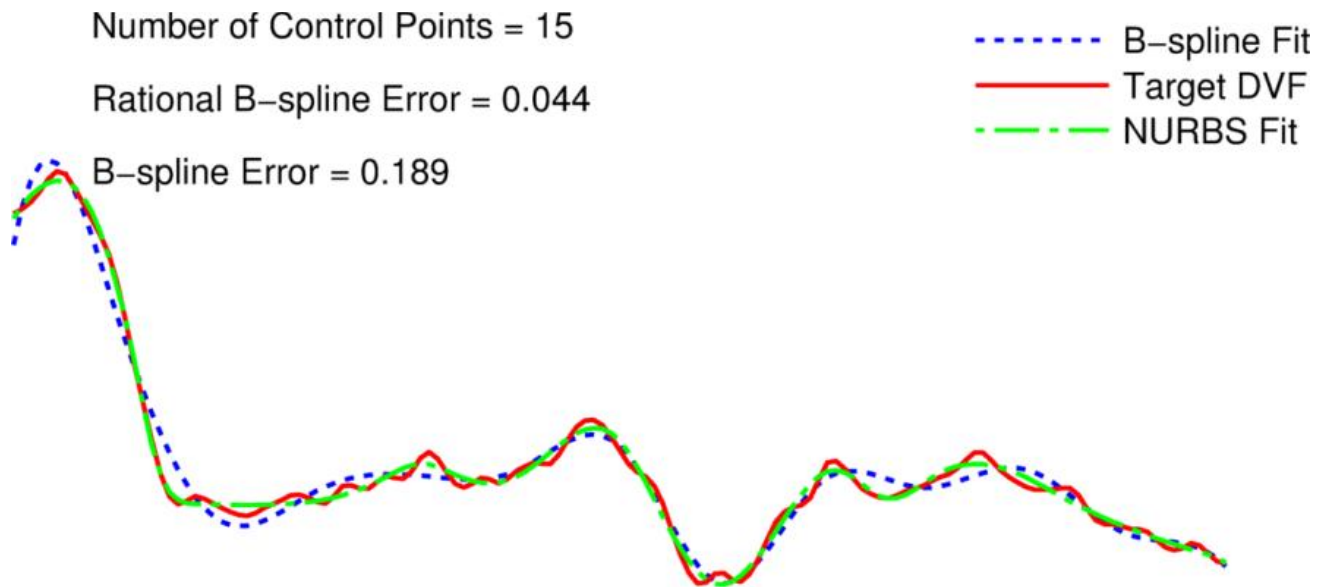


**Figure 17:** Fitting the z-direction 1D DVF with a rational B-spline using the gradient descent method to compute five control point values and their associated weights decreases the fitting error by more than a factor of two compared to the conventional B-spline fit with the same number of control points





**Figure 18:** The inclusion of adjustable weights again demonstrates improved fitting accuracy as compared to the conventional B-spline fit using 15 control points

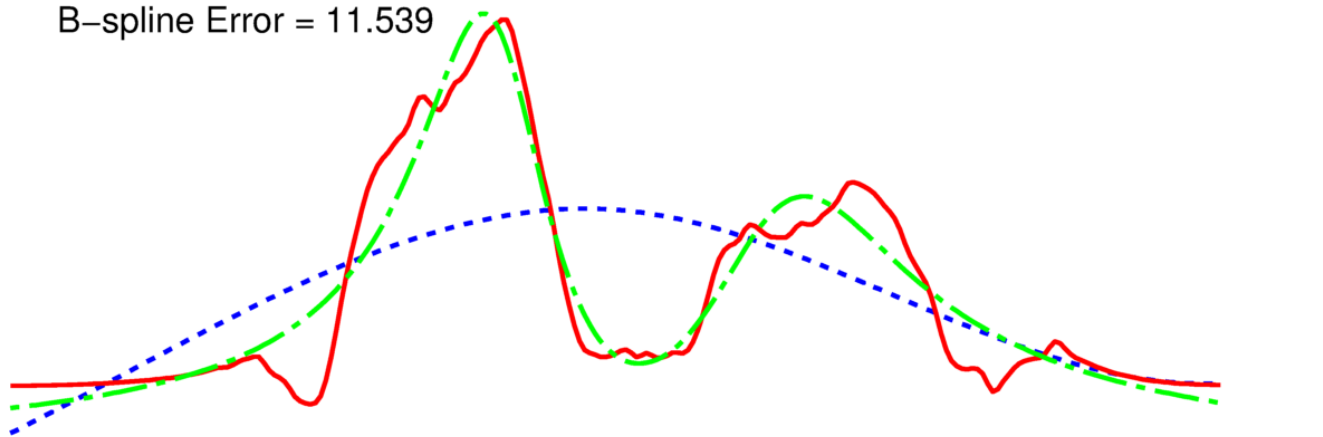


**Figure 19:** The inclusion of adjustable weights again demonstrates improved fitting accuracy as compared to the conventional B-spline fit using 15 control points

Number of Control Points = 5

Rational B-spline Error = 1.635

B-spline Error = 11.539

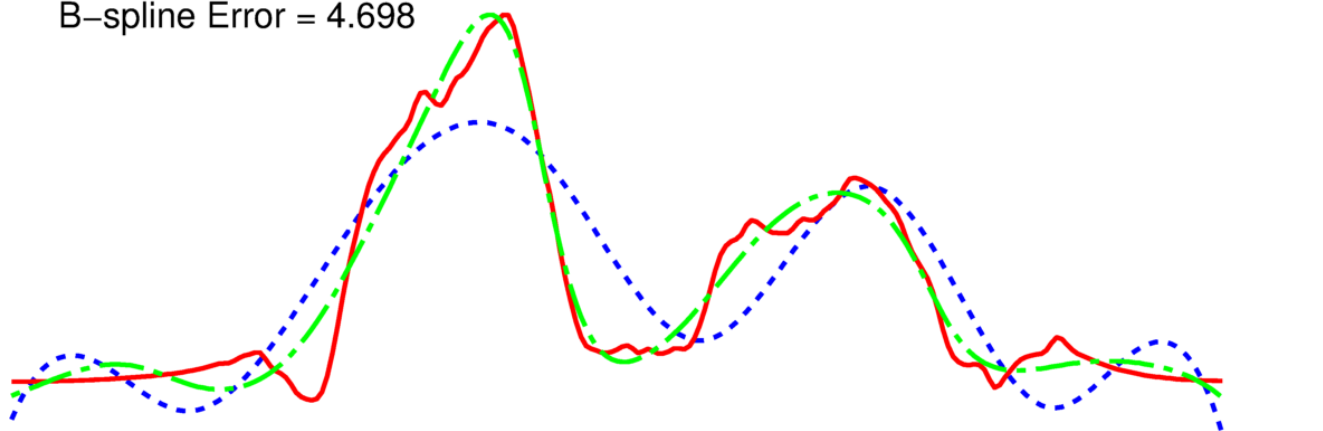


**Figure 20:** Fitting the x-direction 1D DVF with a rational B-spline using the gradient descent method to compute five control point values and their associated weights decreases the fitting error by more than a factor of five compared to the conventional B-spline fit with the same number of control points

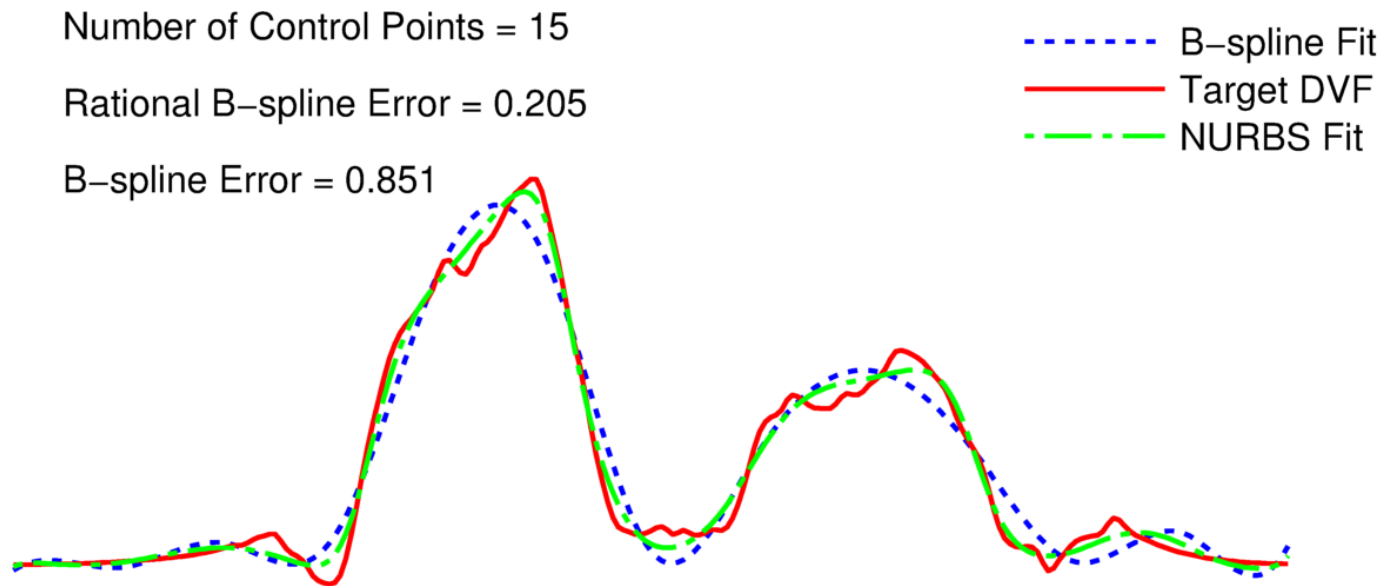
Number of Control Points = 10

Rational B-spline Error = 0.793

B-spline Error = 4.698

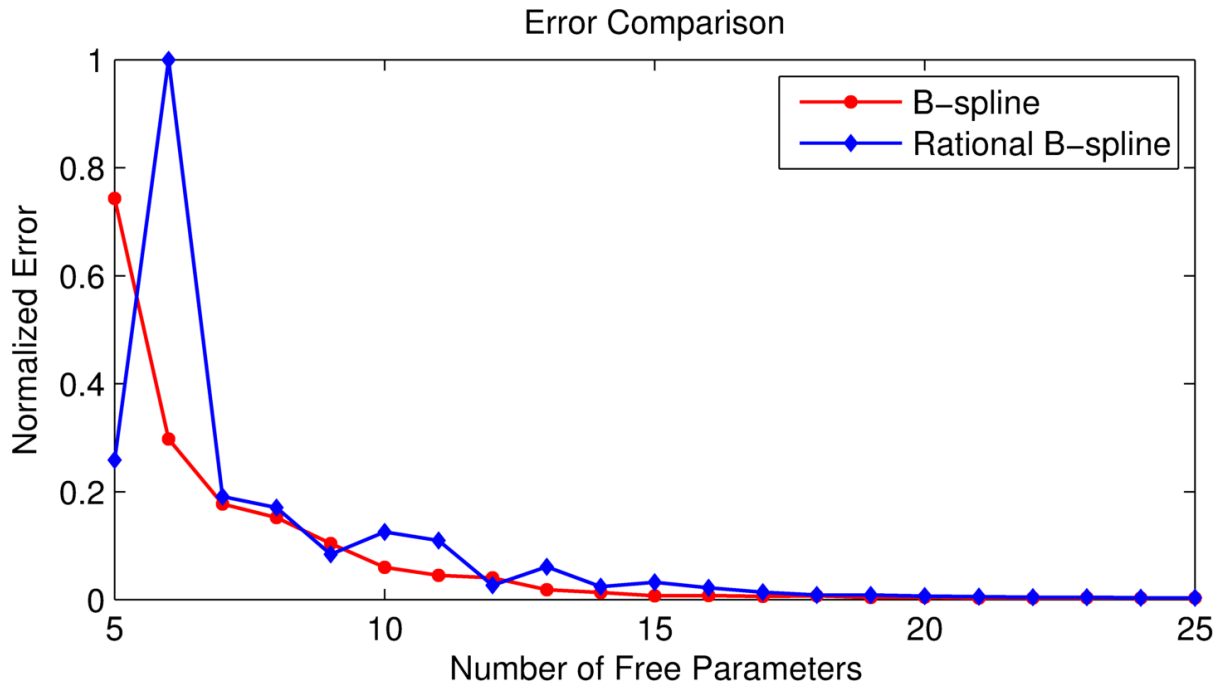


**Figure 21:** The inclusion of adjustable weights again demonstrates improved fitting accuracy as compared to the conventional B-spline fit using 10 control points

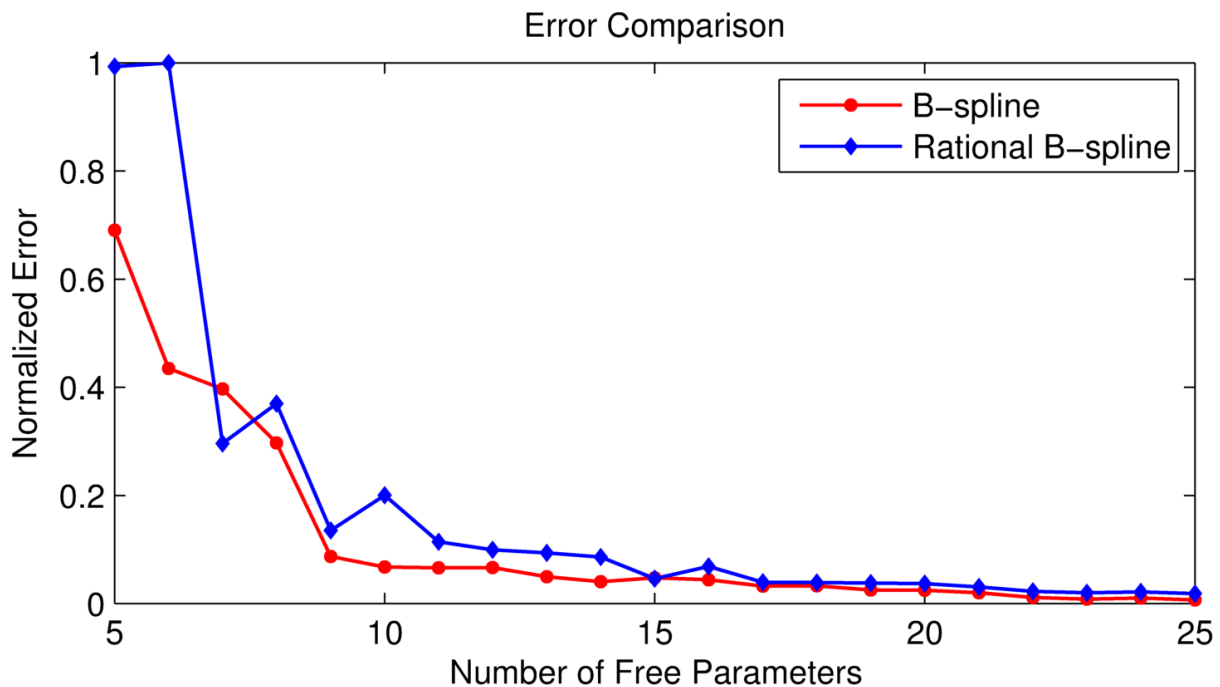


**Figure 22:** The inclusion of adjustable weights again demonstrates improved fitting accuracy as compared to the conventional B-spline fit using 15 control points

Fitting the DVF with a rational B-spline using the gradient descent method to compute both the control point values and their associated weights consistently produces a smaller error than conventional B-splines whose shape control is dictated by control values alone. However, it is worth noting that the rational B-splines weights computed via gradient descent contribute a computational cost similar to the control point values. For each additional control point, one must compute the gradient with respect to the control point and with respect to its weight. It is therefore more reasonable to compare the fitting performance of a rational B-spline with a B-spline containing twice as many control points. This puts the two methods on equal footing in terms of total number of free parameters to be optimized via gradient descent. When the two methods are compared this way, the advantage of using weighted control points disappears (see Fig 23, 24).



**Figure 23:** Comparison of errors of rational and conventional B-spline when fitted to the 1D DVF in the x direction. Conventional B-splines with twice as many control points exhibit similar fitting accuracy to rational B-splines with half the control points plus their associated weights (same number of total free parameters)



**Figure 24:** Comparison of errors of rational and conventional B-spline when fitted to the 1D DVF in the z direction. Conventional B-splines with twice as many control points exhibit similar fitting accuracy to rational B-splines with half the control points plus their associated weights (same number of total free parameters).

The inconsistent performance observed in the rational B-spline fitting (most notably the second data point in figure 24) is attributable to a poor choice of initial values for the free parameters. The parameter space has many local minima, so it's best to start optimization in the proximity of a solution good solution. As a method to avoid local minima, B-splines are often fit first with a coarse grid followed by subsequent refinement fitting in which an increasing number of control points are used. This strategy is referred to as multi-level B-spline fitting, and it is used widely in DIR [Xie and Farin, 2004]. At each step, the updated control point configuration is initialized by a least-squares fit to the previous B-spline, computed by a coarser resolution of control points. There is a well-known knot insertion algorithm [Piegl, 1997] for doing this with conventional B-splines, but not much attention is paid in the literature to a method which would simultaneously solve for the optimal weights. A more general method of finding an algebraic fit to an arbitrary curve or surface is presented in the next section.

### 3.2.2 Direct Solution for Control Point Weights

Suppose that one wishes to find the least squares NURBS fit to a 1D function  $f(x)$ . Additionally, suppose that the degree  $p$ , the number of control points  $n$ , and the complete knot sequence  $U$  are known. Then, equation 3.8 can be written in compact matrix form as

$$\mathbf{BP} = \bar{\mathbf{D}}\mathbf{B}\mathbf{w} \tag{3.15}$$

where  $\mathbf{P} = [w_1c_1, w_2c_2, \dots, w_nc_n]^T$  represents the scalar product of control points and their associated weights  $\mathbf{w} = [w_1, w_2, \dots, w_n]$  represents the weight values,  $\mathbf{D} = \text{diag}\{f(x_1), f(x_2), \dots, f(x_m)\}$  represents a diagonal matrix of the functional values, and

$$\mathbf{B} = \begin{bmatrix} b_1(x_1) & b_2(x_1) & b_3(x_1) & \cdots & b_n(x_1) \\ b_1(x_2) & b_2(x_2) & b_3(x_2) & \cdots & b_n(x_2) \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ b_1(x_m) & b_2(x_m) & b_3(x_m) & \cdots & b_n(x_m) \end{bmatrix}_{m \times n} \quad (3.16)$$

represents the matrix of basis function values. Equation (3.15) can also be written as

$$\begin{bmatrix} \mathbf{B} & -\bar{\mathbf{D}}\mathbf{B} \end{bmatrix}_{m \times 2n} \begin{bmatrix} \mathbf{P} \\ \mathbf{w} \end{bmatrix}_{2n \times 1} = [\mathbf{0}]_{m \times 1}. \quad (3.17)$$

Fitting the function is now reduced to solving the overdetermined system in equation 3.17 for  $\mathbf{P}$  and  $\mathbf{w}$ . To further simplify this computation, Ma and Kruth manipulate the first matrix to separate the weights from the control points. Multiplying equation 3.17 by the transpose of the first block matrix yields

$$\begin{bmatrix} \mathbf{B}^T \mathbf{B} & -\mathbf{B}^T \bar{\mathbf{D}}\mathbf{B} \\ -\mathbf{B}^T \bar{\mathbf{D}}\mathbf{B} & \mathbf{B}^T \bar{\mathbf{D}}^2 \mathbf{B} \end{bmatrix}_{2n \times 2n} \cdot \begin{bmatrix} \mathbf{P} \\ \mathbf{w} \end{bmatrix}_{2n \times 1} = [\mathbf{0}]_{2n \times 1}. \quad (3.18)$$

After eliminating the first element in the second row to make the matrix upper triangular, we have

$$\begin{bmatrix} \mathbf{B}^T \mathbf{B} & -\mathbf{B}^T \bar{\mathbf{D}}\mathbf{B} \\ \mathbf{0} & \mathbf{M} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{P} \\ \mathbf{w} \end{bmatrix} = [\mathbf{0}]_{2n \times 1} \quad (3.19)$$

where  $\mathbf{M}$  is the  $n \times n$  matrix

$$\mathbf{M} = \mathbf{B}^T \bar{\mathbf{D}}^2 \mathbf{B} - (\mathbf{B}^T \bar{\mathbf{D}}\mathbf{B})(\mathbf{B}^T \mathbf{B})^{-1}(\mathbf{B}^T \bar{\mathbf{D}}\mathbf{B}). \quad (3.20)$$

Now the weights are separated from the control points as

$$\mathbf{M} \cdot \mathbf{w} = [\mathbf{0}]_{n \times 1}, \quad (3.21)$$

and the weights and control points can be solved using equations 3.21 and 3.17, respectively.

The nullspace of  $\mathbf{M}$  might be larger than 1, in which case multiple solutions exist, or it could be of full rank, in which case no exact solutions exist. Ma and Kruth suggest least squares minimization for determining the weights  $\mathbf{w}$ , and offer two numerical methods for solving this equation. First, they suggest singular value decomposition (SVD) of  $\mathbf{M}$ , in which  $\mathbf{M}$  is factorized as

$$\mathbf{M} = \mathbf{QDP}^T, \quad (3.22)$$

where  $\mathbf{D}$  is a  $n \times n$  diagonal matrix whose elements are the singular values of  $\mathbf{M}$ . They are arranged in decreasing order, and the last non-negligible element  $d_r$  determines the rank of  $\mathbf{M}$ .

The general solutions of  $\mathbf{w}$  are given by linear combinations of the singular vectors  $\{\mathbf{p}_i\}_{r+1}^n$  contained in the matrix  $\mathbf{P}$ :

$$\mathbf{w} = \sum_{i=r+1}^n \alpha_i \cdot \mathbf{p}_i. \quad (3.23)$$

With the proper selection of the coefficients  $\alpha_i$ , a set of positive weights can be found.

Unfortunately, Ma and Kruth do not offer a method for choosing these values. Additionally, it is not clear what values should be considered “negligible” in the context of defining the rank of  $\mathbf{M}$ .

This ambiguity affects the goodness of fit even in simple one-dimensional cases. For example, fitting a one-dimensional Gaussian with 12 control points using the SVD method results in the following singular values in the diagonal matrix  $\mathbf{D}$ :

1.432094568968276
0.204206878890761
0.002452125410338
0.000513991288032
0.000000036673866
0.000000031527862
0
⋮
0

If we consider all non-zero elements to be non-negligible, the rank is 6. If we consider the 5<sup>th</sup> and 6<sup>th</sup> elements to be negligible, the rank is 4. This is not a trivial decision to make because the error of fit of the resulting rational B-spline varies largely for different rank values, as is illustrated in Table 1.

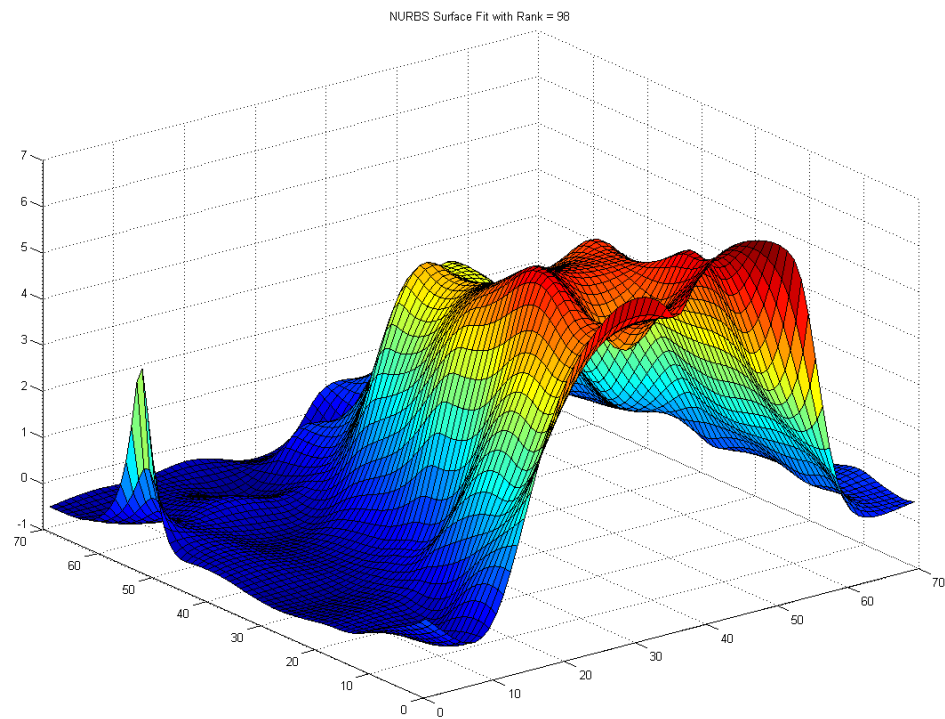
**Table 1:** Fitting error variation with respect to rank of matrix M.

Rank	Error of fit
3	1.223
4	.2553
5	.0267
6	.6243

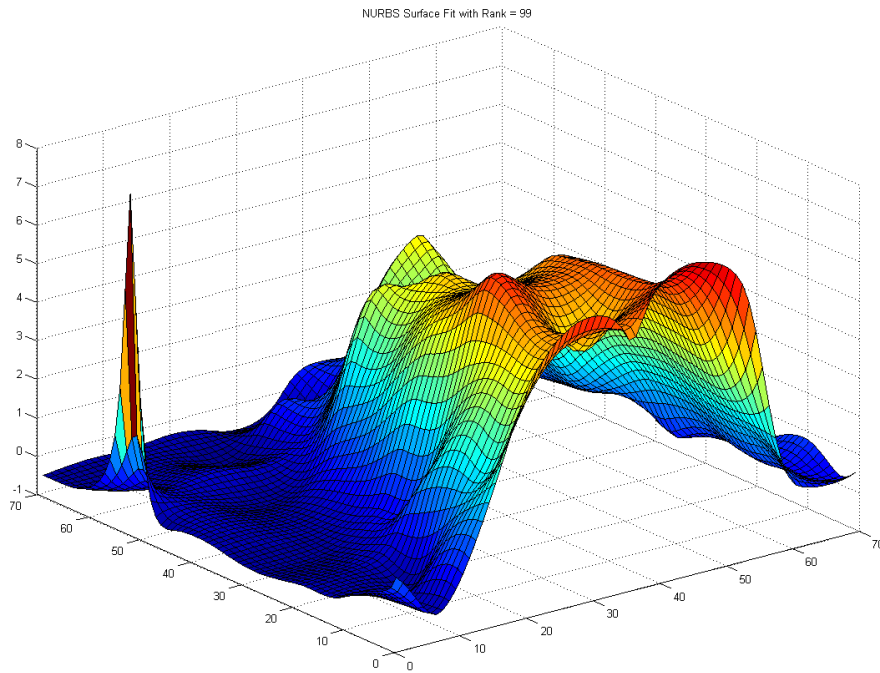
The rational B-spline function with the smallest error is computed when the rank is assigned to 5. This is surprising and unfortunate because it does little in the way of developing an algorithm to decide which values are negligible.



The ambiguous distinction between negligible and non-negligible elements in  $\mathbf{D}$  and the subsequent determination of the rank of  $\mathbf{M}$  is a pervasive problem in the SVD method. For example, in the more complex case of fitting surface data, a small change in rank can produce appreciable differences in the 2D rational B-spline tensor product fit. Figures 25 and 26 show the NURBS fits to DVF data using 100 control points. Height differences can be seen throughout, most notably in the far left peak. This performance instability makes the SVD method undesirable.



**Figure 25:** NURBS Surface fit with rank = 98.



**Figure 26:** NURBS surface fit with rank = 99.

To the best of the author's knowledge, the Ma and Kruth algebraic solution is the only solution available in the literature. The lack of a fast and reliable method to initialize weights in the hierarchical fitting scheme, combined with the absence of a performance gain per free parameter observed in fitting 1D DVF data via gradient descent, led the author to abandon the attempt to incorporate rational B-splines in DIR. Instead, the focus of the project shifted to the task of solving for the optimal non-uniform knot placement, which is discussed in the next section.

### 3.3 Non-uniform Knot Placement in B-spline Fitting

As mentioned previously, the B-spline basis functions are defined by a recursive formula consisting of polynomial contributions determined by the order of the B-spline and by the values of the knot vector (see Equation 2.1, 2.2). The basis functions are then combined linearly with their associated coefficients (control points) in order to form the B-spline function. Therefore, optimizing the knot vector offers another method for B-spline shape control. By preferentially placing knots in regions of steep gradients and local complexity, it is possible to generate B-spline fits with higher accuracy and fewer control points.

Non-uniform knot placement is a powerful B-spline shape control method. It allows for knots to be distributed sparsely in flat regions and densely in areas of greater complexity. This increases local spatial resolution without increasing the number of control points globally. In addition, multiple internal knots can be used to reduce continuity and enable representation of discontinuities (see figure 6). Despite these advantages, there is no consensus on how to optimize knot placement in NURBS curve approximation.

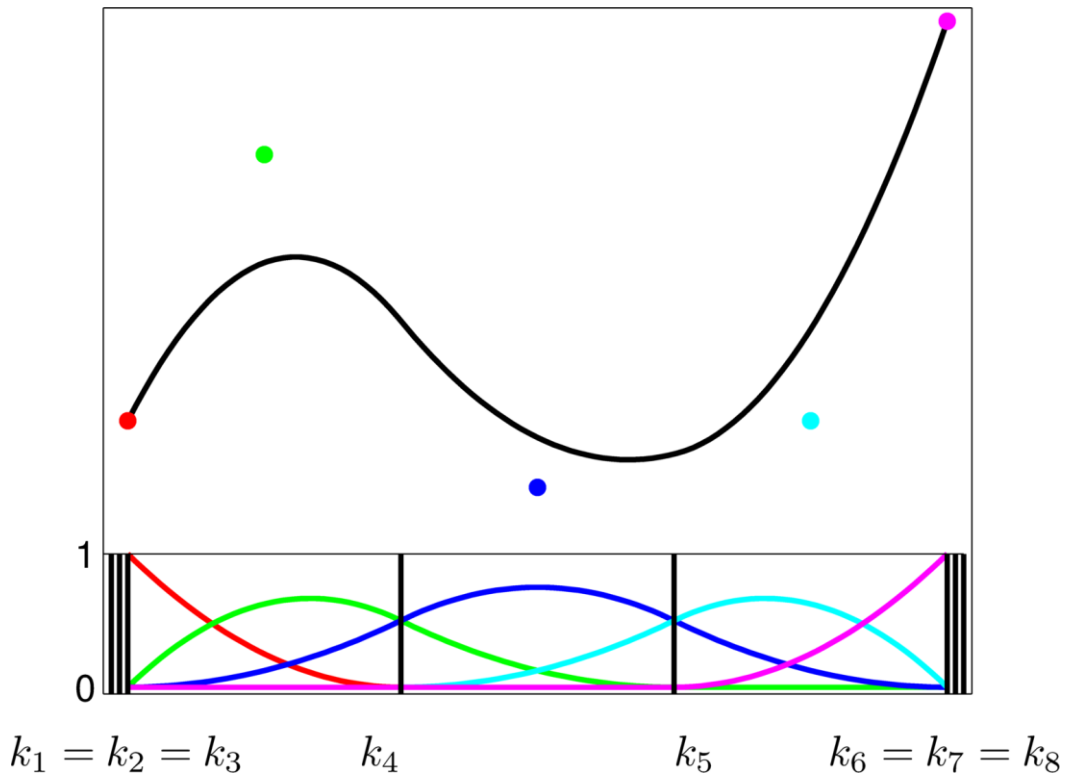
Knot nomenclature varies in the literature, but generally there are two varieties of knot vectors (periodic and open) with two sub-types (uniform and non-uniform). Open knot vectors differ from periodic knot vectors in that the first  $p+1$  knots are equal to some constant  $\alpha$  and the last  $p+1$  knots are equal to  $\beta$  ( $\alpha < \beta$ ). Such a knot configuration “clamps” the B-spline curve to the first and last control points (see figure 27). Periodic knot vectors, on the other hand, have no duplicate knots. All knot vectors in this work are assumed to be open. “Uniform” and “non-uniform” describe the spacing of the internal knots (Note: this is contrary to most of the B-spline DIR algorithms in the literature, which tend to prefer periodic knot vectors). Uniform knot

vectors have equally-spaced internal knots, but may still have duplicate knots at the endpoints.

Formally,

$$\begin{aligned} k_h &= 0 && \text{for } i = 0, 1, \dots, p \\ k_h &= h - p && \text{for } i = p + 1, p + 2, \dots, n \\ k_h &= n - p && \text{for } i = n + 1, n + 2, \dots, n + p \end{aligned} \tag{3.24}$$

Non-uniform knot vectors are free to have unevenly-spaced knots. Computational methods of the non-uniform knot vector are discussed in this section.



**Figure 27:** An open uniform quadratic B-spline. Below the curve is the basis functions defined on the knots  $k_i$ . Each basis function has an associated control point displayed as a circle of the same color. The knot positions determine the shape of the basis functions, which in turn determine the local support of the control points. The vertical lines labelled  $k_i$  represent the knots on this interval. Here, knots of multiplicity three are denoted by the bunched hash marks at either end of the interval. The knot multiplicity forces the basis function to be 1 and "clamps" the curve to its first and last control point.

As demonstrated in section 3.1.1, when the knots are fixed, the problem of least squares B-spline approximation is linear. In order to take advantage of this linearity, the NURBS approximation problem is formulated such that the knot vector is established first, followed by the solution of the control points. In naive implementations of NURBS curve fitting in CAD, the non-uniform knots are computed in tandem with the point parameterization such that there are nearly an equal number of parameters in each knot segment. This technique satisfies the Schoenberg-Whitney condition and ensures a solution, but does not explicitly seek to minimize the approximation error. However, the accuracy of the approximation is strongly dependent on the knot distribution. The constraints imposed on the knot optimization problem are of the form

$$k_h \leq k_{h+1} \quad \text{for } i = 0, \dots, n \quad (3.25)$$

These constraints appear innocuous, but the resulting optimization problem is non-linear and non-convex, with many local minima and stationary points [Beliakov, 2004].

Optimal knot placement for approximation to data is an on-going area of research in Computer Aided Design (CAD), Computer Aided Engineering (CAE), and applied mathematics. A wide variety of approaches have been taken to find a satisfactory solution to this problem. One method is to fit the data with a large number of knots and iteratively remove redundant knots. This, however, is not useful for the purpose of DIR as it starts with too many free parameters. Other proposed methods include the introduction of logarithmic transformations and penalty functions to transform the problem into an unconstrained form [Jupp, 1978; Dierckx, 1995], stochastic methods such as genetic algorithms [Goldenthal and Bercovier, 2003] and simulated annealing [Javidrad and Rohollah, 2013], and a deterministic optimization method which guarantees a global minimum has been reached [cite Beliakov, 2004]. However, these methods

have only been applied to B-spline curve approximation; their extension to higher dimensions has not been demonstrated. Furthermore, the computational burden of these mathematically complex optimization techniques may be prohibitive for use in DIR. A separate, more intuitive set of approaches based on considerations of local geometry is examined by Razdan [Razdan, 1999], Li et al [Li et al, 2005], and presented algorithmically by Park and Lee [Park and Lee, 2007]. This method computes a knot vector which is iteratively refined through the selection of “dominant” feature points on the curve. Initial dominant points as defined by Park and Lee include the end points and points corresponding to local curvature extrema of the target curve. Once initial dominant points are chosen, the knot vector is computed by averaging the parameter values of the dominant points, and a least-squares B-spline curve approximation is generated. Additional dominant points are selected iteratively to refine the curve approximation until an error tolerance has been met. The adaptive refinement paradigm followed by Park and Lee has motivated a similar method from Jacobson and Murphy [Jacobson and Murphy, 2011]. In the following sections, multiple non-uniform knot algorithms are implemented to fit 1D DVFs.

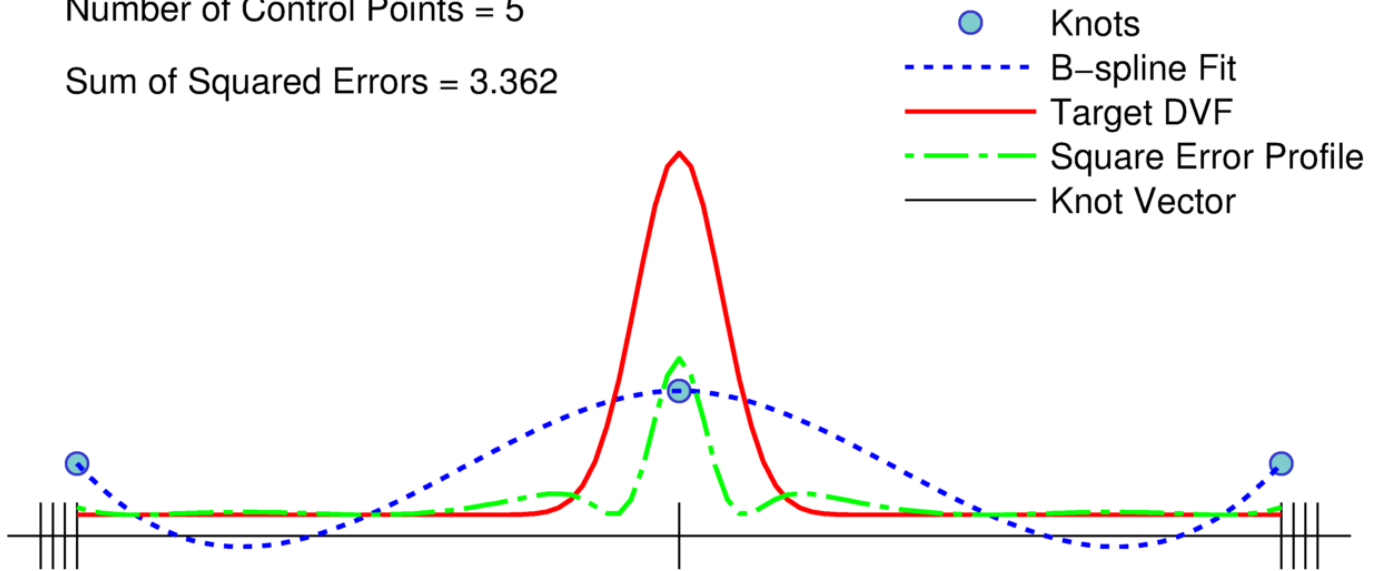
### **3.3.1 Simplex Method for Knot Optimization**

As Xie and Qin note, the derivation of the gradient of a NURBS curve with respect to the knots is complicated by the local support properties of NURBS [Xie and Qin, 2001]. For example, moving a knot past its adjacent neighbor will alter the span interval over which a control point is active and may result in discontinuities. To avoid this issue, a gradient-free optimization method, the Nelder-Mead Downhill Simplex algorithm is used. The cost function to be minimized is the same as before (the sum of the squared differences), and the free parameters are the  $n-(p+2)$

internal knot positions, where  $n$  is the number of control points and  $p$  is the degree of the B-spline, both of which are specified prior to optimization. For each function evaluation, it is necessary to solve equation 3.5 using the basis values defined by the new test point (vector of free knots). Because the Simplex method requires a large number of function evaluations, this approach to optimizing the knot placement becomes computationally prohibitive for large datasets and would not be suitable for implementation in the DIR process. However, for simple cases it reliably converges to a reasonable solution and serves as a good basis of comparison for the other non-uniform knot methods.

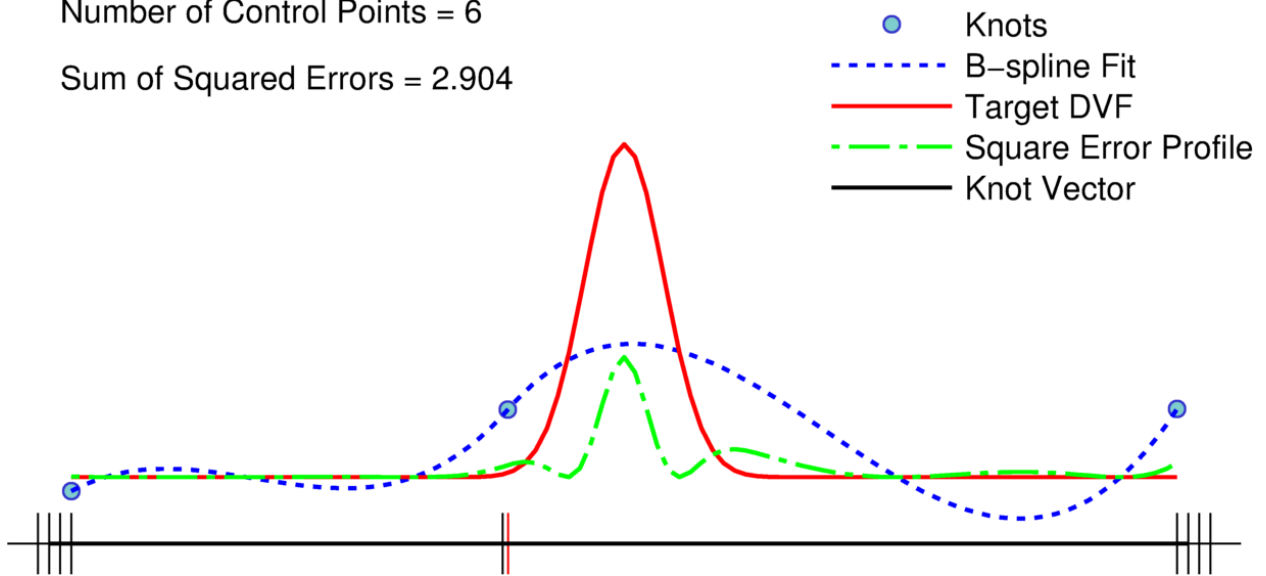
The optimal knot positions computed by the Simplex method were used as a gold standard for the novel knot placement techniques covered in the sections that follow. The challenge was to develop a set of heuristic rules which adapted the knot positions and closely followed the knot placement behavior observed in the solutions of the Simplex method. Figure 28 displays the least squares solution for a cubic B-spline with one free knot, which is a single knot located in the center of the Gaussian. This is no different from the uniform knot distribution with a single free knot. However, the solution with two free knots, shown in Figure 29 is surprising. One would reasonably expect that the optimal knot distribution for a symmetric target function would also be symmetric. Instead, it was observed that duplicate knots, asymmetrically located on either side of the Gaussian, yielded the least square fit. This non-intuitive result, which was confirmed through brute force by solving equation 3.5 for every valid combination of knot values, hints at the difficulty of developing a heuristic rule for knot distribution.

Number of Control Points = 5  
 Sum of Squared Errors = 3.362



**Figure 28:** Simplex solution for one free knot, which coincides with the uniform knot distribution.

Number of Control Points = 6  
 Sum of Squared Errors = 2.904

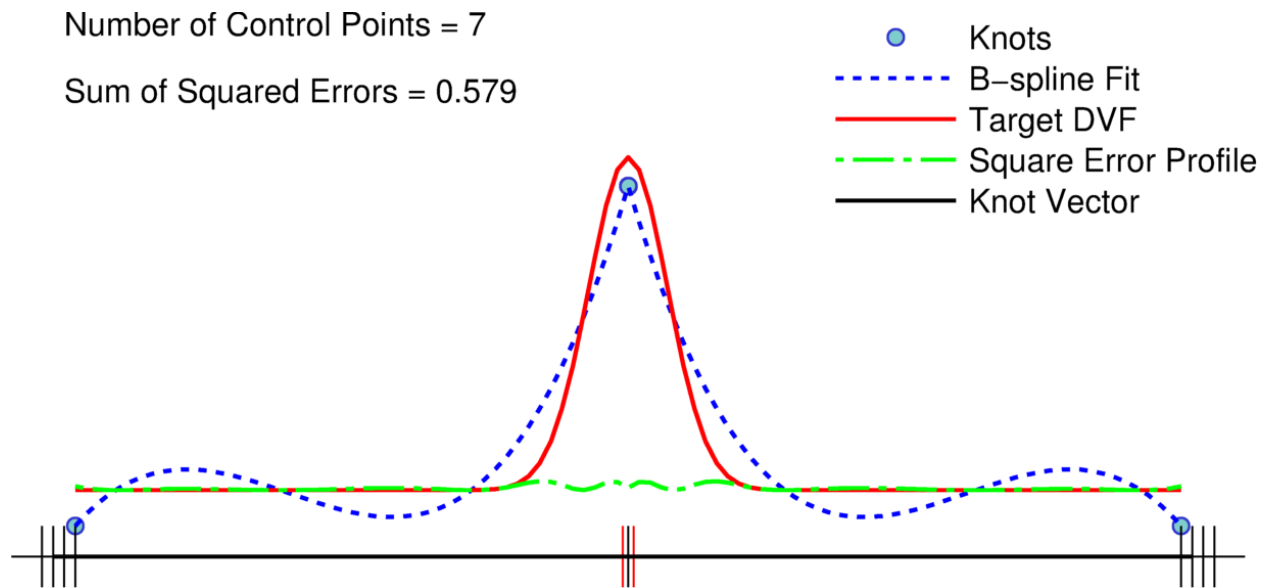


**Figure 29:** Simplex solution for two free knots, which exhibits advantage in asymmetric distributions even on symmetric target functions. The internal duplicate knot is indicated by the red hash mark.

The best-fit cubic B-splines for the number of free knots ranging from 3 to 6 are displayed in figures 30 through 32. Repeated knot values and asymmetric knot distributions continue to



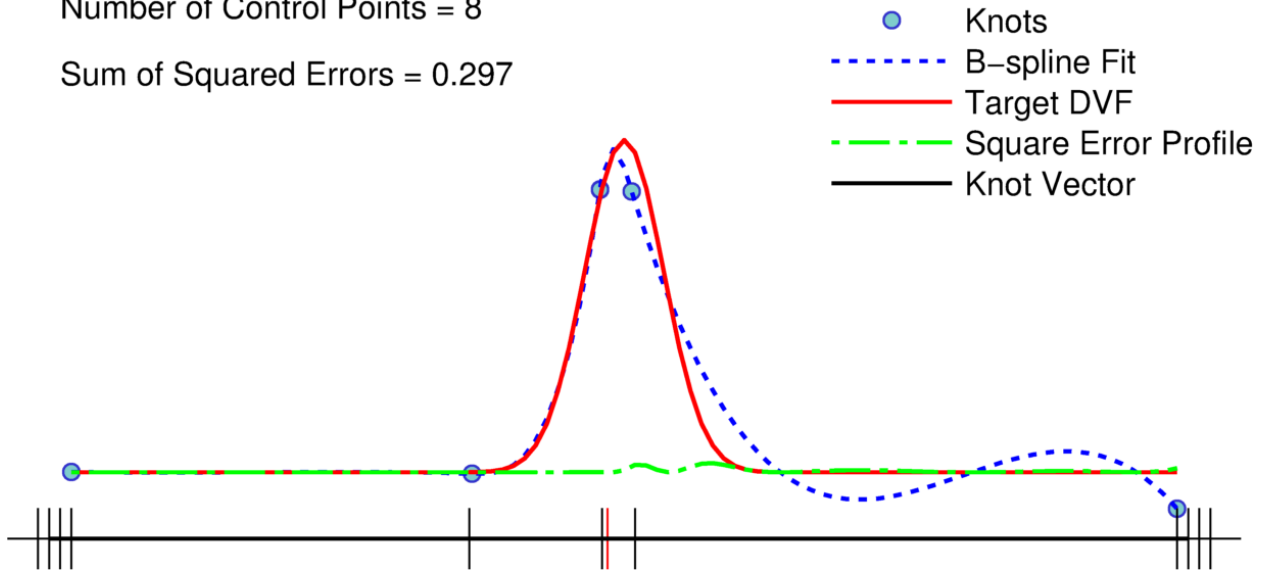
produce the best fit curves. With 6 free knots, the cubic B-spline is able to reproduce the Gaussian curve within a tolerance of 0.1% of the initial sum of squared errors. This result becomes the metric by which the proposed knot placement methods are evaluated.



**Figure 30:** The solution for three free knots is repeated knot values at the central parameter value. Repeated internal knots are indicated by red hash marks on either side of the true knot value in black Knots values with multiplicity equal to the degree of the B-spline can produce cusps and discontinuities

Number of Control Points = 8

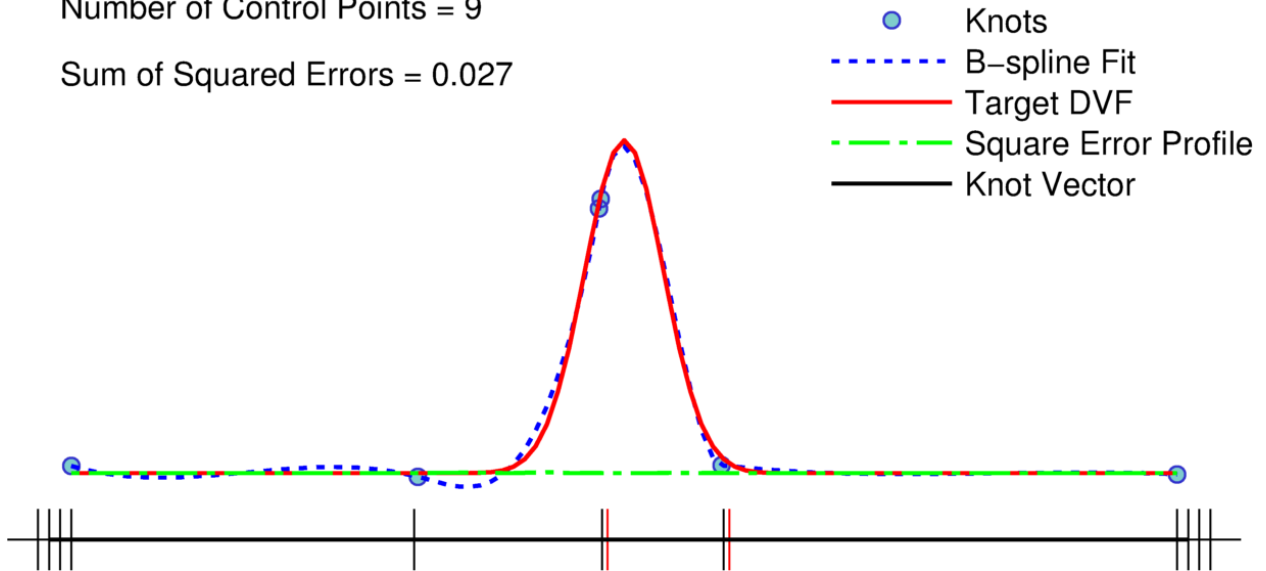
Sum of Squared Errors = 0.297



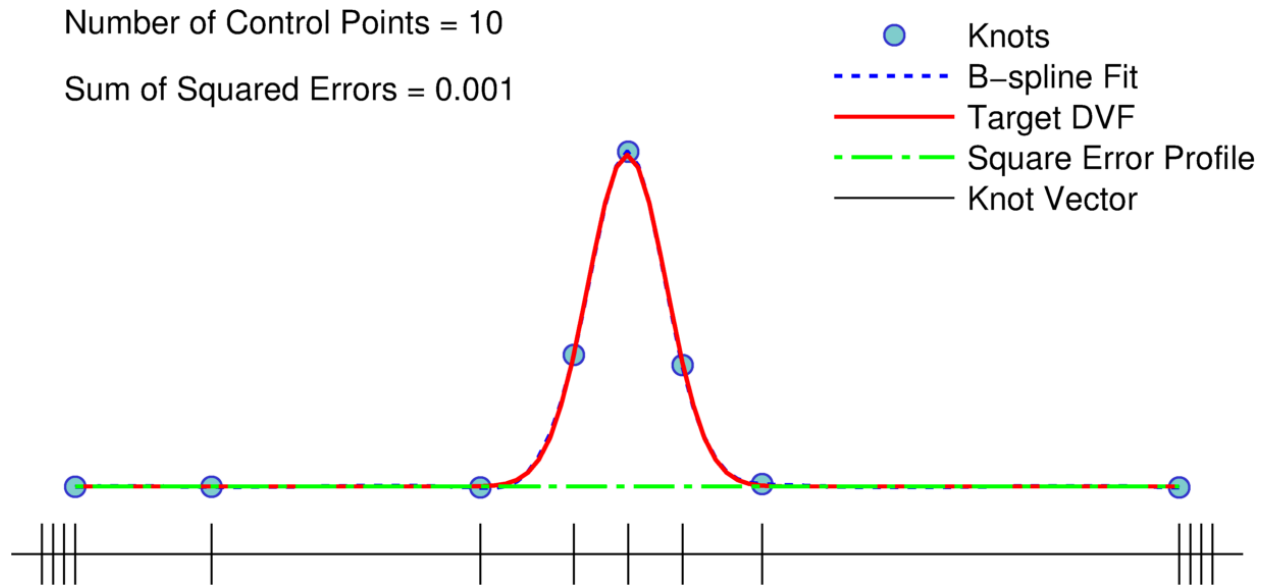
**Figure 31:** The solution for four free knots, which again exhibits repeated knot values (in red) and asymmetry.

Number of Control Points = 9

Sum of Squared Errors = 0.027



**Figure 32:** The solution for five free knots, which again exhibits repeated knot values (in red) and asymmetry.



**Figure 33:** The solution for six free knots. There are no longer any repeated knots, and the spacing of the central knots closely resembles the spacing of the uniform distribution.

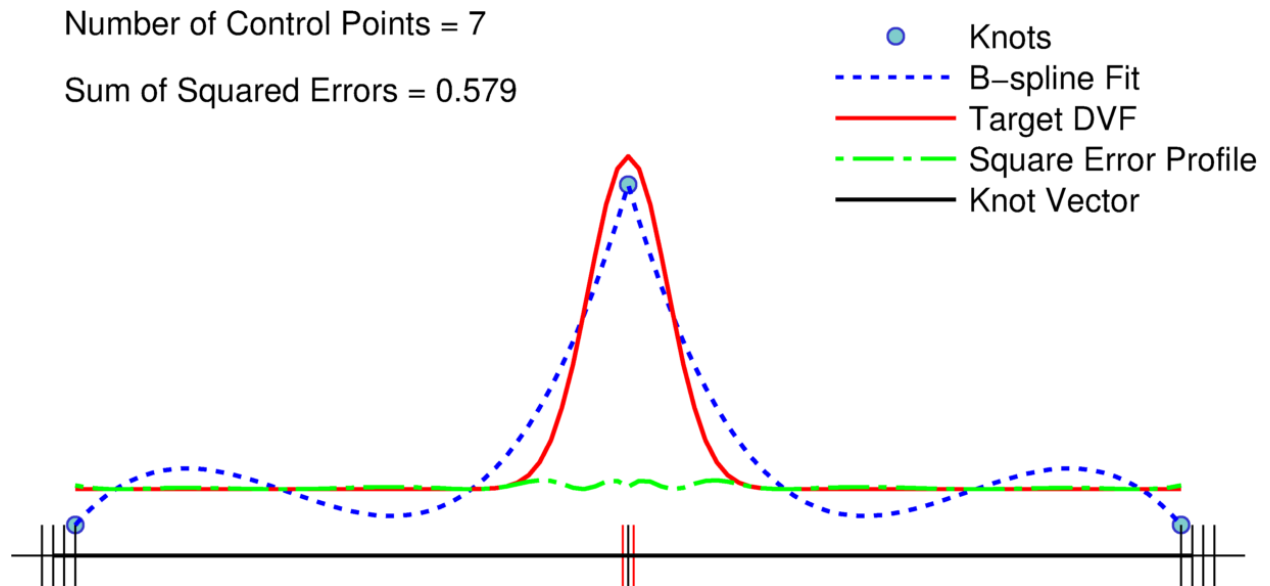
### 3.3.2 Knot Insertion at Maximum Error

The aforementioned dominant points method of knot optimization computes the knots through iterative samplings of feature points in the target function, such as areas of maximum curvature. This strategy relies upon *a priori* knowledge of the target function, which makes it incompatible with the DIR process where the underlying DVF is unknown. In the general DIR method, a DVF  $u(x)$  is used to map points  $x' = x + u(x)$  in a source image  $A(x')$  to points  $x$  in a target image  $B(x)$  such that  $A(x+u(x)) = B(x)$ . The DVF function  $u(x)$  is meant to approximate the true underlying DVF, which is unknown. Despite the fact that nothing is known about the true underlying DVF, one may still obtain information about where the DVF fit is deficient from the resultant difference image  $I(x) = A(x+u(x)) - B(x)$ . Using this observation, algorithms for B-spline fitting are developed which place knots based on the (absolute) difference between the target function  $f(x)$

and the fitting function  $u(x)$ . The first proposed method is to simply insert a knot at the parameter value for which the squared distance between  $u(x)$  and  $f(x)$  is a maximum. For the 1D test case of the Knot Insertion at Maximum error method (KIM), an initial fit is made with a uniform one-dimensional B-spline  $u(x) = \sum_i c_i b_{i,j}(x)$ . Distance to the target function  $f(x)$  is computed at each data point, and these values are squared to form the error function

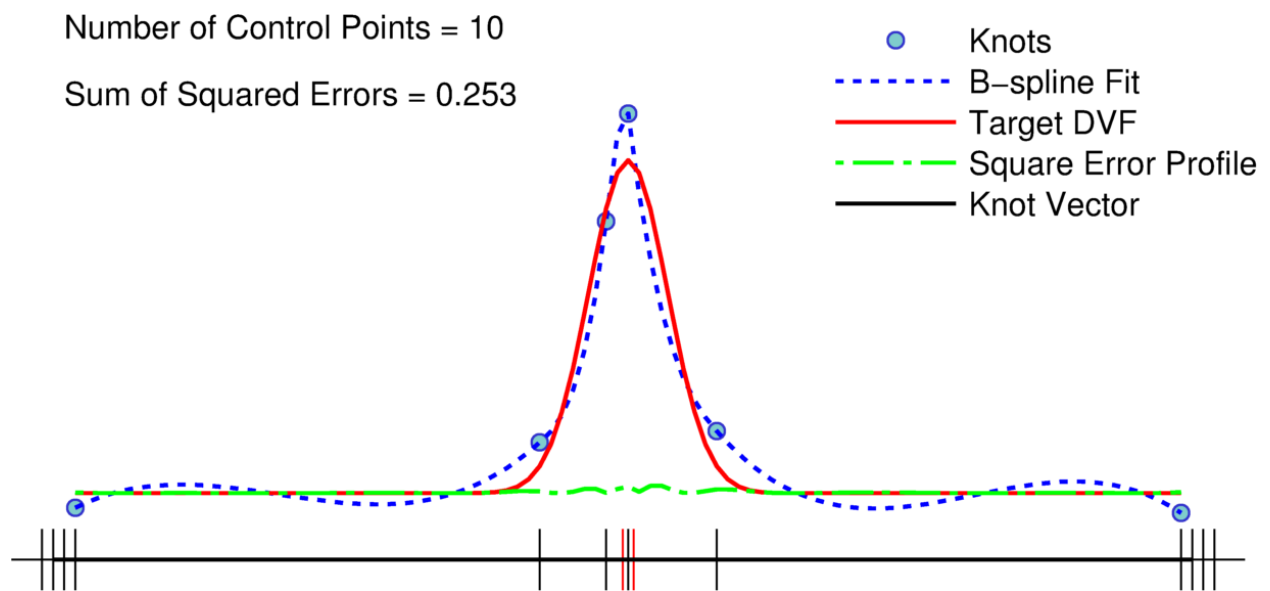
$\varepsilon^2(x) = [u(x) - f(x)]^2$ . This process is repeated iteratively so that each subsequent B-spline fit contains an additional control point with an associated knot. In the event that two equal maxima exist, the first one is selected as the position of the inserted knot. Iteration proceeds until an error tolerance criterion is met. For this test (as well as those that follow), the error tolerance is set to the sum of squared errors with four control points (zero free knots) divided by 1000.

Figures 34 through 36 demonstrate the method.

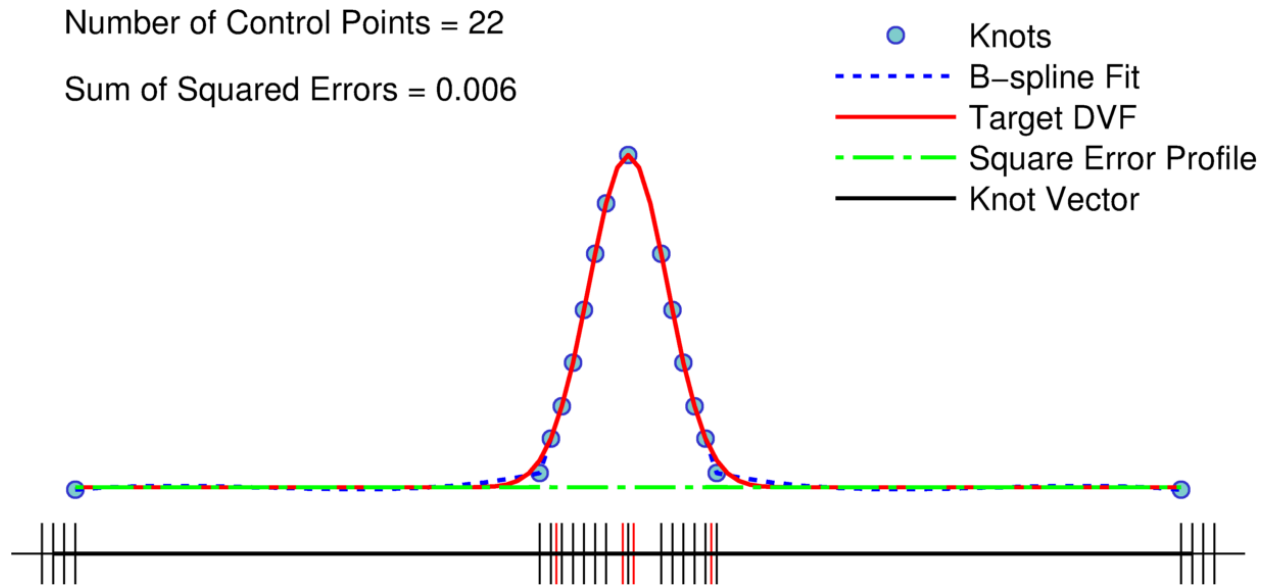


**Figure 34:** The first three inserted knots are all placed at the same central parameter value. Duplicate knots are displayed in red.

The first three knots are inserted at the same central point. As discussed at the beginning of this chapter, a knot value  $k$  with multiplicity equal to degree  $p$  collapses the B-spline basis function at that location. Essentially, a curve segment is “lost”, and the curve is forced to interpolate through the control point associated with the collapsed basis function. In the case of only three free knots, this distribution is optimal, as verified by the Simplex method. However, as the number of control points increases, the benefit of repeated knots is reduced. Inserting knots at the point of maximum error quickly reduces the error at that point but is suboptimal for the global error. In short, the method overcompensates. Note that while the Simplex and KIM methods both contain repeated knots in their solution vectors, the Simplex method shows a trend towards reducing the multiplicity of its repeated knots. In fact, the knot vector which satisfies the specified tolerance criterion does not contain repeated knots. This observation motivates the next method, which introduces a means by which the knot positions are automatically adapted after knot insertion.



**Figure 35:** The discontinuous kink created by the multiple knots reduces the shape control in the surrounding curve segments.



**Figure 36:** The KIM method overcompensates by adding too many knots in areas of high local gradient.

### 3.3.3 Force Equilibration Method

The Force Equilibration method (FE) adaptively refines the knot vector by allowing knots to migrate to areas in which a previous fit, computed with arbitrary knot spacing, produced the largest errors. Knot refinement is guided only by error information in order to make it compatible with the DIR process, where the underlying function the B-spline estimates is unknown. After an initial fit is made, distance to the target function  $f(x)$  is computed at each data point, and these values are squared to form the error function. A knot is inserted at the position  $x$  for which the error function is a maximum, as was the case in the KIM method. However, inserted knots are no longer required to remain at their initial position between iterations. Instead, the knots are free to redistribution according to a force equilibration scheme. The error  $\varepsilon^2(x)$  at any particular position  $x$  exerts a force to pull the knots toward that point (or, conversely, resisting efforts to move the knot further away). The force is a simple spring-like resistance proportional to the product of the

distance of the knot from a particular error magnitude  $\varepsilon^2(x)$  and the error magnitude itself. By summing over all of the restoring forces on a particular knot  $i$  over the range of points  $x$  bounded by knot  $i-1$  and  $i+1$ , one obtains the knot position at which the forces are equilibrated. For each knot  $i$ , solve for the knot position  $k_i$  that minimizes the sum of the contributions of each point of the error function contained in the knot segment belonging to  $k_i$

$$\mathbf{k}_{min} = \arg \min_{\mathbf{k}} \int_{x=k_{i-1}}^{x=k_{i+1}} |x - k_i| \cdot \varepsilon^2(x) dx \quad (3.26)$$

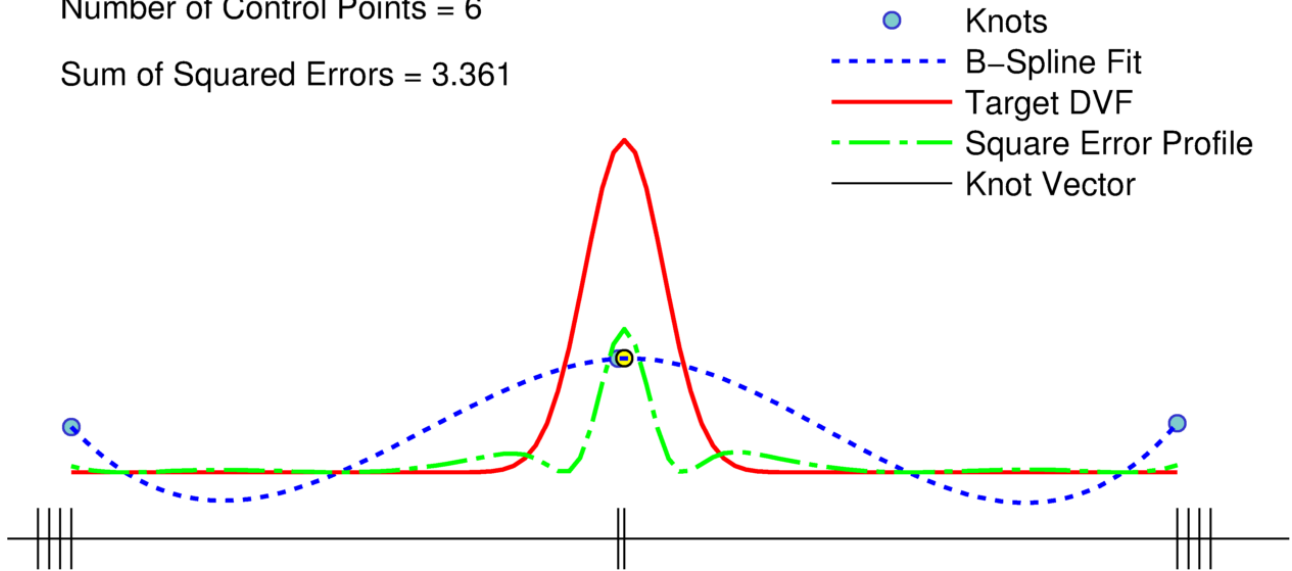
Starting with the first knot, each knot position was solved using the Nelder-Mead Downhill Simplex technique for minimization. The knot positions were found in sequence, such that the updated knot position  $k_i$  was used in the calculation of knot position  $k_{i+1}$ .

One can see intuitively that if a knot starts out somewhere between two symmetric error peaks, it will settle down half-way between them. If one of the two error peaks is larger, the knot will gravitate closer to it, but will still be restrained by the other errors, causing it to settle at a point that neutralizes the forces on it. The sum total of the restoring forces prevents the knots from piling up at the location of maximum error.

To mimic the hierarchical process of current B-spline DIR implementations, and also to underline the consistent advantage of the knot optimization method, the target DVF is fit iteratively. The initial fit is made with four control points (zero free knots). During each iteration, the error function is calculated and a knot is inserted at the point of largest error, the target is fit with the new knot vector, the error function is updated, knot placement is optimized by the force equilibration scheme, and finally the target is refit with the optimized knot vector. Iteration proceeds until a predetermined error tolerance criterion is met. Figures 37 through 44 show the first 4 iterations of the FE method.

Number of Control Points = 6

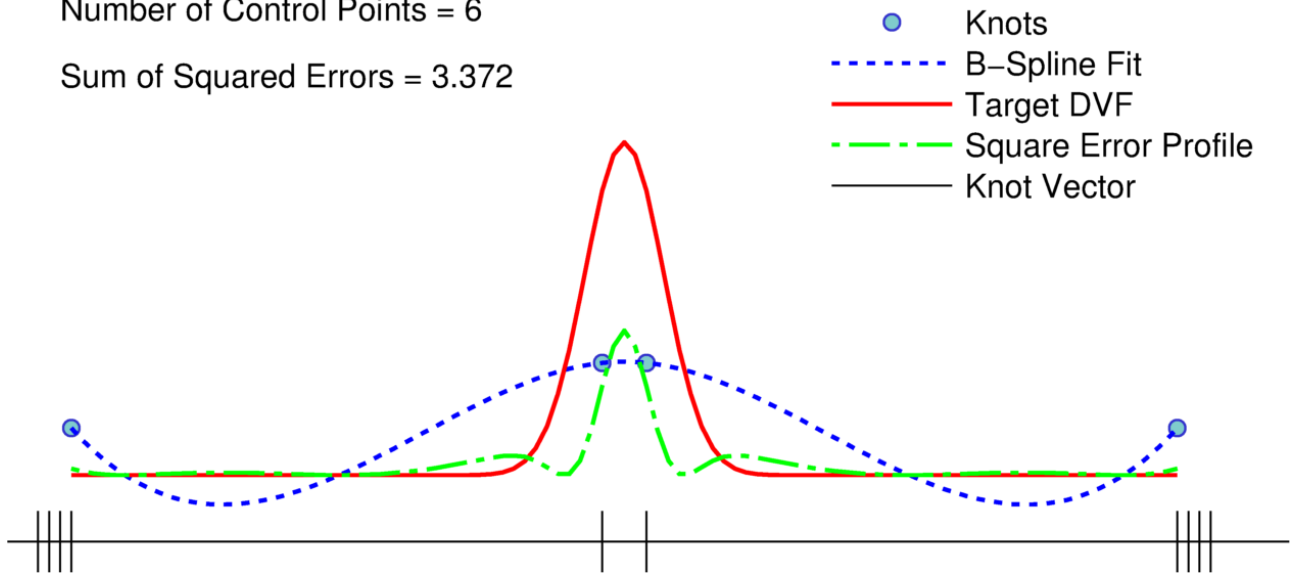
Sum of Squared Errors = 3.361



**Figure 37:** The first two knots are inserted at the maximum error, located at center of the Gaussian.

Number of Control Points = 6

Sum of Squared Errors = 3.372

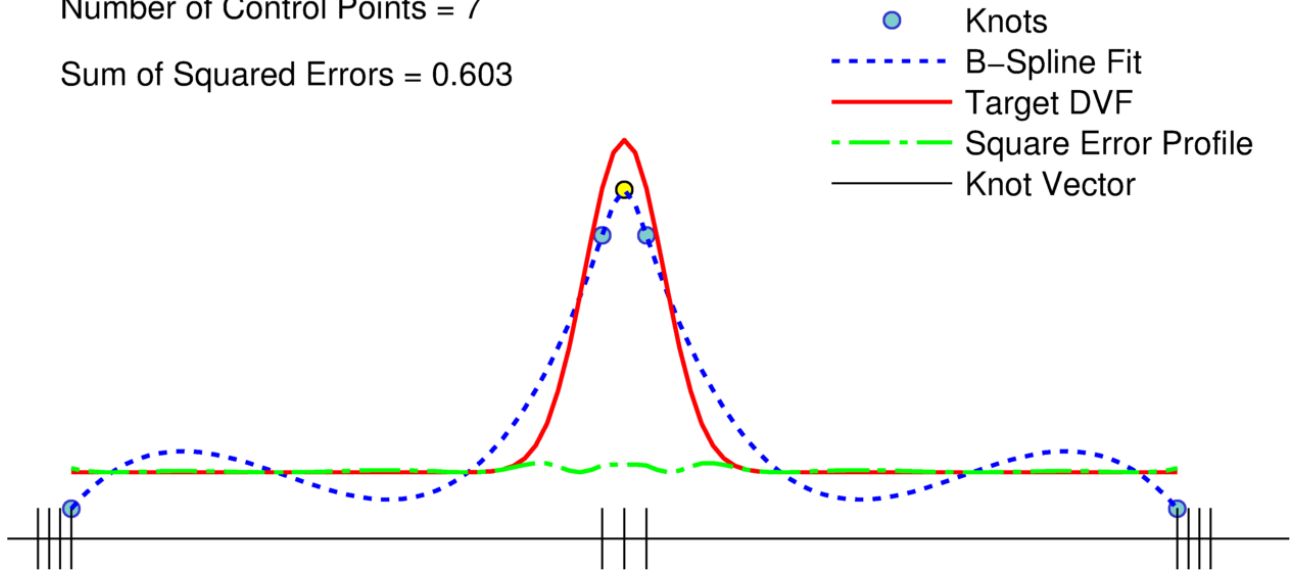


**Figure 38:** Force equilibration redistributes the two internal knots.



Number of Control Points = 7

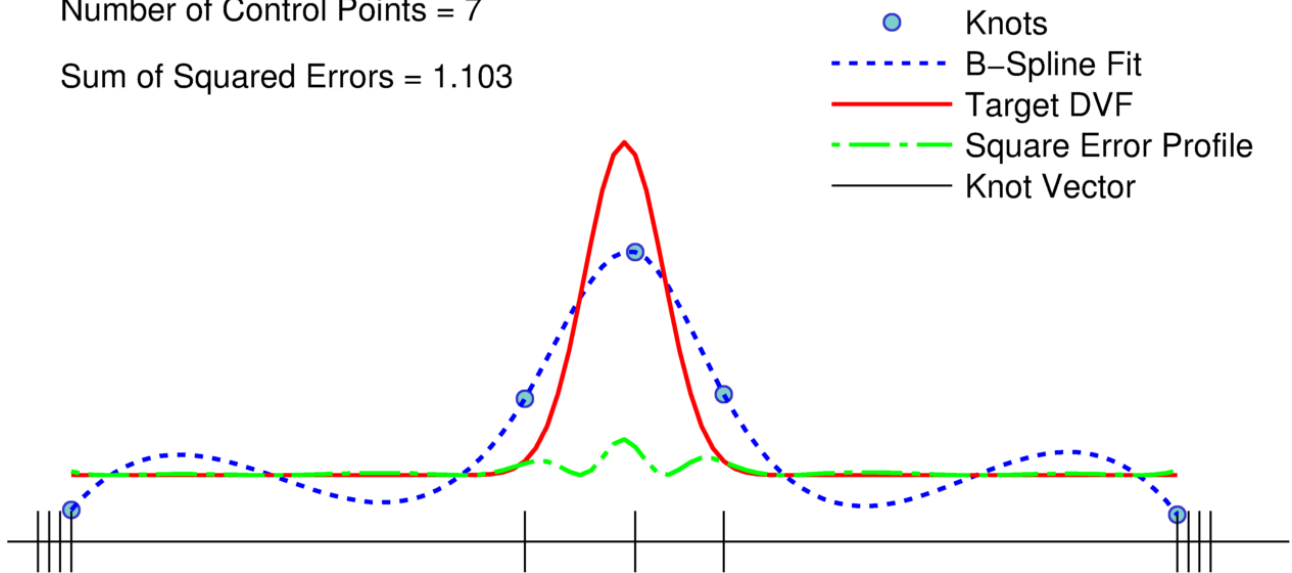
Sum of Squared Errors = 0.603



**Figure 39:** The third knot is placed at the midpoint of the Gaussian.

Number of Control Points = 7

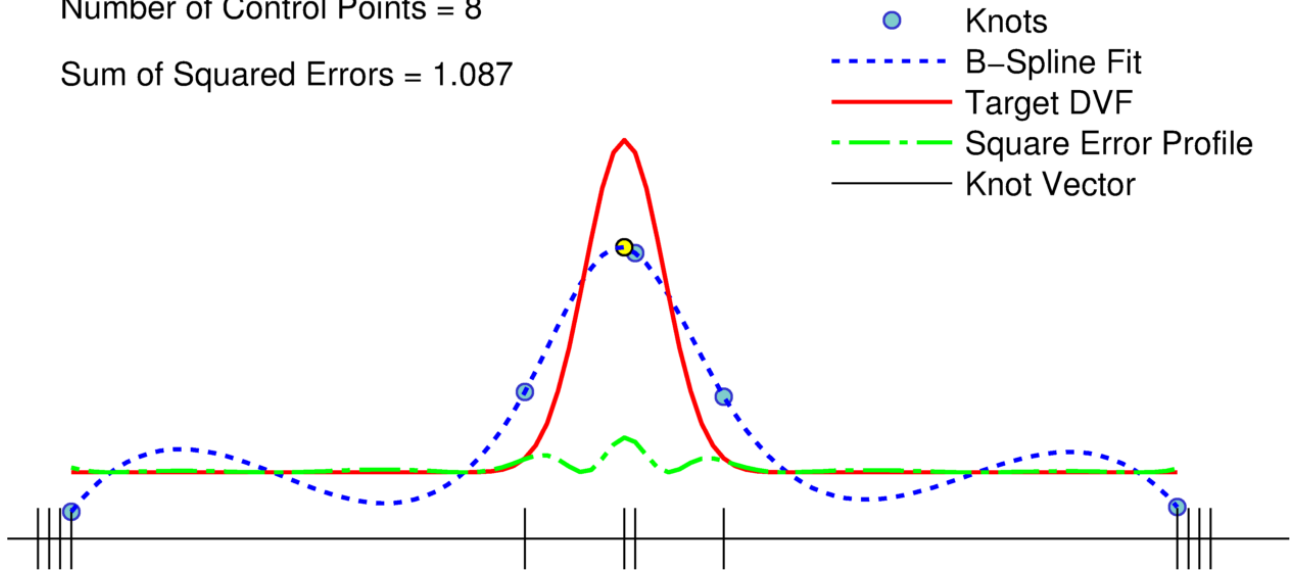
Sum of Squared Errors = 1.103



**Figure 40:** Force equilibration redistributes the three internal knots.

Number of Control Points = 8

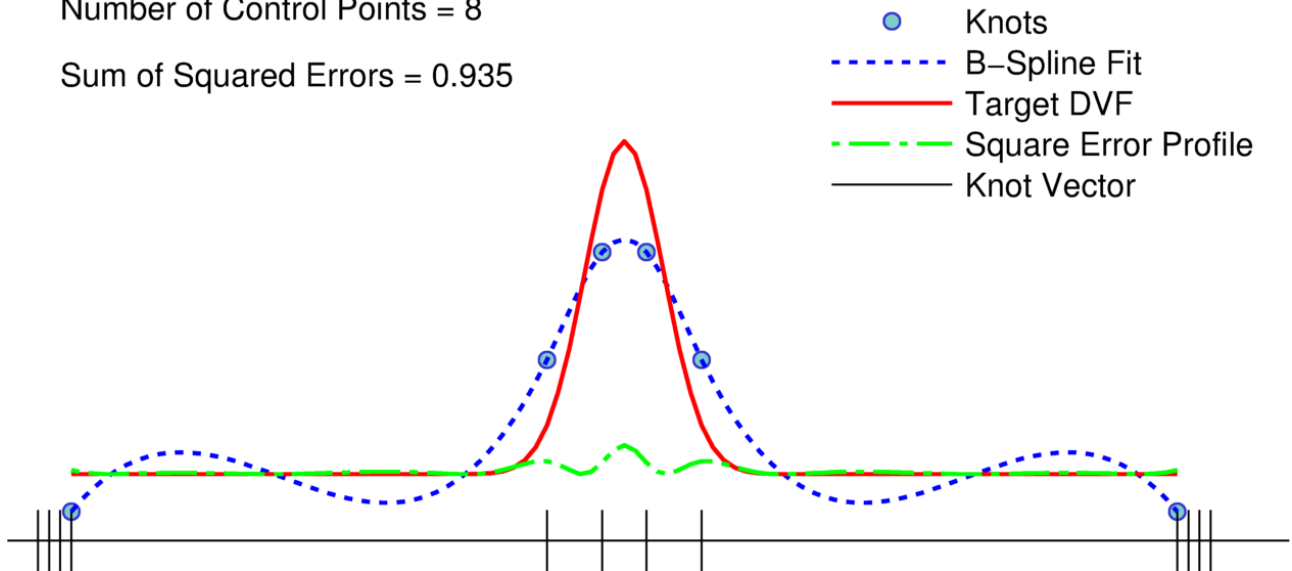
Sum of Squared Errors = 1.087



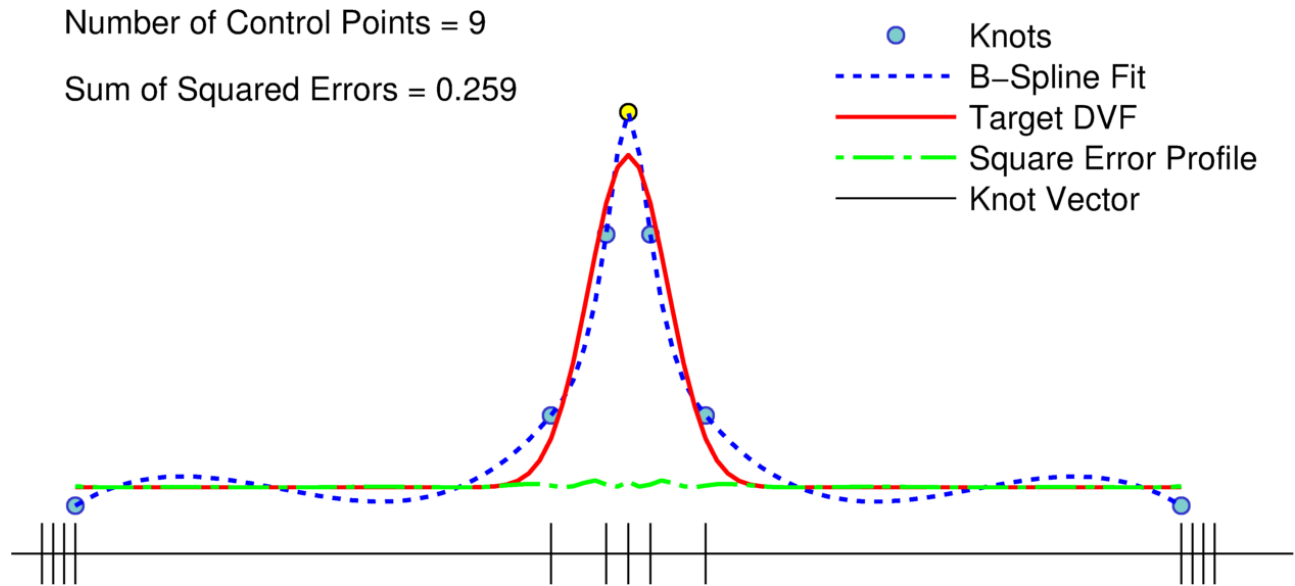
**Figure 41:** Another knot is inserted at parameter value corresponding to the center of the Gaussian.

Number of Control Points = 8

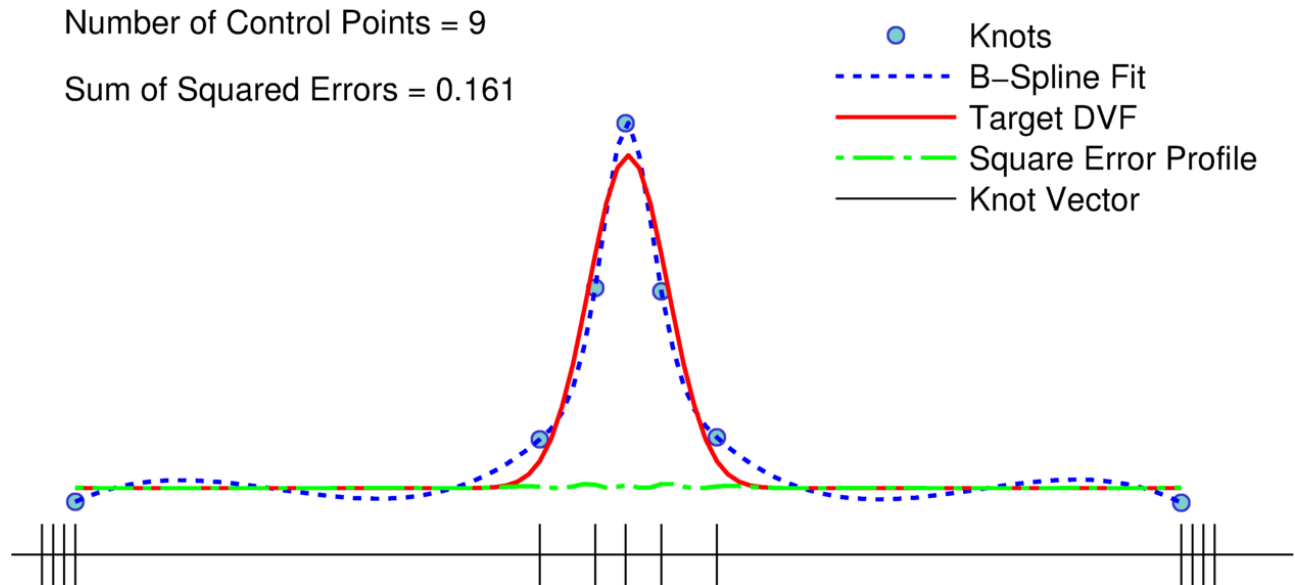
Sum of Squared Errors = 0.935



**Figure 42:** The force equilibration scheme has prevented the knot pile up observed in the KIM method.



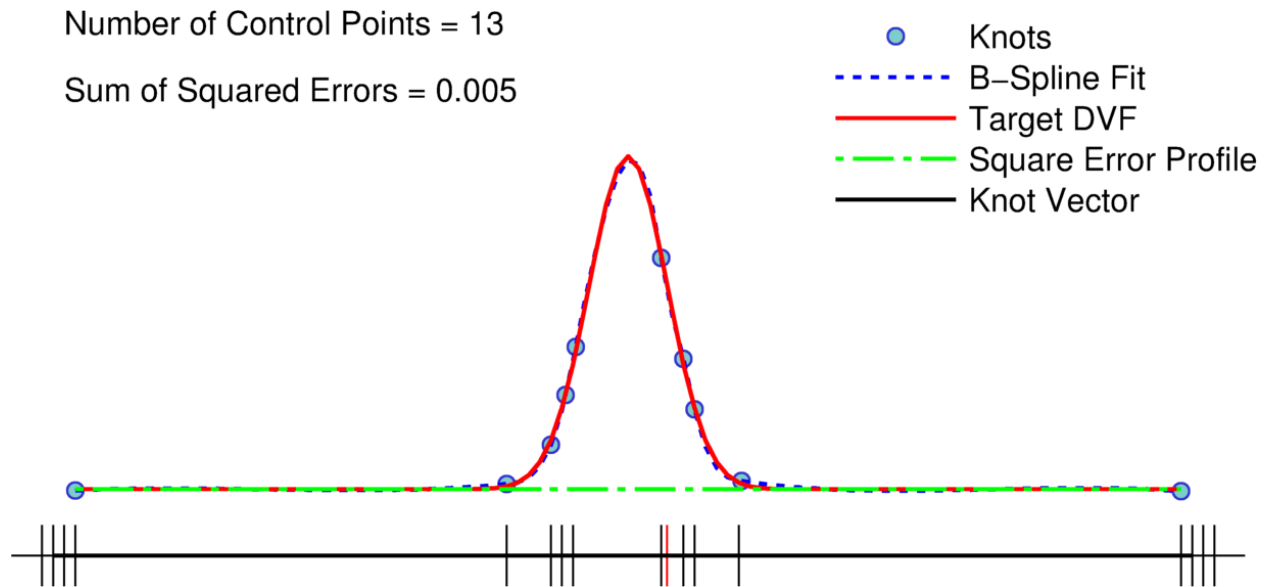
**Figure 43:** The largest error continues to reside at the center of the curve.



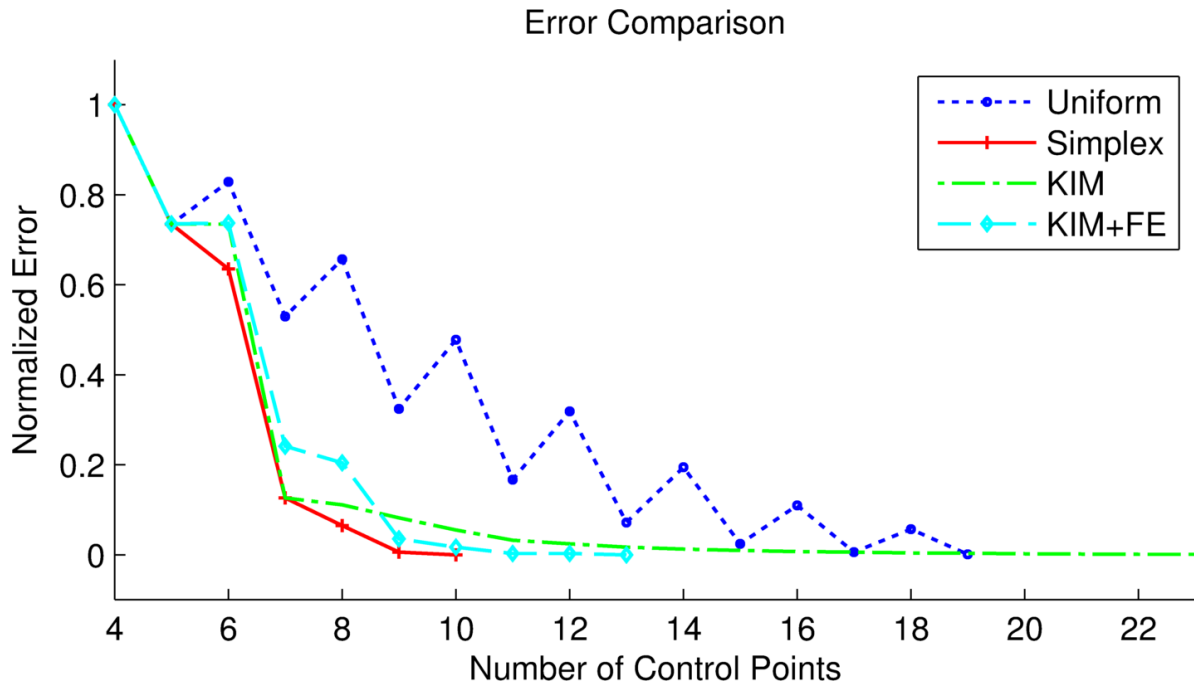
**Figure 44:** The force equilibration scheme continues to spread the control points away from the center, thereby avoiding a knot distribution that is too narrow.

The inclusion of a redistributive step after knot insertion corrects for the KIM method's tendency to place too many knots over a small segment of the B-spline curve. It's worth noting

that the knot redistribution is not guaranteed to produce a better fit for the same number of control points. This can be seen in figures 37 and 38. Despite this fact, the force equilibration strikes a balance between the stability of the uniform knots and the flexibility of locally dense, non-uniform knots. This enables it to meet the error tolerance with fewer control points (Figure 45). Figure 46 compares sum of squared errors produced by the uniform, simplex, KIM, and FE methods as a function of the number of control points. The non-uniform methods consistently out-perform the uniform knot fit. They also exhibit a monotonic decrease in error, whereas the uniform fit fluctuates. This is attributable to the placement of the uniform knots fortuitously coinciding with areas of high curvature.



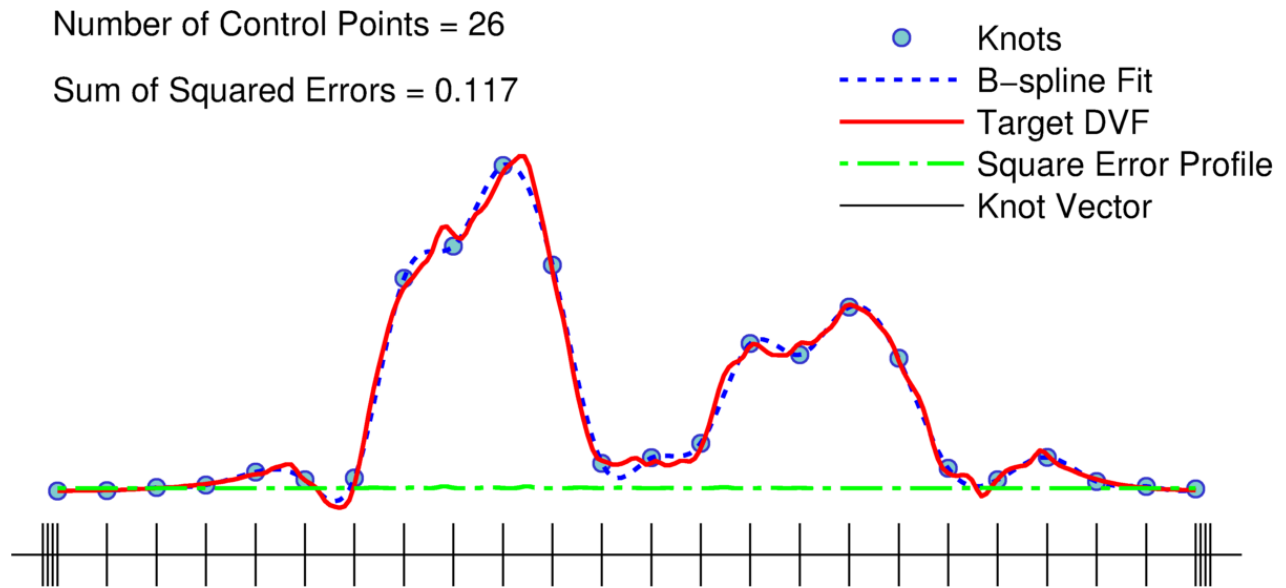
**Figure 45:** The final knot distribution for the force equilibration method.



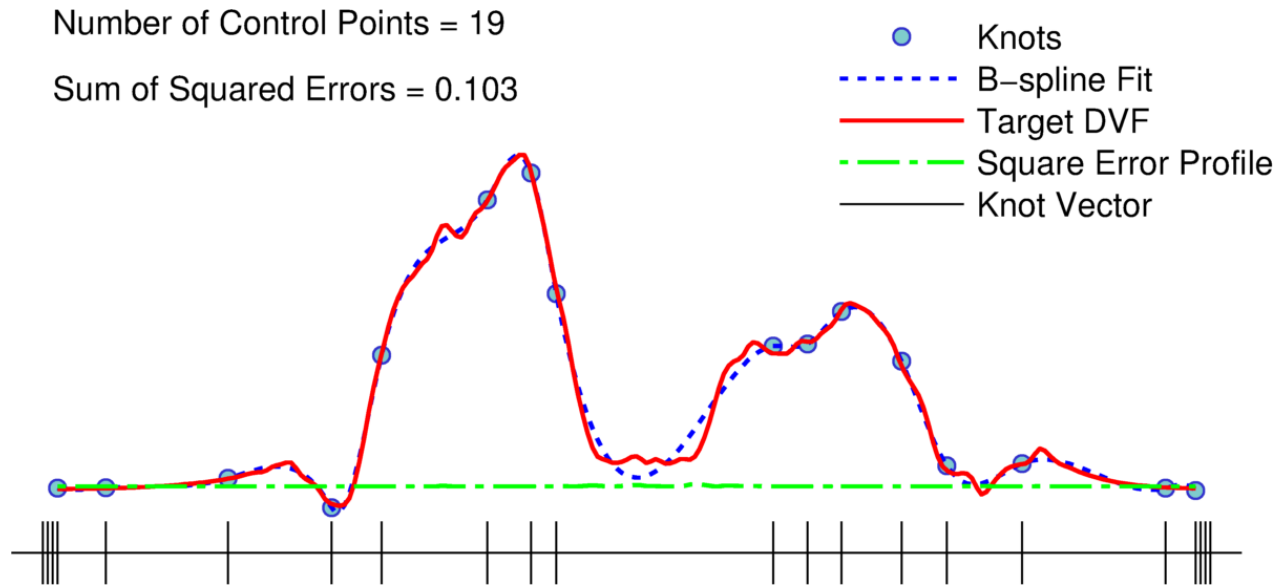
**Figure 46:** Comparison of the sum of squared errors normalized to the initial error.

To further evaluate the comparative efficacy of the knot placement methods, each algorithm was used to fit the 1D DVFs from section 3.2. Knots (or equivalently, control points) were added incrementally until the sum of squared errors was within one percent of the initial sum of squared error (the fit with zero free knots). Figure 47 shows the uniform B-spline fit to the first target DVF. This target curve is composed of the z-component of the DVF, sampled along a line parallel to the x-axis, and contains the structure of the breathing motion present in the lungs. Once again, the non-uniform knot distribution exhibits an efficiency gain. The Simplex method (figure 48) produces a better fit with 7 fewer control points. The KIM method with force equilibration (KIM+FE) also shows improved efficiency over uniform knots (figure 49). However, without the equilibration step, it fails to converge. This test showed that repeatedly adding knots to the parameter value of worst fit is not always adequate to reduce the

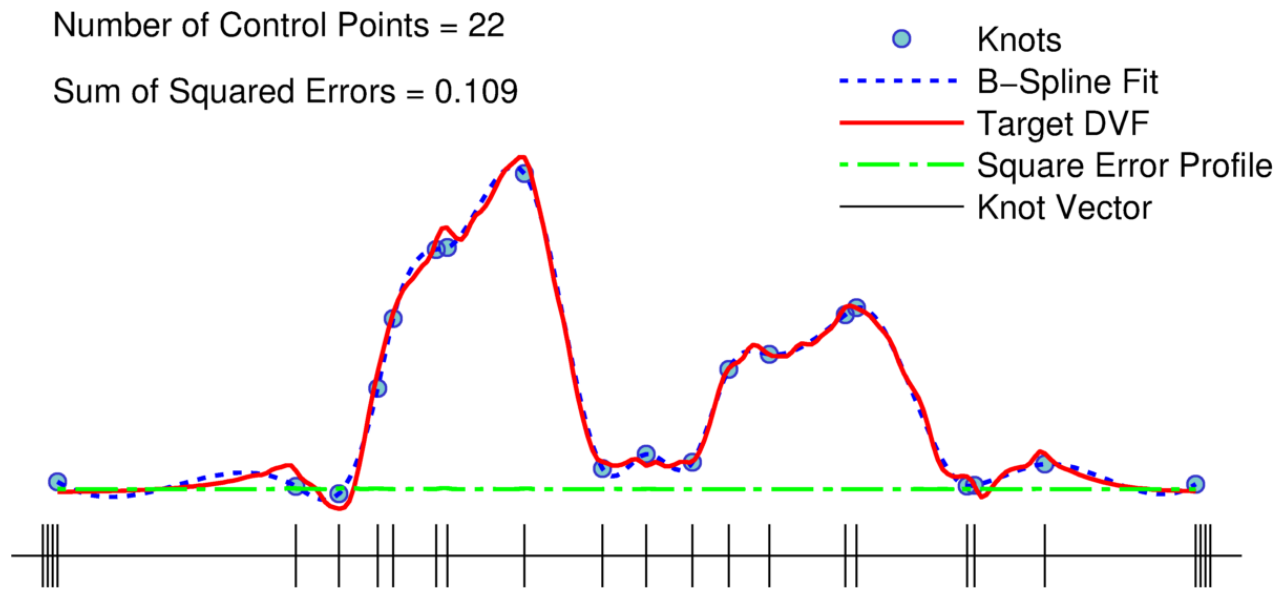
error at that point. The algorithm eventually inserted too many knots at a single parameter value, which produced a knot vector that fails to satisfy the Schoenberg-Whitney condition (figure 50). In this case, the matrix  $A^T A$  is singular and the algorithm fails to solve equation 3.5. To mitigate this problem, a new knot insertion strategy was created. Instead of inserting the knot at the location of maximum error, the integral error of each segment is computed and a knot is inserted into the segment with maximum integral error such that it bisects the segment's integral error. This method is referred to as the knot insertion by bisection of error method (KIB).



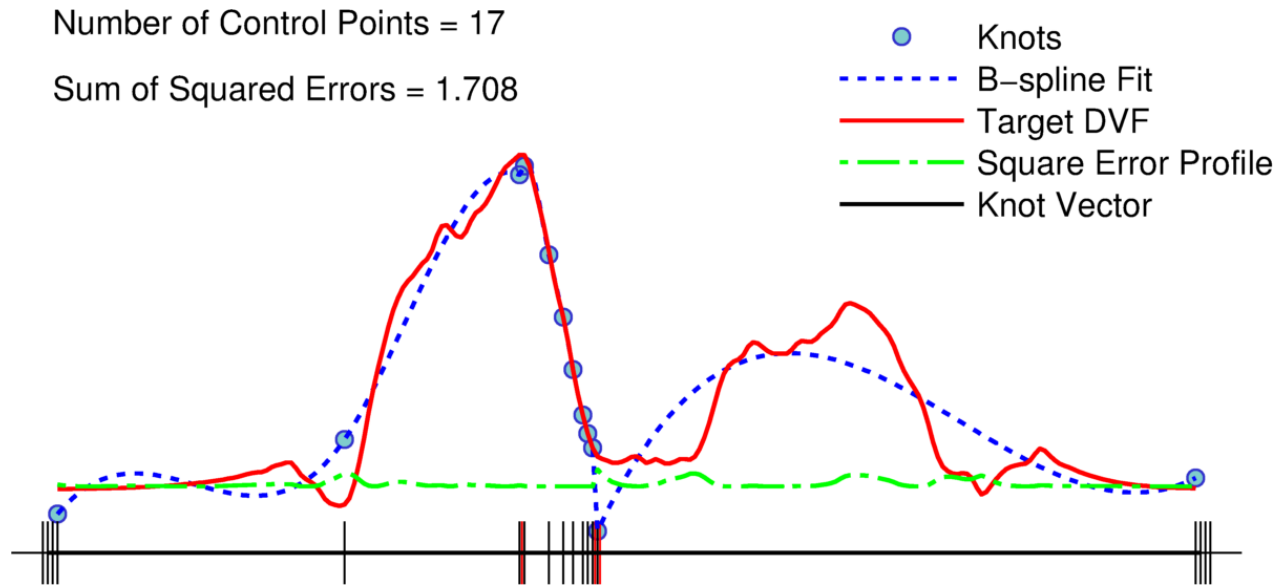
**Figure 47:** The 1D DVF profile in the x-direction fitted with a uniform B-spline.



**Figure 48:** The 1D DVF profile in the x-direction fitted with a non-uniform B-spline computed by the Simplex method.



**Figure 49:** The 1D DVF profile in the x-direction fitted with a non-uniform B-spline computed by knot insertion at location of maximum error followed by knot force equilibration (KIM+FE).



**Figure 50:** The 1D DVF profile in the x-direction fitted with a non-uniform B-spline computed by knot insertion at location of maximum error. The repeated knot insertion in a small region led to an algorithmic error, motivating a new knot insertion scheme.

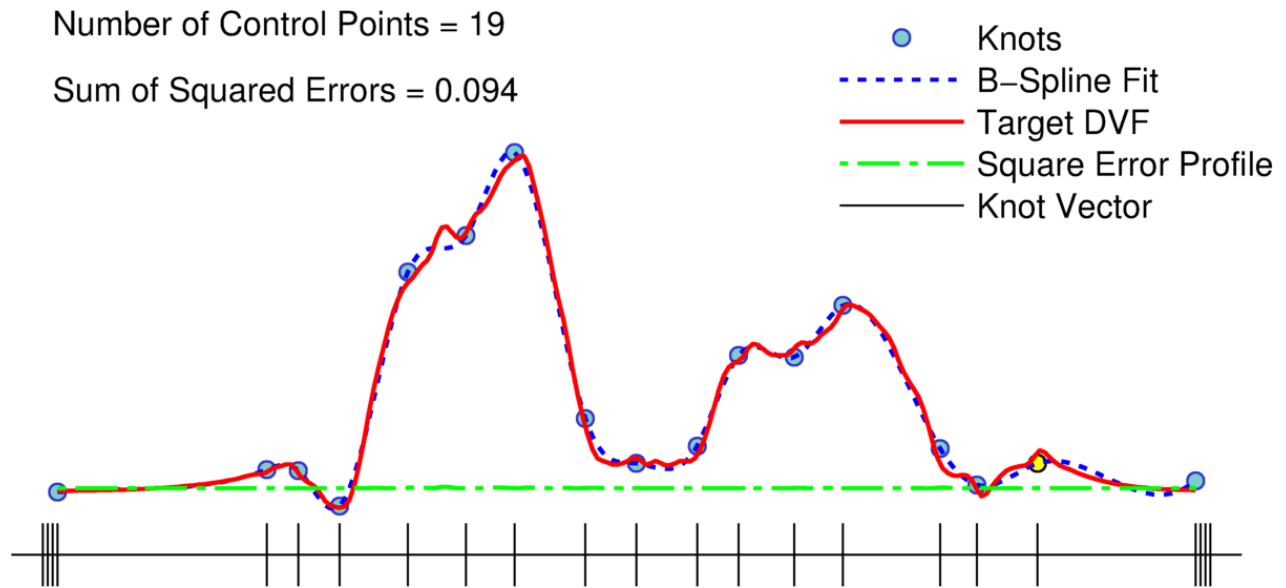
### 3.3.4 Knot Insertion by Bisection of Error

As was shown in the previous section, the performance of the KIM method suffered from a tendency to place too many knots in a small region of the target function. To counteract this, a modification was made such that a knot was placed within the region that exhibited the largest error rather than at the point of the largest error. The regions were defined as the knot segments (or knot spans), i.e. all the points  $x$  between two consecutive knots  $k$ . The integral error is computed for each knot segment, and a new knot value is placed such that it bisects the segment with the largest error. This prevents the bunching seen in the KIM method because as more knots are added to a local region of large error, the knot segments within that region contain fewer points contributing to the integral error. As the number of points within a knot segment decreases, so does the integral error relative to those of larger knot segments. This creates a

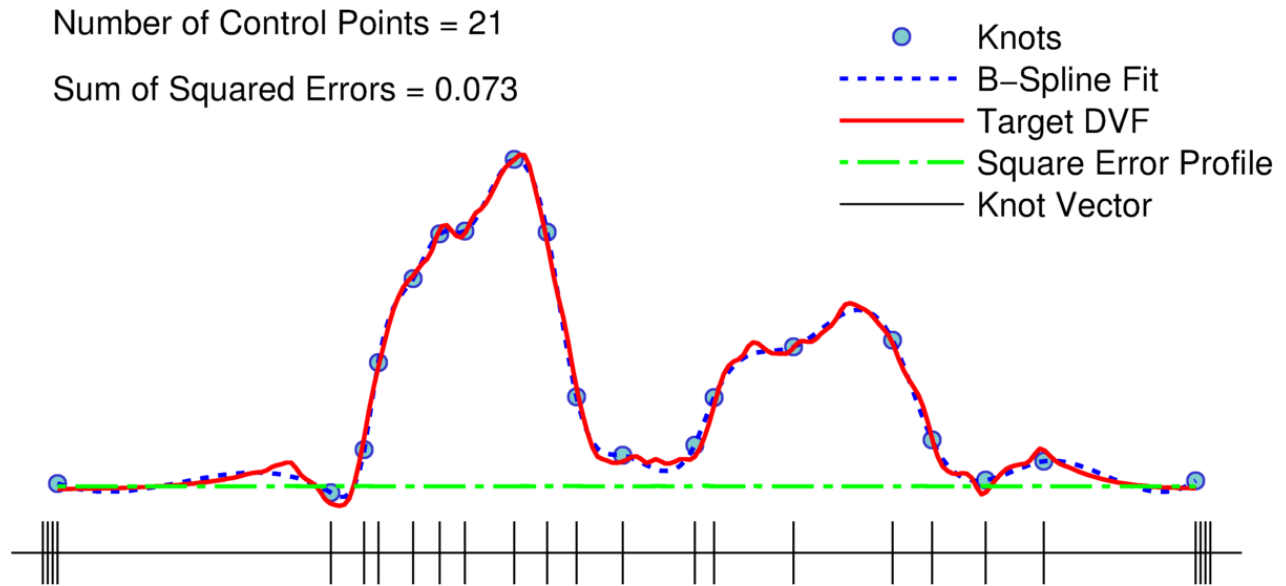


balance in knot placement between dense knot regions with extreme outliers and sparse knot regions with many points having above average error. Subsequently, the KIB is less susceptible to overcompensating in a small region or single point.

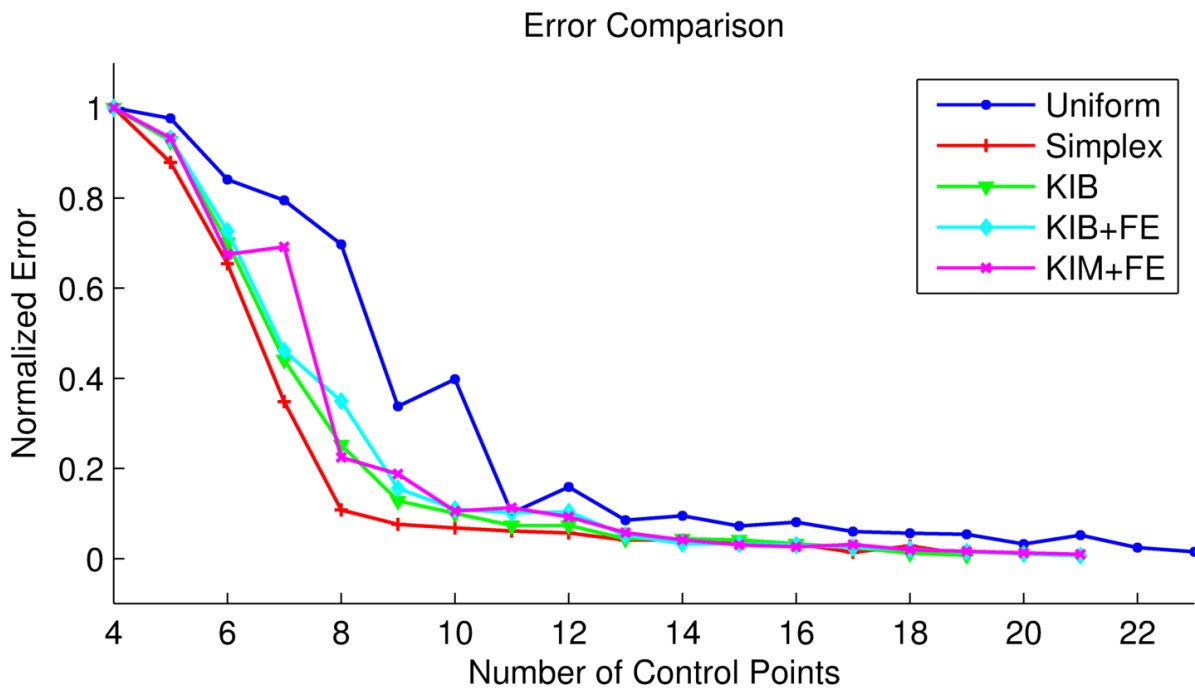
The fitting results using KIB are shown in Figure 51. Surprisingly, it achieves a better fit than the Simplex method for the same number of control points (figure 48). This reveals that the Simplex method does not necessarily find the optimal knot configuration, even in 1D. For this test case, the KIB method performs better on its own than when it is combined with the FE method. Figure 53 compares the normalized errors of the algorithms at the end of each knot insertion step.



**Figure 51:** The 1D DVF profile in the x-direction fitted with a non-uniform B-spline computed by the KIB method.

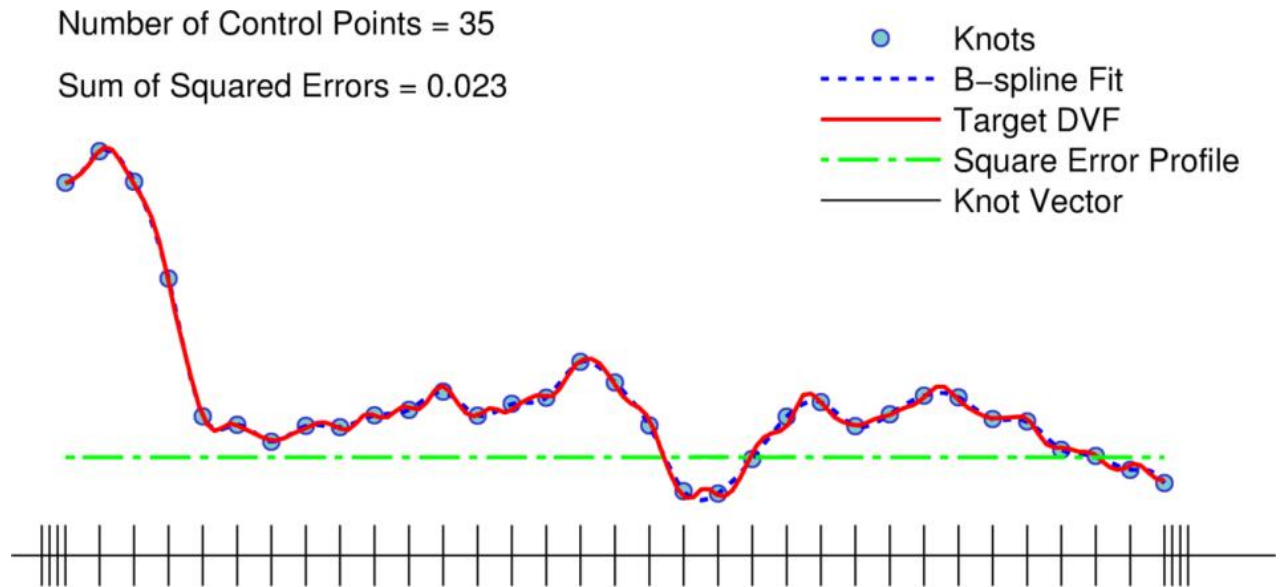


**Figure 52:** The 1D DVF profile in the x-direction fitted with a non-uniform B-spline computed by KIB+FE.



**Figure 53:** Comparison of the sum of squared errors of each algorithm fitting the 1D DVF profile in the x-direction.

A third 1D fitting test case was created by sampling the x-component of the DVF along the z-axis. This dataset contains a peak at towards its left end, and smaller, noise-like undulations throughout. The fine detail of these undulations requires that the B-spline have more knots than the previous case in order to reach the same tolerance criterion. With the exception of one data point with a small number of control points, the non-uniform methods continue to show an increase in efficiency in B-spline curve fitting. The outlying data point is indicative of the knot placement algorithms overcompensating for the large initial error at the left-most peak. However, the non-uniform algorithms consistently produce smaller sum of squared errors for larger number of control points (see figure 59).



**Figure 54:** Uniform B-spline fit to 1D DVF profile in the z-direction.

Number of Control Points = 25

Sum of Squared Errors = 0.024

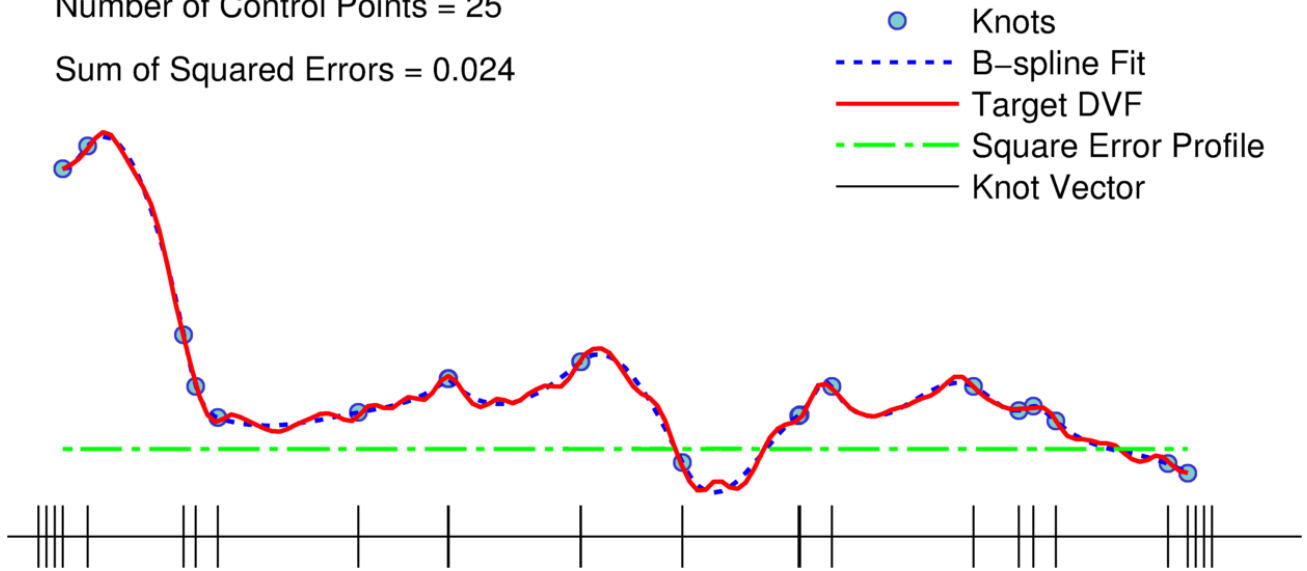


Figure 55: Non-uniform B-spline fit using Simplex method.

Number of Control Points = 28

Sum of Squared Errors = 0.021

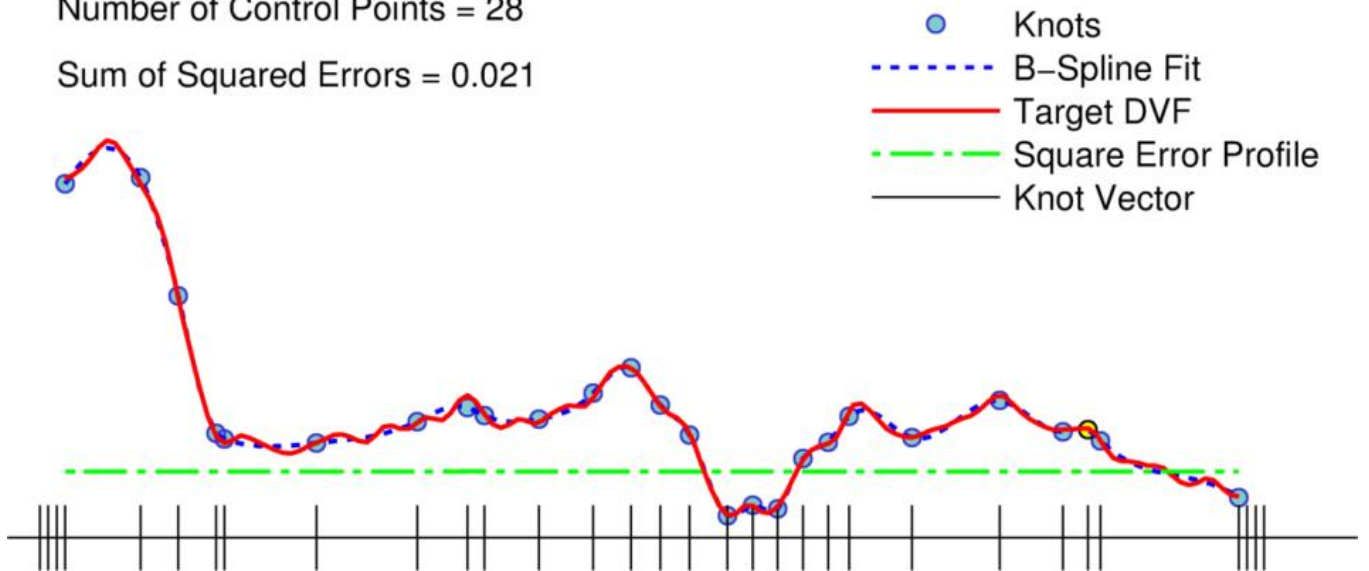


Figure 56: Non-uniform B-spline fit using KIB method.

Number of Control Points = 34  
Sum of Squared Errors = 0.025

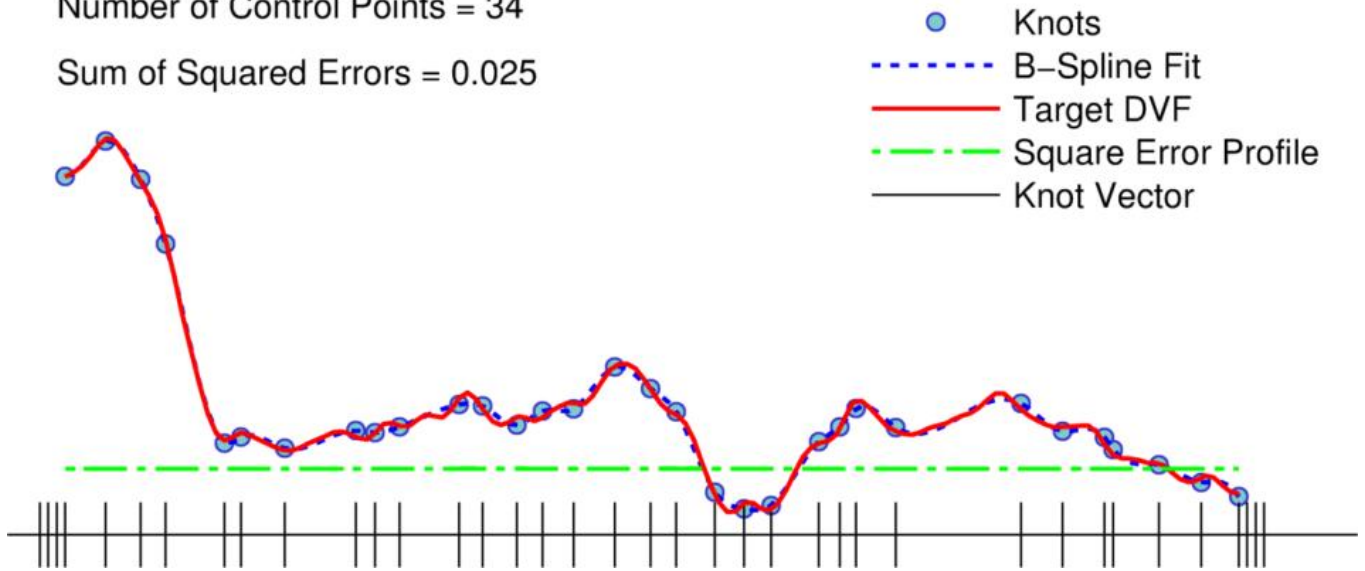


Figure 57: Non-uniform B-spline fit using KIB+FE method.

Number of Control Points = 38  
Sum of Squared Errors = 0.016

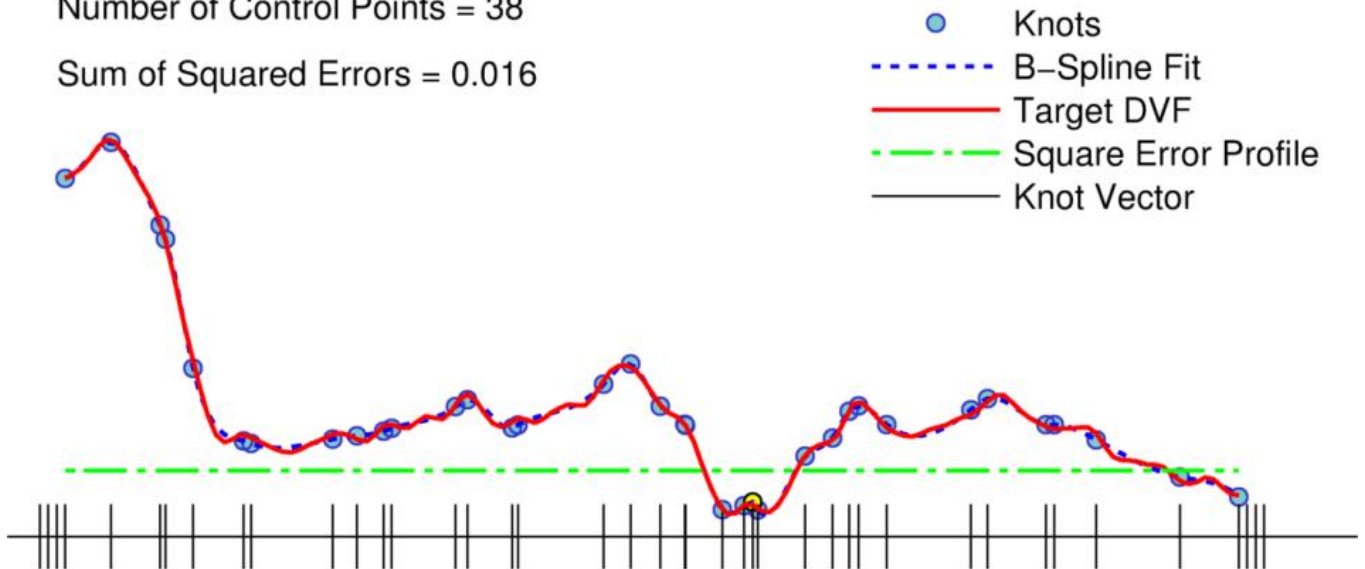
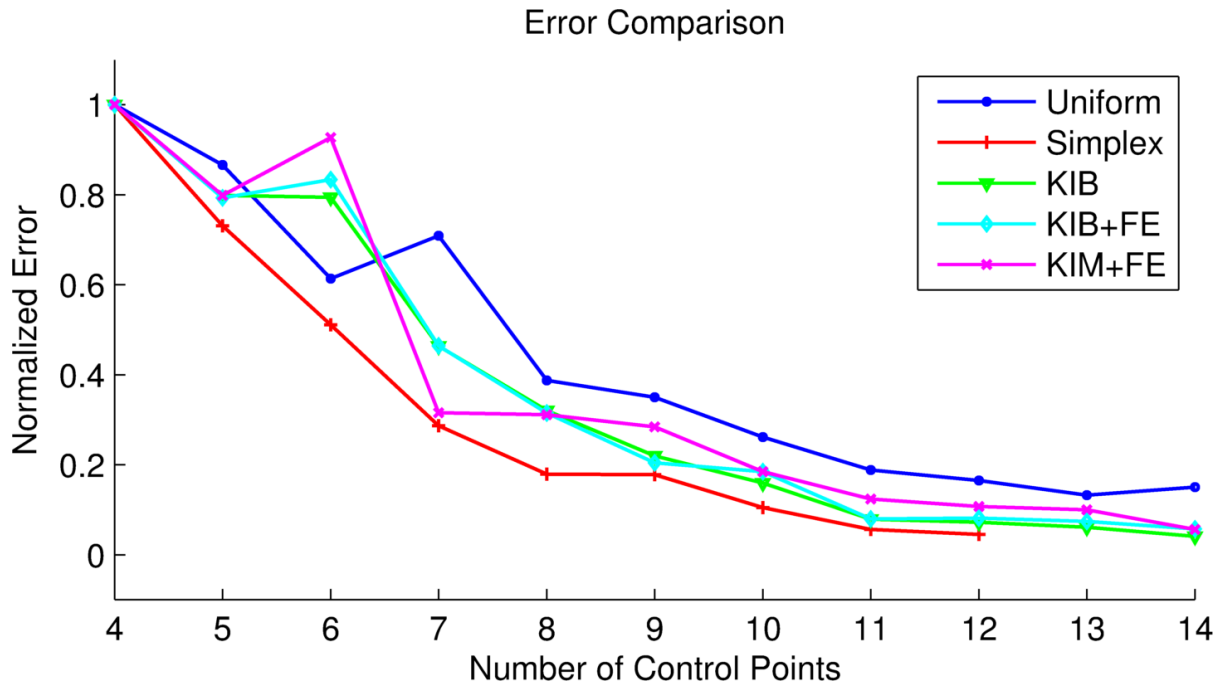


Figure 58: Non-uniform B-spline fit using KIM+FE method.



**Figure 59:** Comparison of the sum of squared errors of each algorithm fitting the 1D DVF profile in the z-direction.

### 3.4 Summary of Investigation

Chapter 3 introduced NURBS and investigated methods for fitting them to 1D functions. In Section 2, it was discovered that for the purpose of fitting 1D DVFs, the rational weights of NURBS offered no benefit over the same number of control points. This surprising result led to the discontinuation of weight optimization and shifted the focus of the study to optimizing the non-uniform knot vector. Section 3 developed heuristic rules for adaptively inserting knots in areas of deficient fit. It was demonstrated that non-uniform knots allow additional shape control of 1D B-splines and eliminate the need for a large number of redundant uniform knots. The next section will investigate whether or not the same benefit is seen in 2- and 3D tensor product NURBS.

## 4. Non-Uniform B-Spline Tensor Products

Tensor products of B-spline basis functions can be used to define surfaces and volumes. By using vector-valued control points, the B-spline tensor products can represent 2D and 3D DVFs. For cubic B-splines, the 3D representation takes the form:

$$u(x, y, z) = \sum_{i=0}^3 \sum_{j=0}^3 \sum_{k=0}^3 b_i(x)b_j(y)b_k(z)P_{ijk} \quad (4.1)$$

Where  $b$  represents B-spline basis functions and  $P_{ijk}$  represents a 3D grid of vector-valued control points. The function represented by equation 4.1 (i.e., the displacement vector amplitude) is smoothly continuous across the full area or volume of the DVF. It cannot represent discontinuities within the DVF.

### 4.1 Non-Uniform Knot Placement for Tensor Product B-Splines

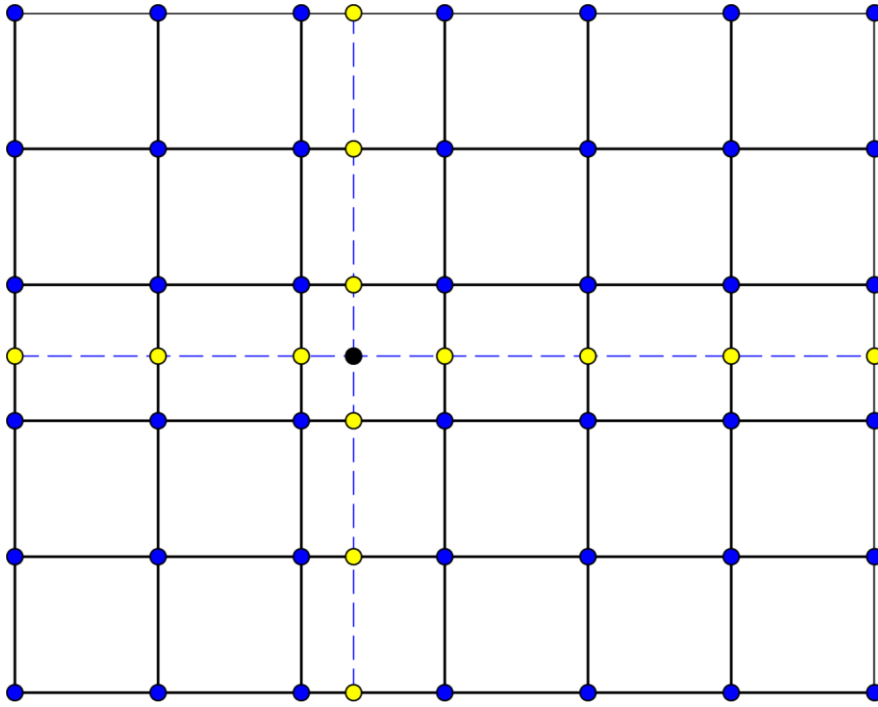
Each basis function segment  $b_i(x)$  contributes to the displacement vector amplitude in a sub area (or sub volume) of the DVF. These local domains are marked off by the so-called knot positions (i.e., the knot vector) for the B-spline. In a conventional B-spline formulation, the connected knot positions form a uniformly-spaced rectilinear grid. Within each cell of the grid the DVF amplitude is shaped by one or more free control point parameters; conversely, each control point influences only the vectors in a small neighborhood of grid cells.

The spatial derivative of the displacement vector function represented by equation 4.1 is determined by the size of the DVF domain controlled by each particular basis function segment.

This domain is related to the knot grid. If the domain is made smaller, the local derivative of the DVF can be increased, and sharper DVF features can be represented. This is conventionally accomplished by increasing the number of knots (and corresponding control points) uniformly across the DVF, which reduces the size of each grid cell. However, one could influence the local derivative of the DVF by spacing the knots non-uniformly. Decreasing the knot spacing in a region reduces the range of the basis functions in that region and allows the representation of steeper gradients in the DVF.

The knot vectors form a rectilinear grid over the surface that determines the control point support. Whereas the value of a point on a 1D B-spline is affected by  $p+1$  control points, the value of a point on a 2D B-spline tensor product is affected by  $(p+1)^2$  control points. By definition, the knots are not permitted to move freely. Instead, they are constrained to move orthogonally as lines of knots in the 2D plane or planes of knots in a 3D volume. Furthermore, a single knot cannot be added without propagating it across the parameter space. For example, inserting a knot at position  $(a,b)$  requires adding the value  $a$  to the knot vector in the  $x$  direction and the value  $b$  to the knot vector in the  $y$  direction. This is illustrated in Figure 60.



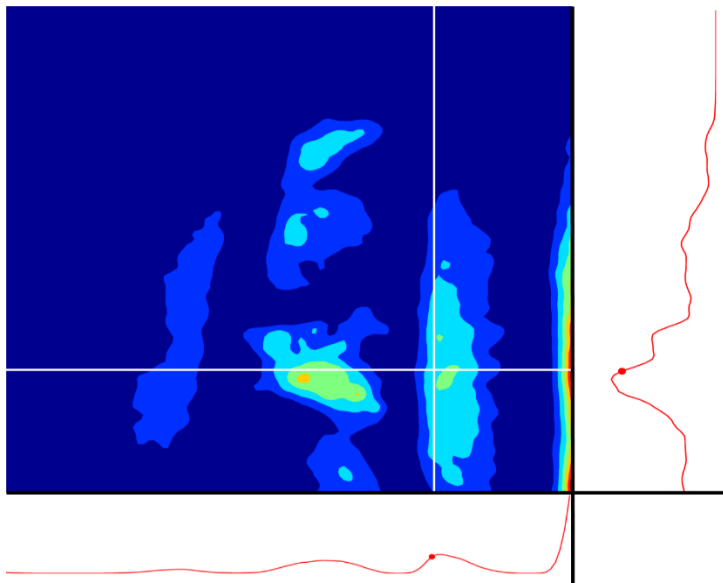


**Figure 60:** Inserting a knot (in black) to the Cartesian point  $(a,b)$  requires adding the values  $a$  and  $b$  to the  $x$  and  $y$  knot vectors, respectively.

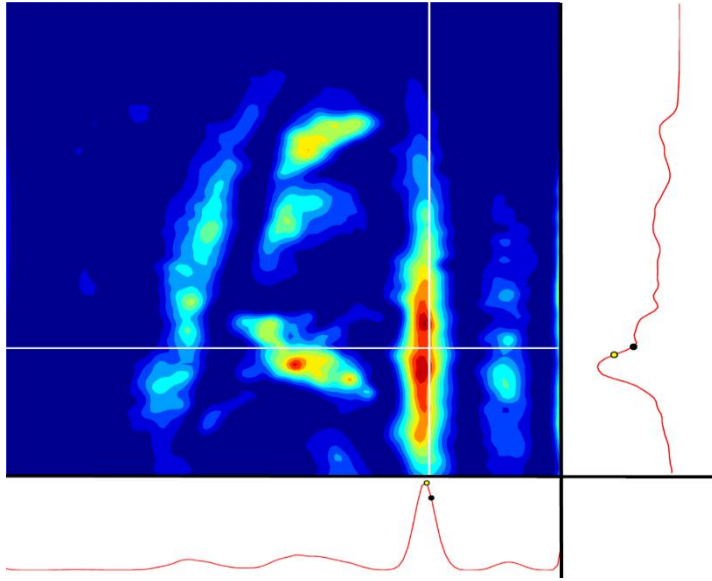
Recalling that each additional knot requires a new control point associated with that knot, it is easy to see that an optimal knot grid could potentially lead to a substantial reduction of the total number of control points in tensor product B-splines. A major difficulty in optimizing a B-spline fit with respect to the knots is that their values (along with the specified degree of the B-spline) explicitly determine the basis functions. This means that the knots must be selected before any attempt is made to solve for the control points values, and any change to the knots would require a new optimization of the control points. The resulting “nested” optimization scheme is computationally expensive, and can be avoided by extending the heuristic rules developed in chapters 2 and 3 to higher dimensions. Given that each knot vector is constrained to move along one axis, it is sensible to re-use the 1D method. To do so, the error (distance to fit) at each point is summed along the  $x$  and  $y$  axes to obtain two 1D error profiles. The 1D fitting

methods are then applied directly to the error profiles, thus creating two independent knot vectors. This approach, in addition to being straight-forward and intuitive, has the advantage of reducing the computational complexity.

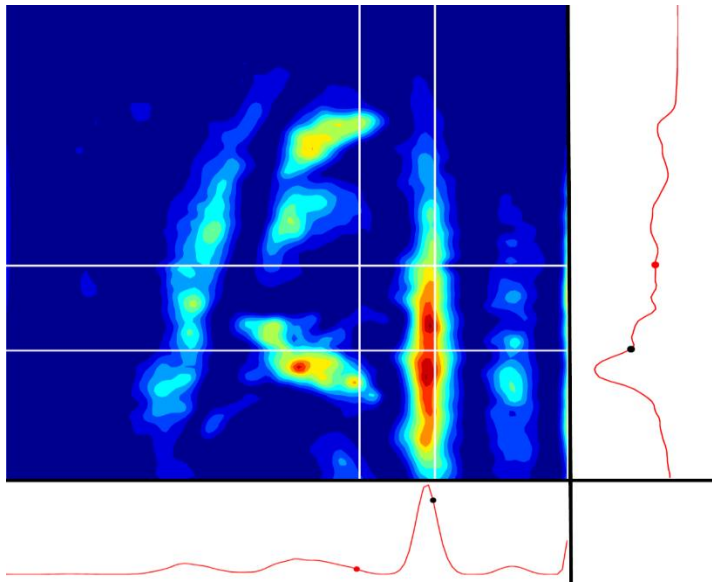
The first two iterations of KIB-FE method extended to surfaces are presented in the figures below. An initial fit with a 4 by 4 control point grid is made to the target DVF surface, and the error surface is computed (figure 61). From this error surface, the error profiles are obtained, and the algorithm proceeds as in the 1D case. First, the integral error in each segment is computed, and the segment with the maximum error is selected. A knot is then inserted into this segment such that segment is bisected into regions of equal integral error. Using the updated knot vectors, the surface is re-fit, producing a new error profile (figure 62). Next, the knots are then equilibrated according to the force equation (figure 63), and the algorithm returns to the knot insertion step. This is repeated for a fixed number of control points or until a specified tolerance is met.



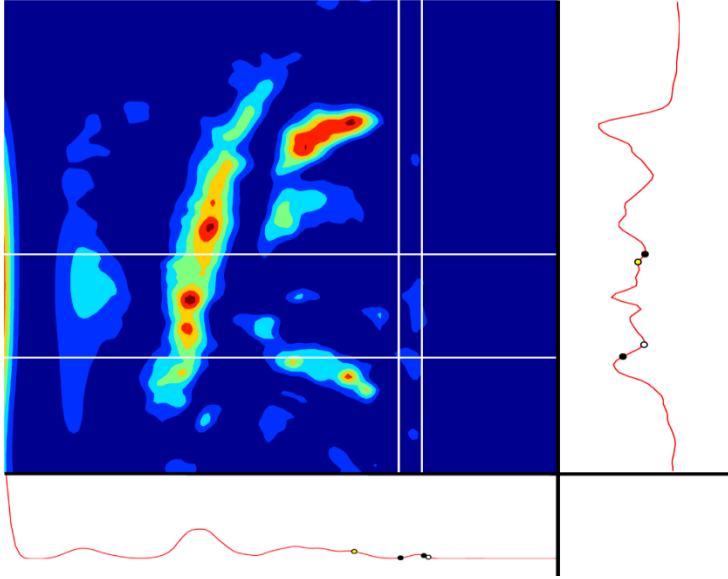
**Figure 61:** The first free knot in each direction is placed such that it bisects the integral error of the projected 1D error profile.



**Figure 62:** The target surface is re-fit with the new knot vector. The fitting error is updated, and the knot is allowed to move according to the force equation. Here the initial knot position is represented by the unfilled circle and the new knot position is represented by the solid circle.



**Figure 63:** New knots (in red) are again inserted according to the error bisection rule.



**Figure 64:** The target surface is once again re-fit, and the knots are allowed to move along the updated error profiles according to the force equation.

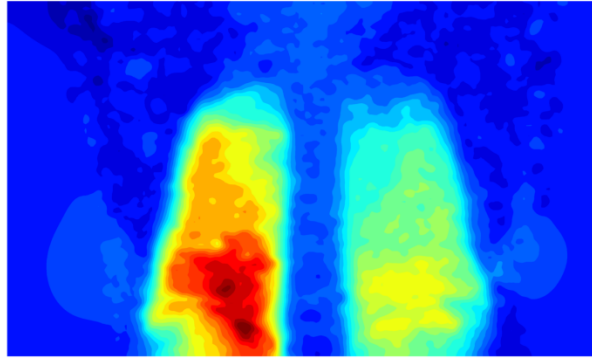
#### 4.2 DVF Surface Fitting with Non-Uniform B-Splines

To evaluate the heuristic knot placement scheme for tensor products developed in section 4.1, non-uniform B-splines were fitted to known DVF surfaces. These surfaces were generated by selecting a slice from a volume DVF, which was computed by the Demon's algorithm and made publically available by Vandemeulebroucke et al [2007]. Slices were chosen in the coronal, axial, and sagittal planes, and target surfaces were generated from the z-component of DVF in each slice. The surfaces exhibit the shape features which motivated the use of non-uniform B-splines, namely, areas of steep local gradients juxtaposed with regions of near-uniformity. Ideally, the knot distribution should be arranged such that knots are preferentially placed in the regions of complexity and away from the smoothly varying regions.

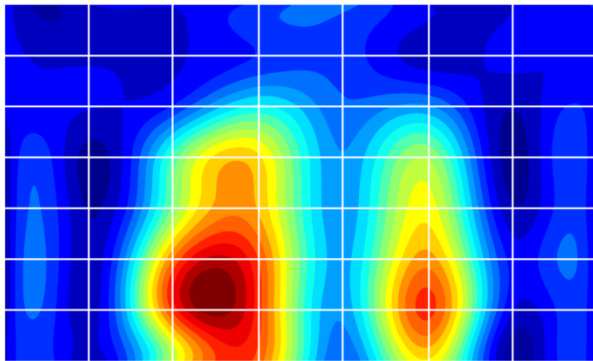
The non-uniform B-spline fits were compared to the uniform, and the Simplex method was once again taken as the gold standard. The top panel in figure 65 shows the target surface in

the coronal plane. In the panels below, the B-spline surfaces for each method are displayed. Each was computed with 6 free knots in each direction (shown as white lines), which corresponds to 100 control points. All contours were plotted on identical color maps. Looking at the uniform fit, one can see qualitatively that the lack of knot density in the regions of steep gradient resulted in an inability to faithfully represent those regions. In terms of the DVF, the uniform B-spline overestimated the motion. This problem is reduced in the non-uniform cases, where both the size and separation of the lung lobes are more accurate. Figure 66 shows the surface fits with 12 free knots in each direction (256 control points). With this many control points, the decrease in error seen with the non-uniform knots is not as visually dramatic, but the fine details in the high gradient areas--particularly in the (patient) right lung--are still better represented by the non-uniform methods. It is also worth noting that while the Simplex knot distribution differs vastly from the proposed knot insertion method, the fitting errors are nearly identical (see Figure 67). This suggests that the problem of B-spline fitting with knot optimization has many local minima.

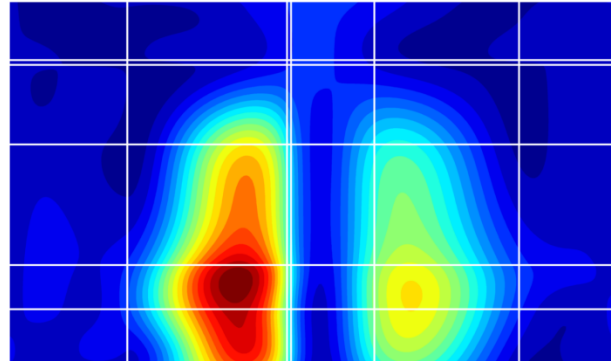
### Target Surface



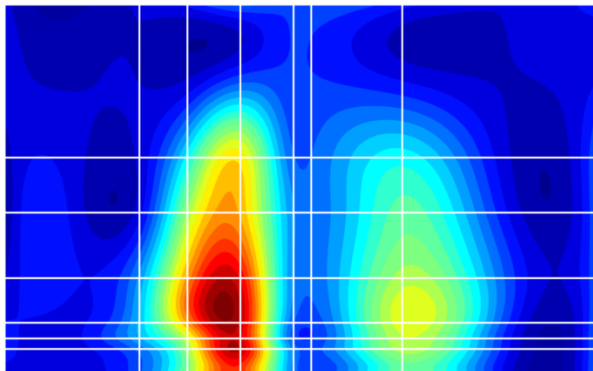
### Uniform Fit



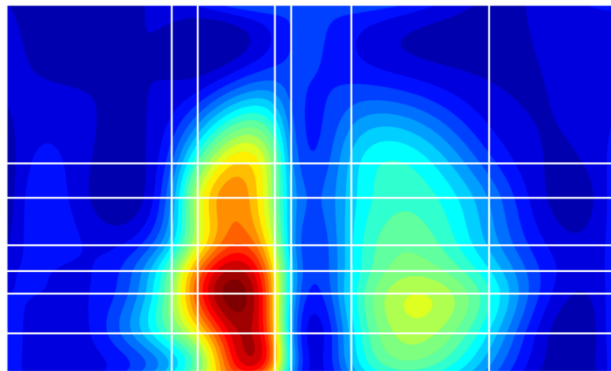
### Simplex Fit



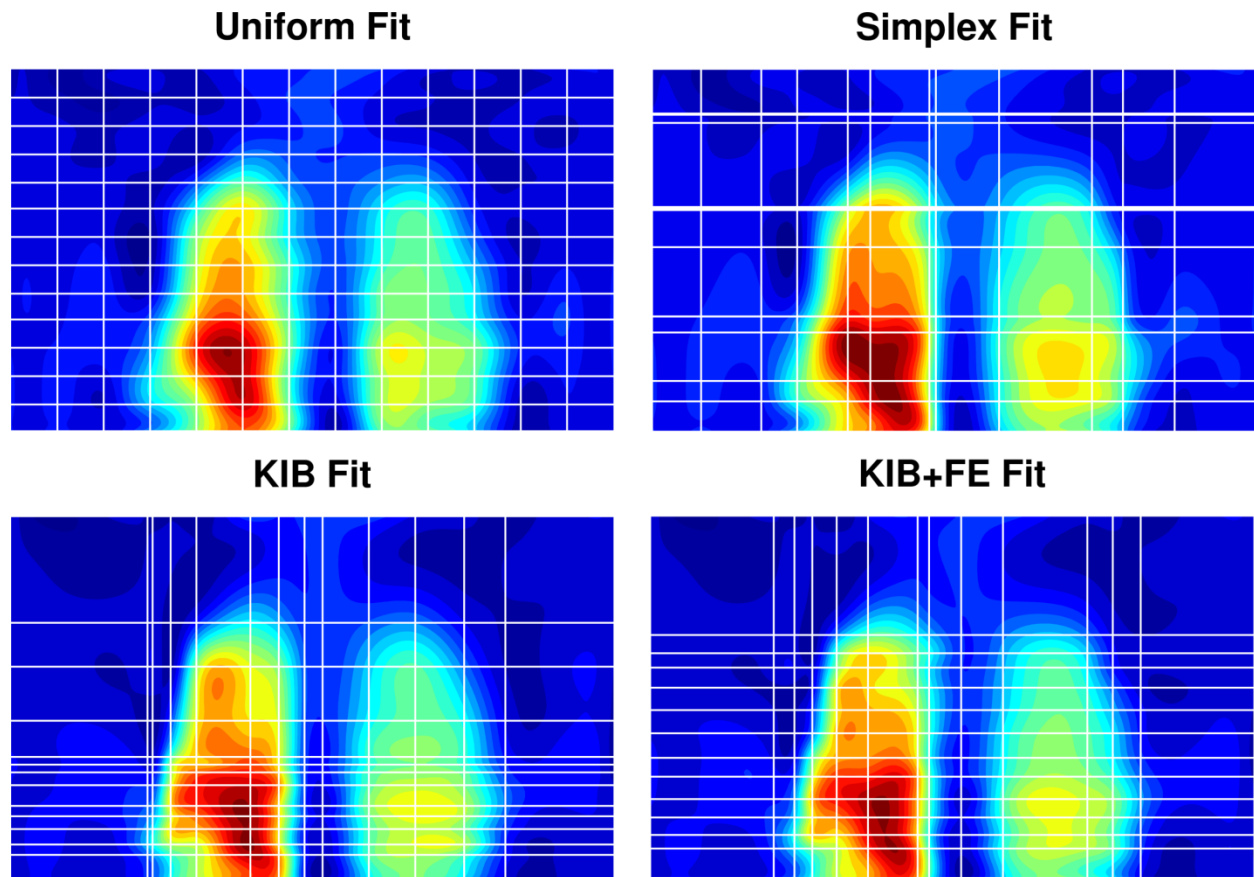
### KIB Fit



### KIB+FE Fit

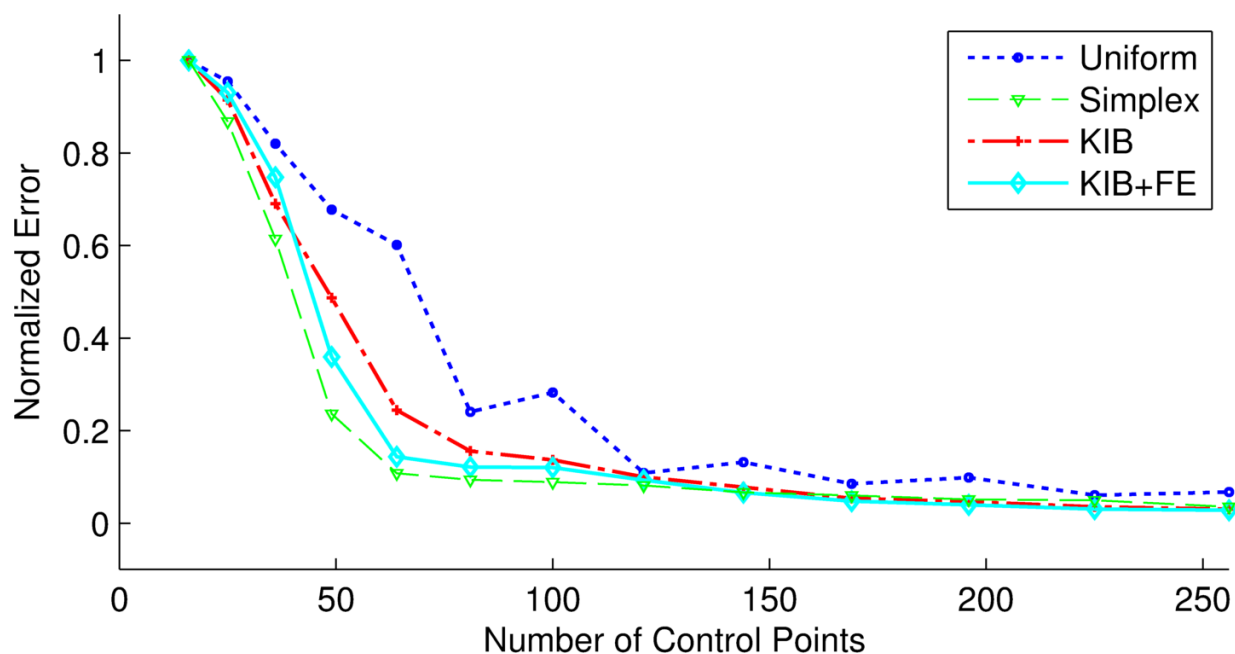


**Figure 65:** Contour plot of the target surface (top panel) generated from a coronal slice with a qualitative comparison of the four B-spline fitting methods using 100 control points.



**Figure 66:** Contour plots of the surfaces generated by the four B-spline fitting methods applied to a coronal slice of the DVF data using 256 control points.

### Coronal Fit Error Comparison

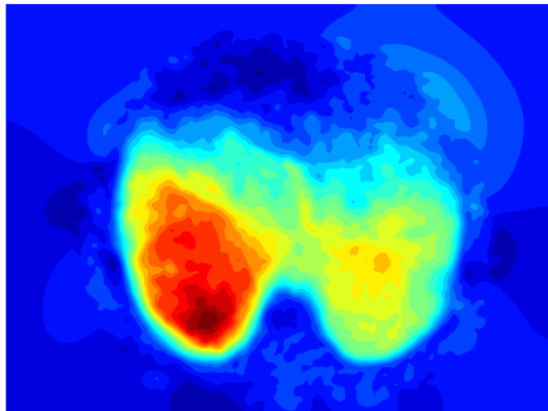


**Figure 67:** Comparison of fitting the error (sum of squared differences) to the coronal surface for each B-spline fitting method for a given number of control points. The errors are normalized to the fit with 16 control points (0 free knots).

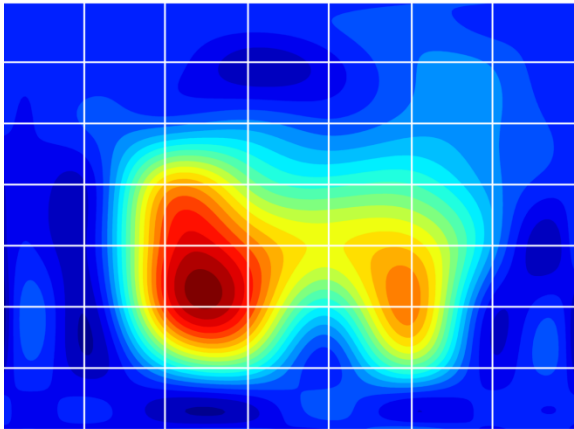
The surface fits of the axial (Figures 68 and 69) and sagittal (figures 70 and 71) slices exhibit the same general trends as the coronal example; however, though the non-uniform methods maintain a slight advantage over the uniform fits, the performance gap has shrunk. For example, in the coronal case the uniform fit required 121 control points to match or surpass the fitting error achieved with just 64 control points with non-uniform knots. Meanwhile, in the axial and sagittal cases the uniform fitting error nearly matches the non-uniform fits at 81 and 64 control points, respectively. Nevertheless, the uniform method never outperforms the non-uniform method for the same number of control points. It is worth mentioning that due to its inherent smoothness, the B-spline fit with a reasonable number of control points (i.e. much less than the number of data points) cannot match the noisy target DVF that has been generated by the Demons algorithm.



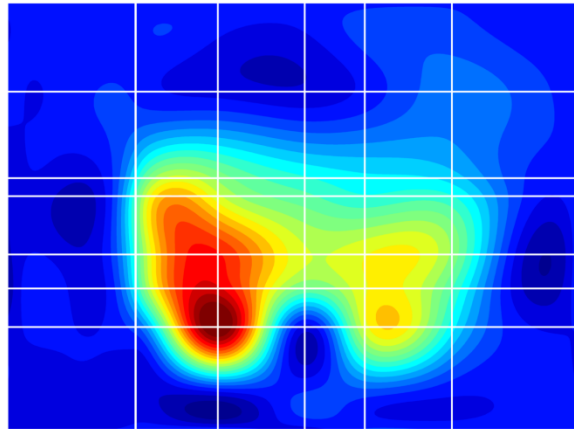
**Target Surface**



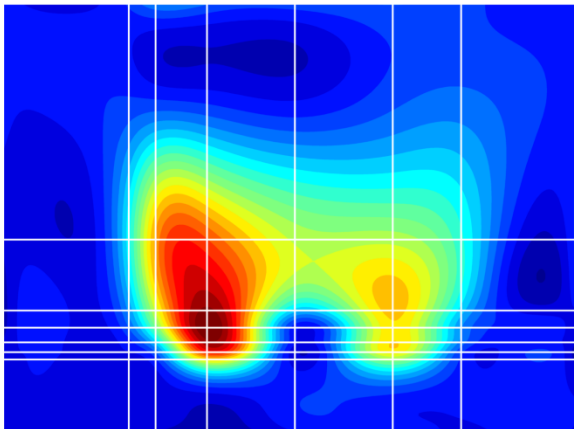
**Uniform Fit**



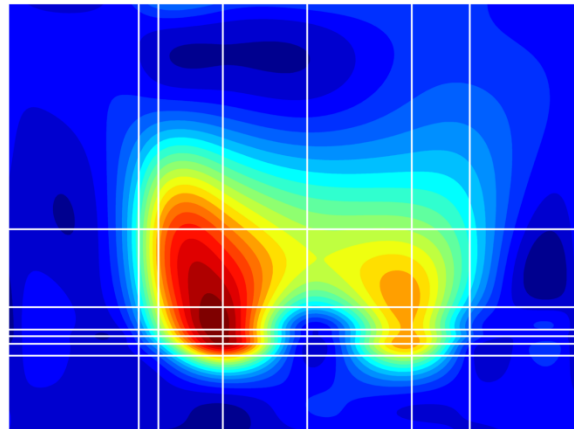
**Simplex Fit**



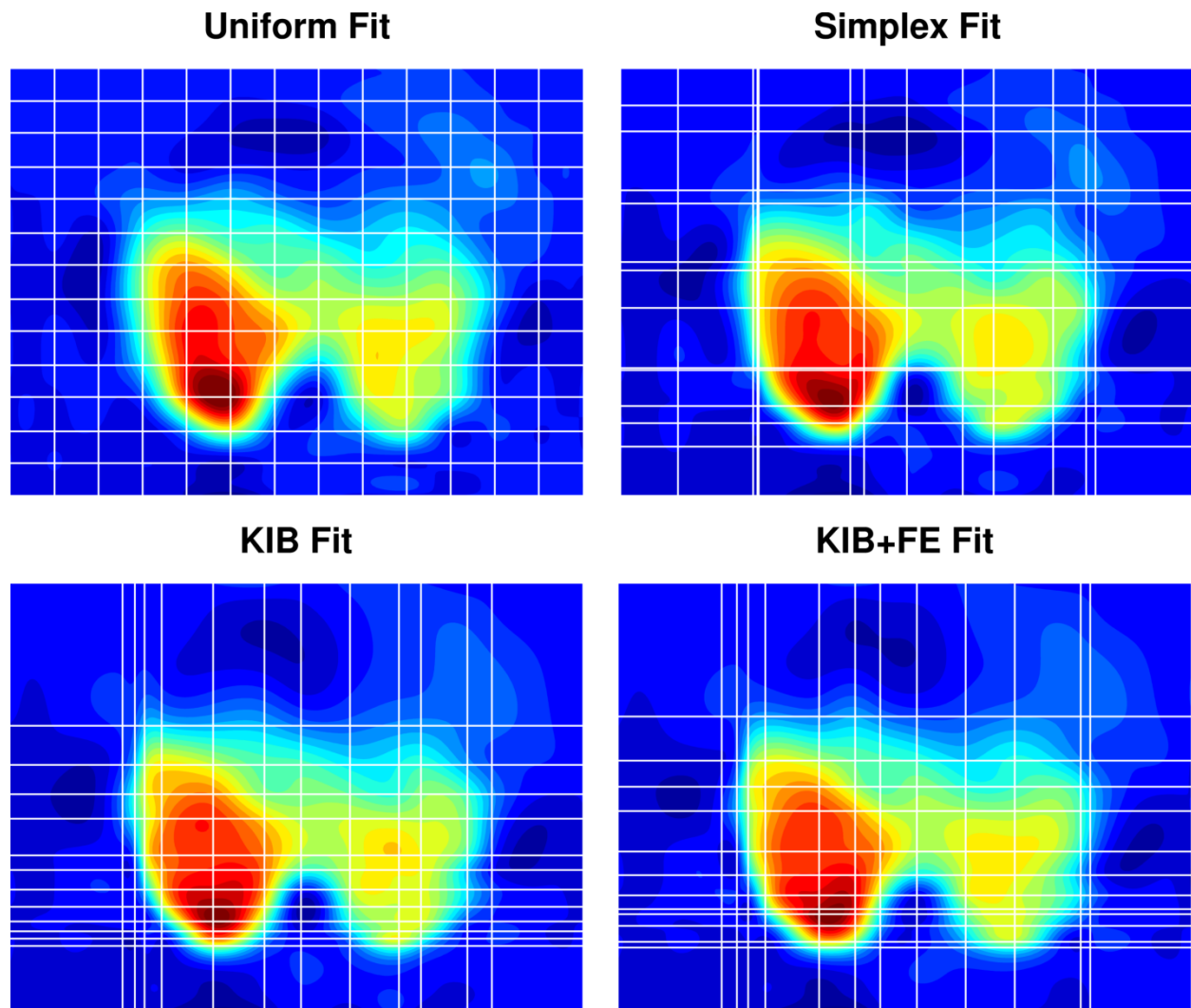
**KIB Fit**



**KIB+FE Fit**

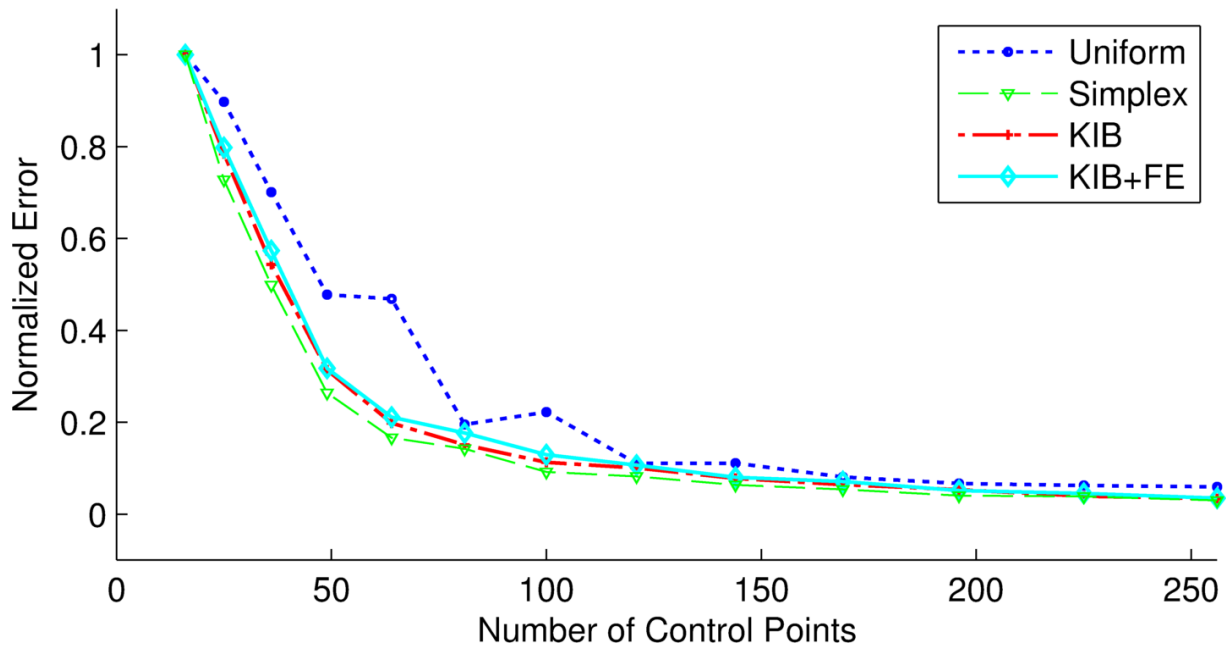


**Figure 68:** Contour plot of the target surface (top panel) generated from an axial slice of the DVF data with a comparison of the four B-spline fitting methods using 100 control points.



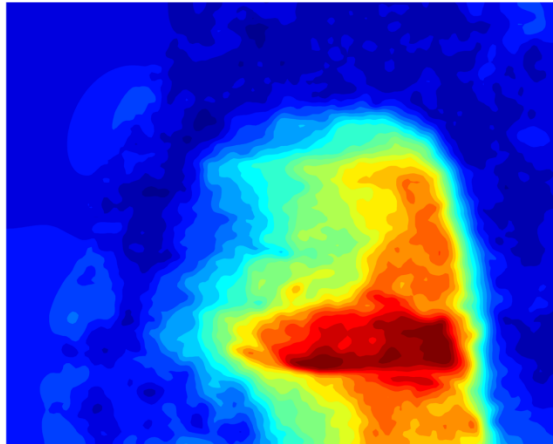
**Figure 69:** Contour plots of the surfaces generated by the four B-spline fitting methods applied to an axial slice of the DVF data using 256 control points.

Axial Fit Error Comparison

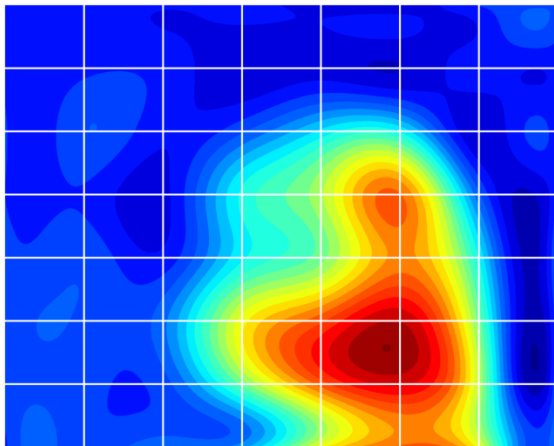


**Figure 70:** Comparison of fitting the error (sum of squared differences) to the axial surface for each B-spline fitting method for a given number of control points. The errors are normalized to the fit with 16 control points (0 free knots).

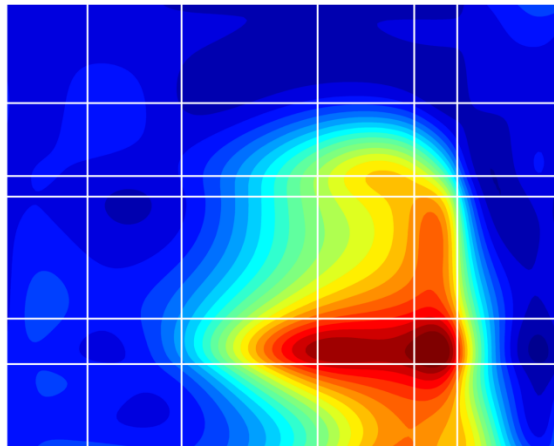
**Target Surface**



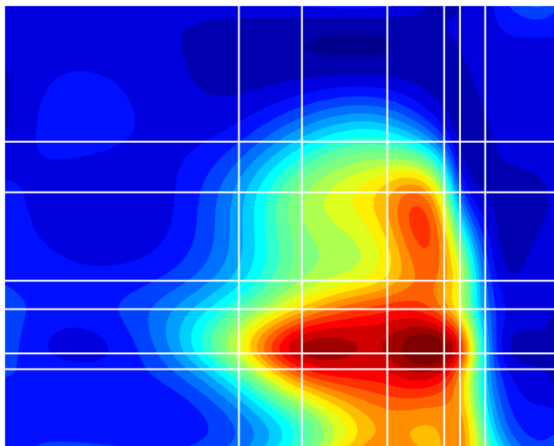
**Uniform Fit**



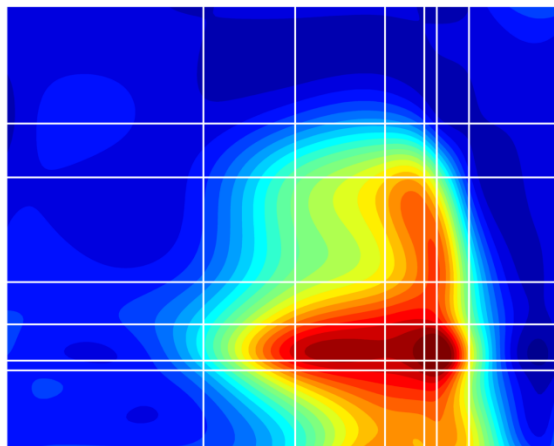
**Simplex Fit**



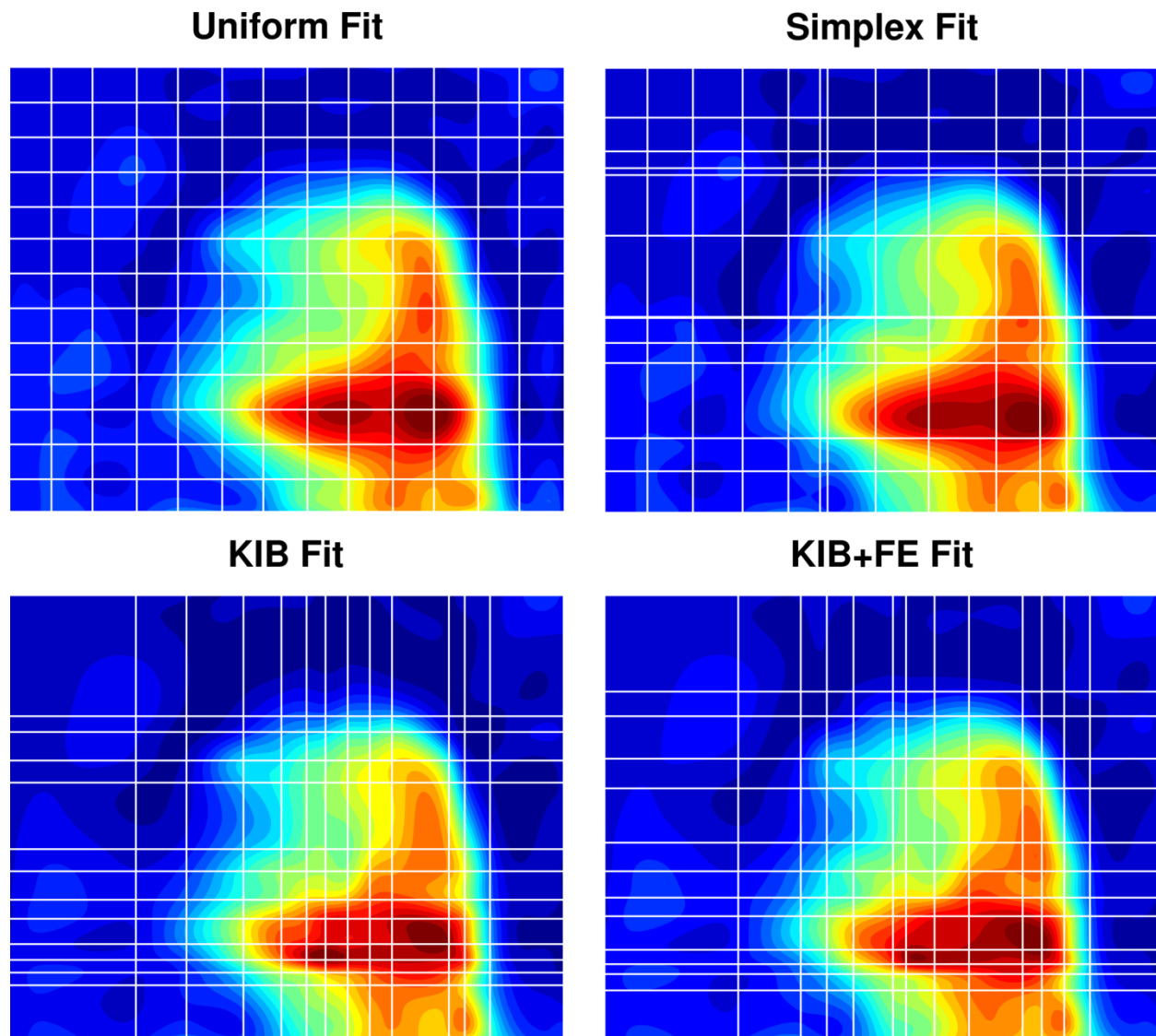
**KIB Fit**



**KIB+FE Fit**

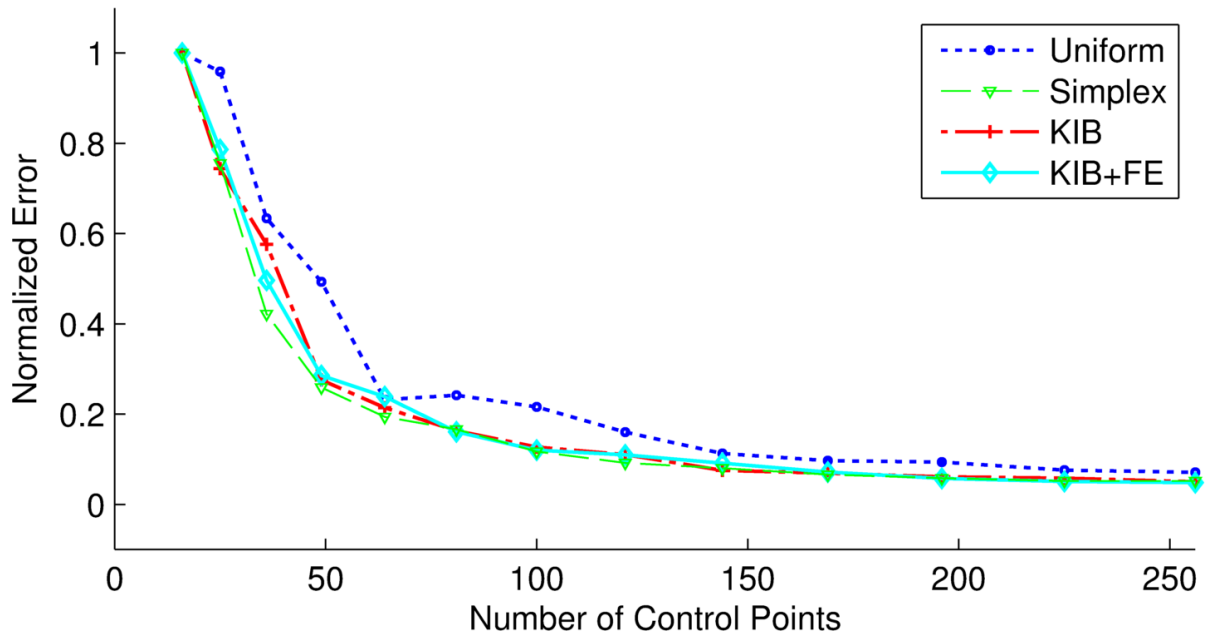


**Figure 71:** Contour plot of the target surface (top panel) generated from a sagittal slice of the DVF data with a comparison of the four B-spline fitting methods using 100 control points.



**Figure 72:** Contour plots of the surfaces generated by the four B-spline fitting methods applied to a sagittal slice of the DVF data using 256 control points.

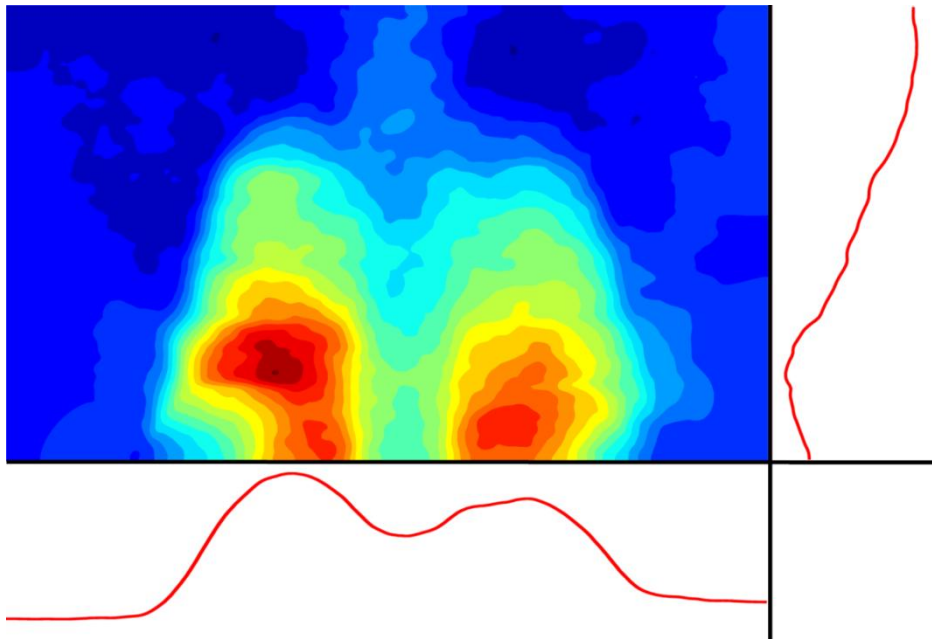
Sagittal Fit Error Comparison



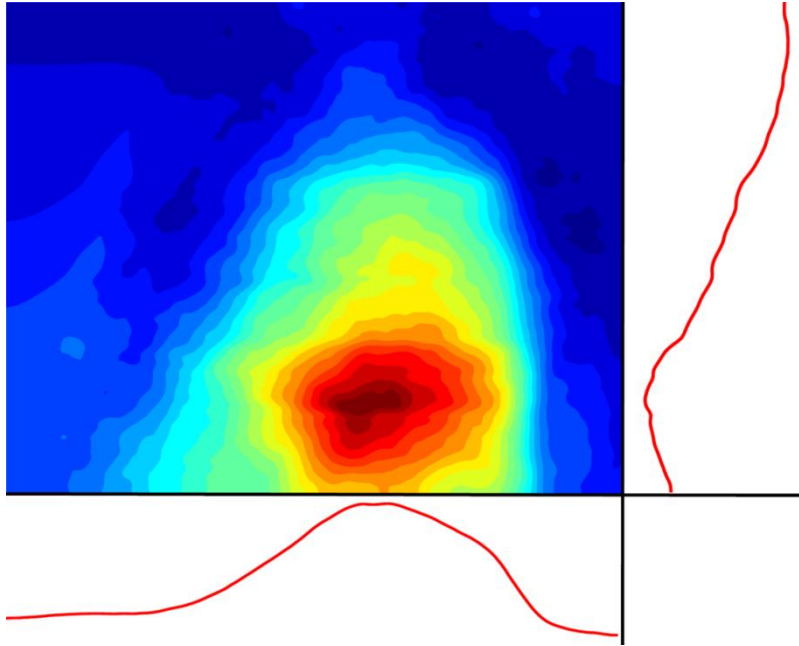
**Figure 73:** Comparison of fitting the error (sum of squared differences) to the sagittal surface for each B-spline fitting method for a given number of control points. The errors are normalized to the fit with 16 control points (0 free knots).

### 4.3 DVF Volume Fitting with Non-Uniform B-Splines

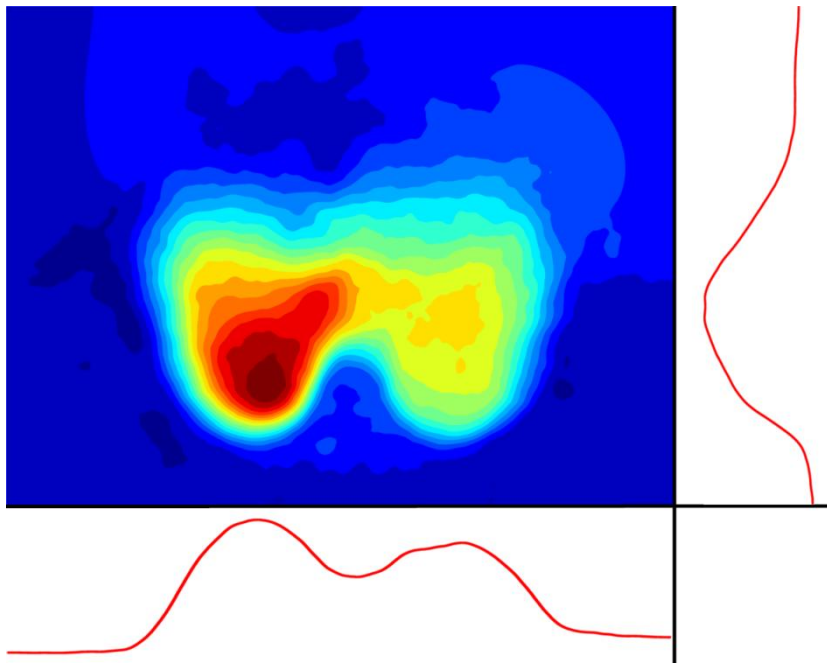
Having demonstrated the advantage of non-uniform knots in B-spline fits to surfaces in the previous section, the next step is to extend the heuristic knot placement rules to fitting volumes. To accomplish this, the slices in the volume data were averaged in three directions, producing an average coronal surface (Figure 74), an average sagittal surface (Figure 75), and an average axial surface (Figure 76). From these surfaces, 1D profiles were obtained, and the fitting routines were applied as before. The simplex method was abandoned for this volume fitting test because it was found to not converge reliably.



**Figure 74:** Contour plot of the average coronal slice.



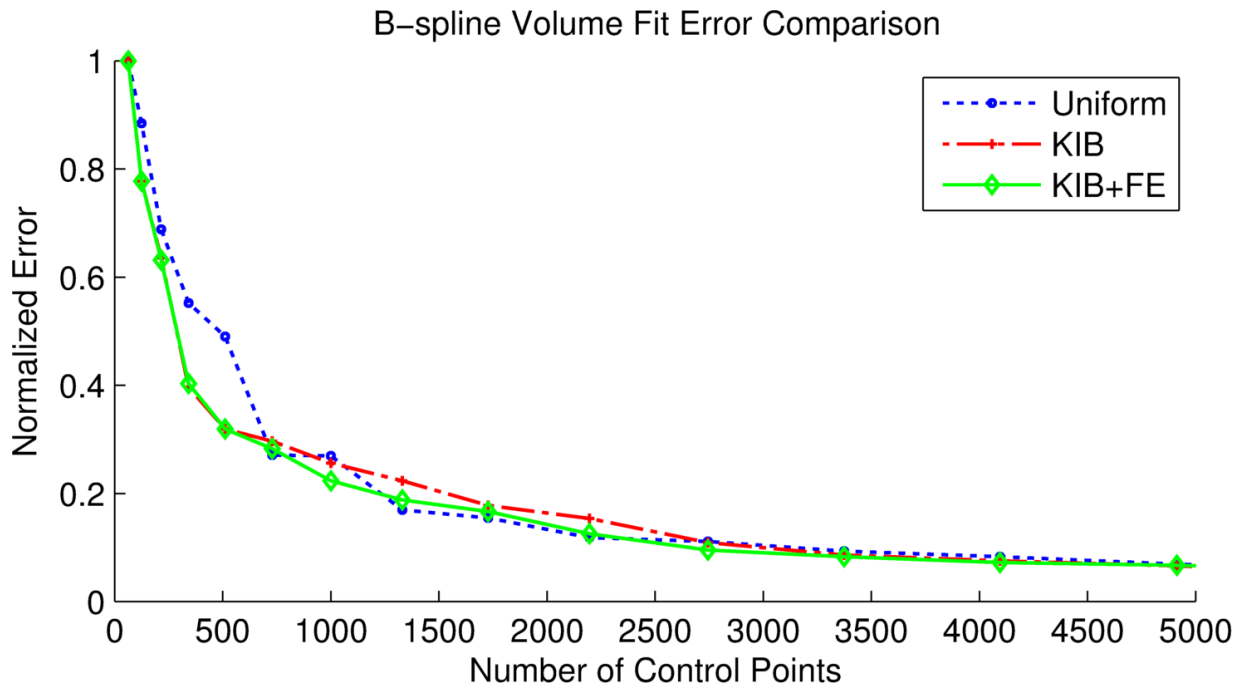
**Figure 75:** Contour plot of the average sagittal slice



**Figure 76:** Contour plot of the average axial slice

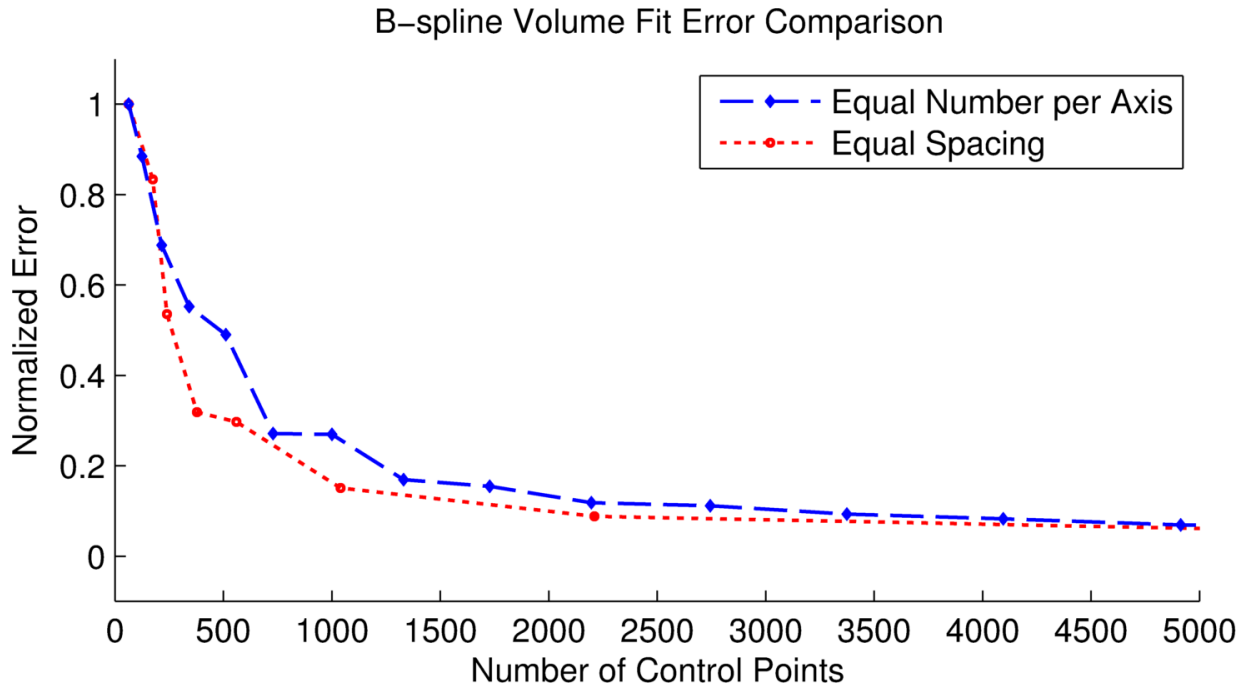


The error comparison for the non-uniform and uniform methods is shown in Figure 77. The advantage of the non-uniform methods is still present for sparse control points, however as the number of control points increases, the modest gains are quickly diminished and soon nonexistent. In fact, the uniform knot distribution outperforms the non-uniform for certain number of control points. One reason for this is that the 1D error profiles are not a good surrogate for the overall 3D error. In the 1D case, a knot can be added where it is needed on an individual basis. However, because of the rectilinear knot structure created by the tensor product formalism of B-splines, adding a single knot is not possible in higher dimensions. Instead, one must add strings of knots that are propagated in all directions. This results in redundant knots, and also overcompensates for regions of maximum error at the expense of regions of slightly lower error. The same trend is evident in moving from the 1D case to the 2D case, and is exacerbated when extending to 3D. The confinement of the knots to a rectilinear grid diminishes the gains in shape control. Nevertheless, the advantage is still evident for a small number of control points, which suggests that the 1D profiles may be useful for initializing grid spacing in the DIR process.



**Figure 77:** Comparison of the sum of squared differences between each B-spline fitting method and the z-component DVF volume. The errors are normalized to the fit with 16 control points (0 free knots).

A general shortcoming of these non-uniform B-spline fitting algorithms when applied to higher dimensions is that they insert an equal number of knots in each direction. On the other hand, the common practice for DIR with uniform knot vectors is to distribute knots with equal spacing. This results in the number of knots in each dimension being proportional to the number of voxels in that dimension. Figure 78 compares the fitting accuracy of a uniform B-spline with equal number of knots per axis and a uniform B-spline with equal spacing between each knot. There is a distinction between the two methods when the number of voxels is not the same in each direction. In this case, the data set is 235 by 176 by 141, and equal spacing produces a smaller error. This observation will be considered in section 5.2.



**Figure 78:** Comparison of the sum of squared differences of a uniform B-spline with equal knot spacing in each axis and a uniform B-spline with an equal number of knots per axis. Equal knot spacing is conventional method, and it is seen here to be superior to the knot distribution with an equal number of knots per axis.

#### 4.4 Summary of Investigation

Chapter 4 extended the 1D non-uniform knot placement rules to tensor product B-spline fitting. This was accomplished by projecting the multidimensional error mapping into a pair or set of 1D error profiles. Optimization of the knots in this way was observed to offer increased B-spline fitting accuracy on a per control point basis. However, the relative benefit decreased as the dimensions increased. One reason for this is that a single knot cannot be added to a 2D or 3D B-spline knot grid without propagating the knot addition in every direction, so adding knots for local shape control impacted the number of knots globally. Another reason for the decreased benefit is that B-splines are by definition a linear combination of basis functions, and while the value of a point on a B-spline curve depends on just four control points and their associated basis

functions, a 3D tensor product B-spline depends on sixty four control points and their associated basis functions. Changing the shape of the basis functions through manipulation of the knots will have a larger impact on shape control when there are fewer control points to optimize.

## 5. Non-Uniform B-Spline Deformable Image Registration

The transition from B-spline fitting to B-spline DIR introduces a new level of abstraction. Whereas in the fitting process the changes in the free parameters directly influence the cost function, the DIR process requires an intermediate step. The general DIR method uses a DVF  $u(x)$  to map points  $x' = x + u(x)$  in a source image  $A(x')$  to points  $x$  in a target image  $B(x)$  such that  $A(x + u(x)) \approx B(x)$ . In B-spline DIR, the DVF is represented by the B-spline function  $u(x) = \sum_i c_i b_{i,p}(x)$ , and the coefficients are optimized such that the sum of the squared pixel intensities differences between the source and target images is minimized:

$$\varepsilon^2 = \frac{1}{2} \sum_i \left[ I_A \left( x_i + \sum_j c_j b_j(x_i) \right) - I_B(x_i) \right]^2 \quad (5.1)$$

The first term in the bracketed difference is often referred to as the test image. Because it is a function of the free parameters of the B-spline, an additional partial derivative must be computed in the optimization routine:

$$\frac{\partial \varepsilon^2}{\partial c_j} = \sum_i \left[ \left[ I_A \left( x_i + \sum_j c_j b_j(x_i) \right) - I_B(x_i) \right] \frac{\partial I_A}{\partial c_j} b_j(x_i) \right] \quad (5.2)$$

A second consequence of the added level of abstraction is the inability to directly compare the current estimate of the DVF and the true underlying DVF which minimizes equation 5.1. For this reason, the heuristic rules for knot placement must again be adapted. Section 5.1 discusses the

methods used to incorporate non-uniform B-splines into 2D DIR and investigates the added benefits. Section 5.2 further adapts the methods to the full 3D DIR case.

## 5.1 2D DIR

Typical uniform B-spline DIR methods follow hierarchical steps, during which the control point grid (and the corresponding spatial resolution of the DVF) is repeatedly refined by adding knots and control points throughout the volume, which reduces the uniform grid spacing. This step is performed without regard to the spatial distribution of the errors in the previous estimate of the DVF. Instead, knowledge of this spatial distribution of the errors can be used to add new control points predominately to areas of registration mismatch. In intensity-based DIR, the control point values are optimized such that they minimize, e.g., the sum of the squared differences (SSD) of pixel intensities between the two images. This means that at each hierarchical step, the current difference image,  $D(x) = A(x+u(x))-B(x)$ , can be computed by subtracting the intensity values at each corresponding pixel in the two images. The difference image serves as an error map and gives insight into the spatial distribution of the local details in the underlying deformation. This information can be used to drive the placement of new control points.

The proposed algorithm begins by registering the two images using a coarse, uniform grid. Whereas the B-spline fitting methods used the error of fit to determine the knot placement, the difference image from the initial uniform registration is now used to add control points non-uniformly in areas with the largest error. It was observed in Chapter 3 that simply adding control points to the maxima of the difference image produced poor results. Instead, the pixel intensities

are summed across the difference image in each direction to obtain two one-dimensional difference profiles. Each difference profile is sub-divided into segments delineated by the grid locations (i.e. the difference profile in the x direction is divided by the y-axis grids, and vice versa). The intensity differences in each segment are summed, and a control point is added to bisect the segment with the largest total intensity difference. This step is analogous to the KIB method in B-spline fitting. The images are then registered with the updated grid configuration, thus producing an updated difference image. The registration step optimizes the control points via a Limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) algorithm. The maximum iterations are set to 500, and the minimum step size and functional change are set to 1e-6.

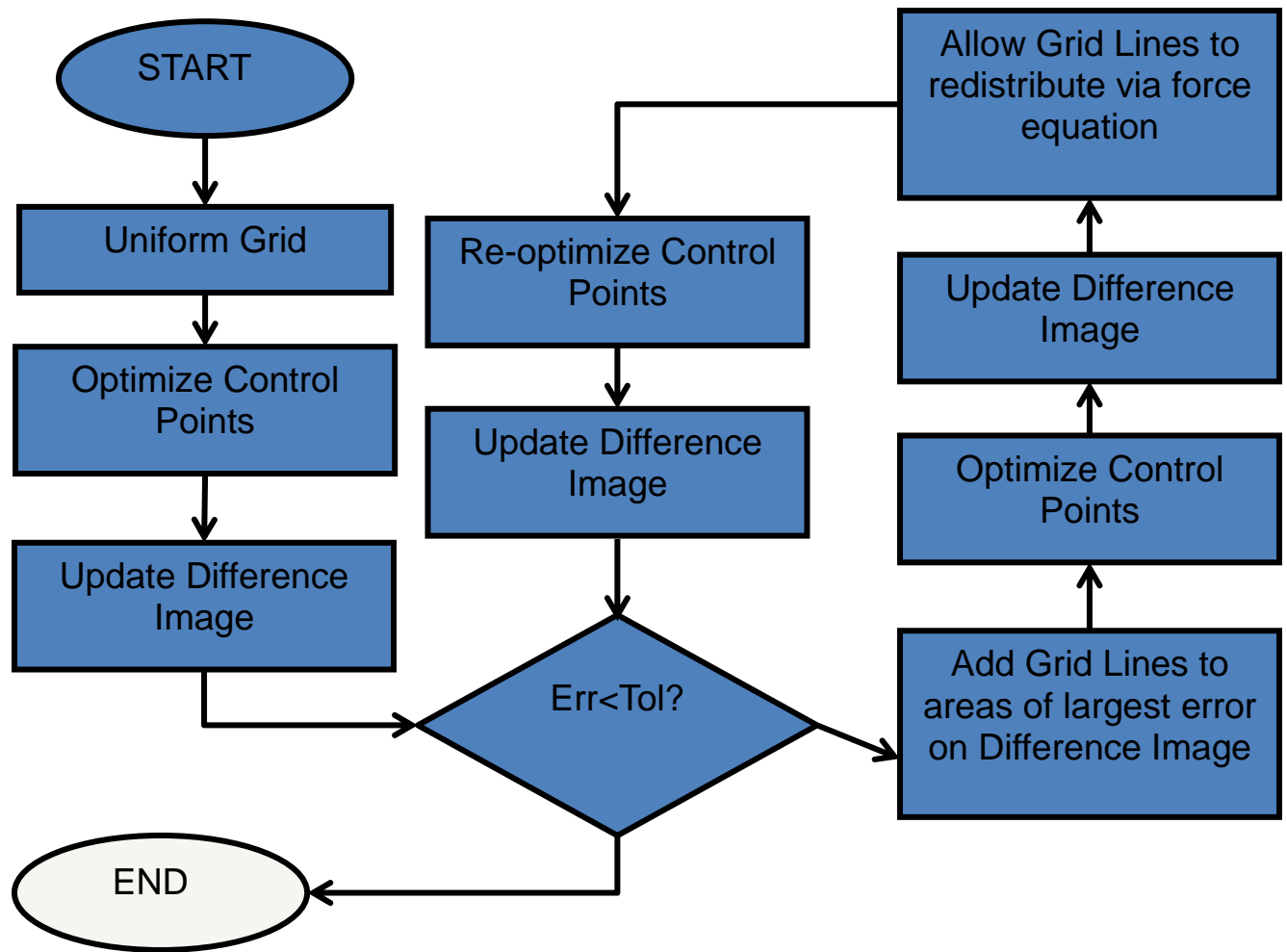
The control point grids are then further refined by allowing the grid lines to migrate according to an attractive force proportional to the local intensity differences (i.e., errors). The one-dimensional difference profiles are again computed for this purpose. The knots, which correspond to the grid lines, move along the difference profile towards regions of large error and away from regions of comparatively small error. To ensure that the algorithm does not overcompensate in regions of large error by moving knots too far from their original locations, a restoring spring-force counteracts the attractive force. Optimization of the knot vector is computed in the same way as it was for the KIB+FE method in B-spline fitting: for each knot  $i$ , equation 5.3 is solved for the knot position  $k_i$  that minimizes the sum of the contributions of each point of the 1D squared error function  $\varepsilon^2(x)$  contained in the knot segment belonging to  $k_i$ :

$$\mathbf{k}_{min} = \arg \min_{\mathbf{k}} \int_{x=k_{i-1}}^{x=k_{i+1}} |x - k_i| \cdot \varepsilon^2(x) dx \quad (5.3)$$

This balancing of forces avoids having too many knots gravitate towards one region, leaving too few in another. The updated knot locations are determined by minimizing this equilibrant system using the downhill simplex method. It is important to note that performing optimization of the knot vectors in this way is a 1D problem which must be performed twice for 2D registration and three times for 3D registration. This introduces computational overhead to the registration process; however, because there are far fewer free parameters, and function evaluations of this problem are fast, its impact on total registration time is minimal in comparison to the time taken to optimize the control point values.

The non-uniform hierarchical grid refinement process is repeated until the SSD is below a pre-defined error tolerance. Figure 79 displays the flowchart of this scheme.



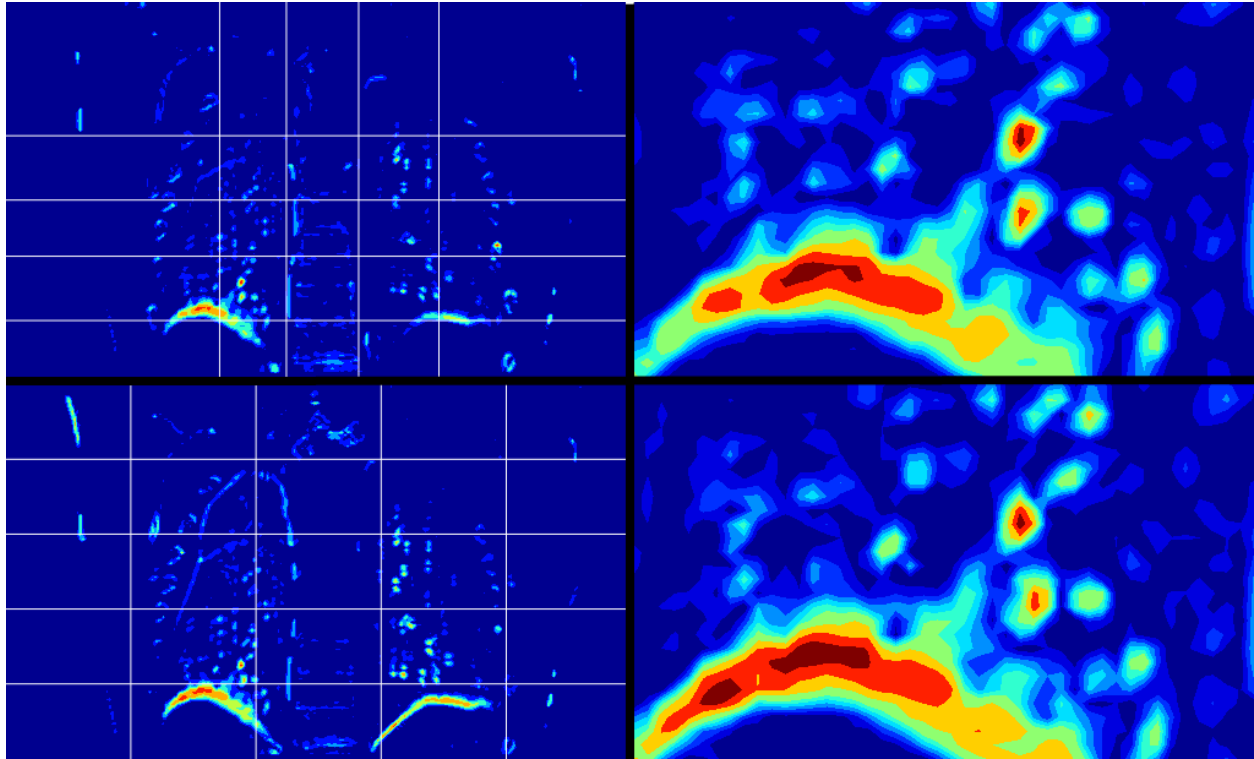


**Figure 79:** Flowchart of the non-uniform refinement algorithm

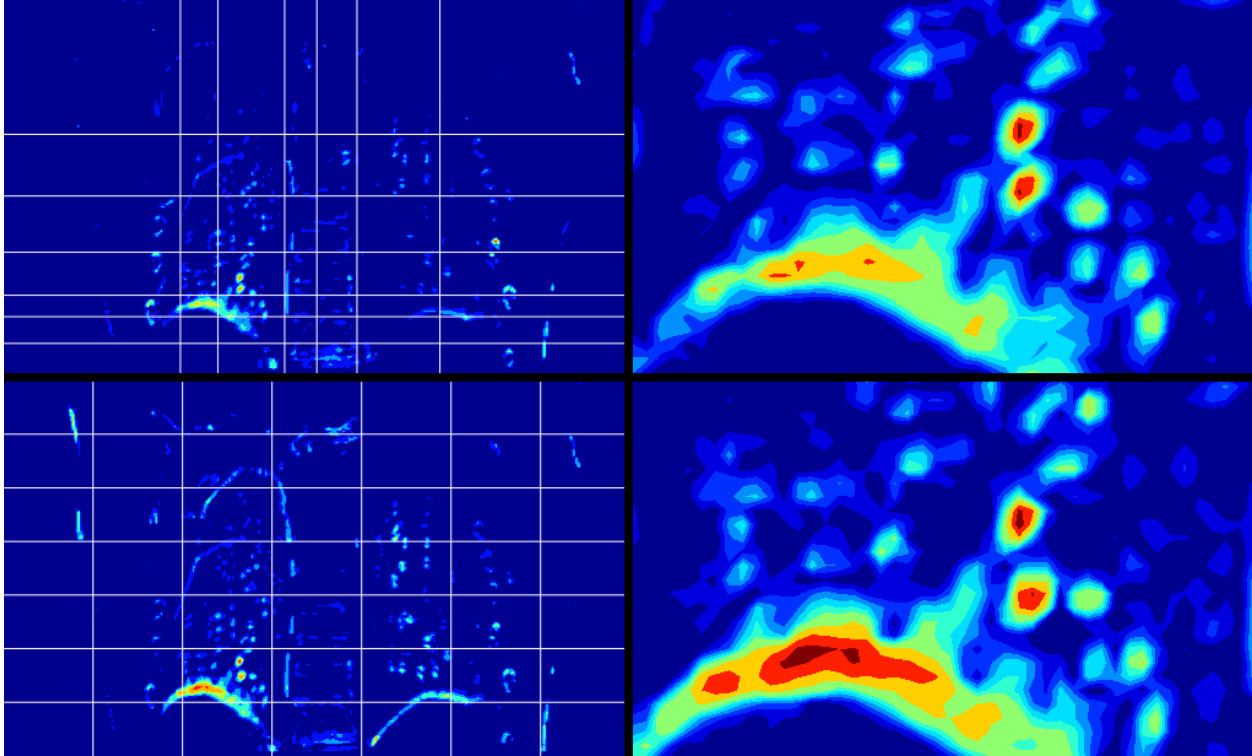
For the purpose of this study, a comparison is made between registration results of non-uniform and uniform two-dimensional control point grid sizes ranging from 4 by 4 to 20 by 20 (8 by 8 to 24 by 24 knots). To ensure a fair comparison, the uniform grid registration was obtained using a refinement scheme that adds two control point grid lines (one in each direction) for each hierarchical step. The source image was a coronal slice of a thoracic CT taken from the POPI dataset. The target image was obtained by applying a known DVF to the source image. Specifically, a slice from the Demon’s-derived DVF of the POPI dataset was used.

A qualitative comparison of the uniform and non-uniform registration results is provided in Figures 80-83. Each figure represents a snapshot of the results of the two registration algorithms for a particular number of control points. In each frame, color-mapped plots of the difference images resulting from the non-uniform and uniform registrations are displayed on the top- and bottom-left, respectively. On the right is a magnified view of the lower right lobe and diaphragm, which we designated as the region of interest (ROI). This region was selected because it encompasses an area where we can expect the largest deformation and steep motion gradients. The position of control point grids are indicated by white lines. Notice that the non-uniform refinement algorithm produces a denser distribution of control points in the lungs and along the diaphragm. Denser control points allow for the B-splines to more closely model the sharp motion gradients present in those regions. By sacrificing control point density in static regions (where having a large number of control points is not advantageous), the non-uniform refinement algorithm is able to automatically reduce the large local registration errors present near the diaphragm while maintaining good accuracy elsewhere. This is shown quantitatively in Figure 84, which compares the number of control points required by the two algorithms to resolve the large local errors present in the ROI after the initial coarse-grid registration. The ability of the non-uniform grid to achieve denser control points in regions of large local error without increasing the number of control points globally results in better registration accuracy with fewer control points. This is most clearly seen in regions where coarse uniform grids are unable to model the sharp local gradients, as exemplified by the ROI. The reduction of error in the ROI occurs with fewer control points using the non-uniform control point grid because control points are more optimally distributed with respect to the underlying motion. It is also evident that additional control points have a diminishing return on the error: the non-uniform

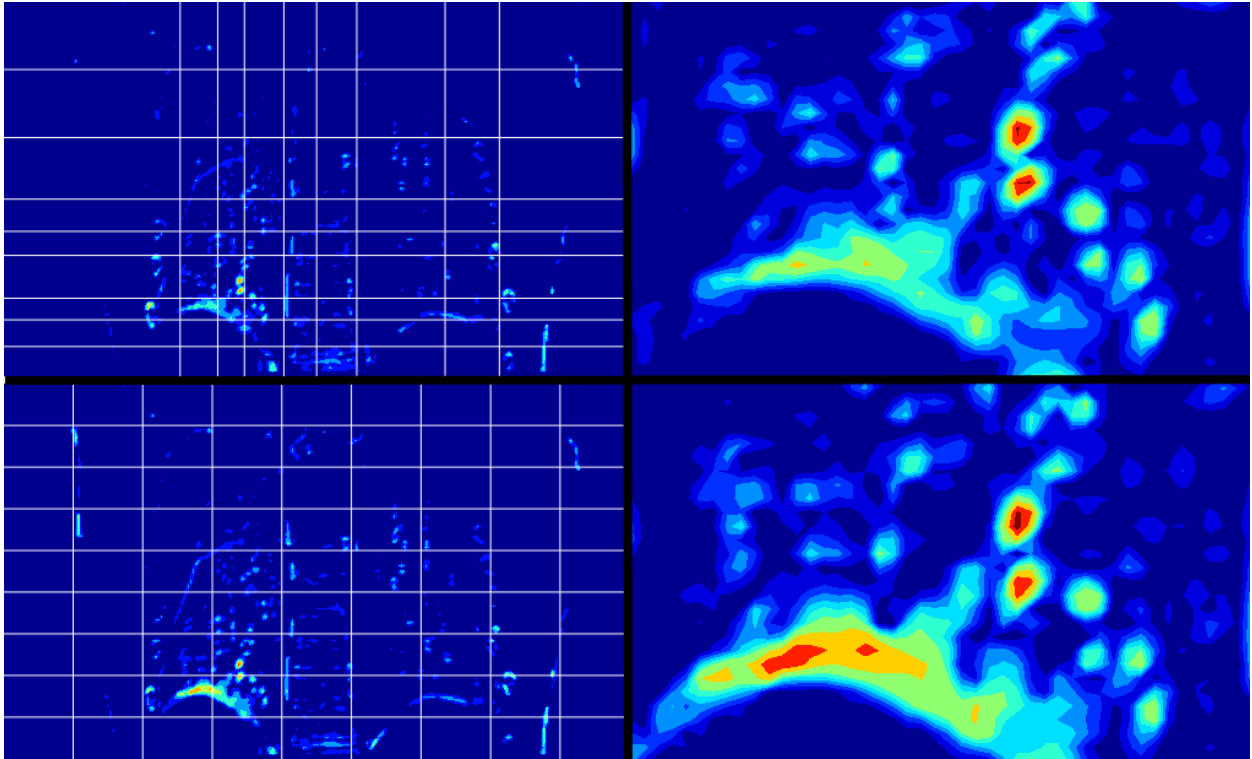
registration with a 14 by 14 grid produced a smaller error in the region of interest than the uniform registration with a 40 by 40 grid.



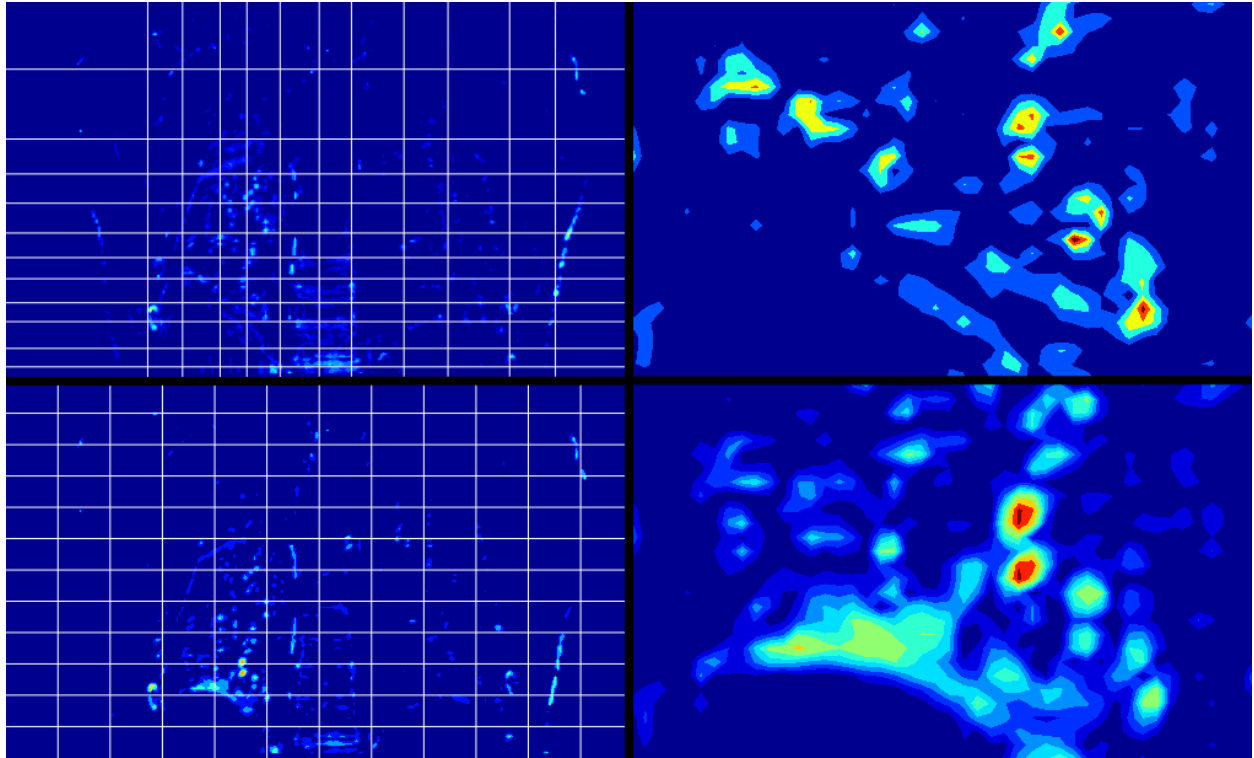
**Figure 80:** Comparison of the non-uniform (upper) and uniform (lower) registration errors in the lung with an 8 by 8 control point grid.



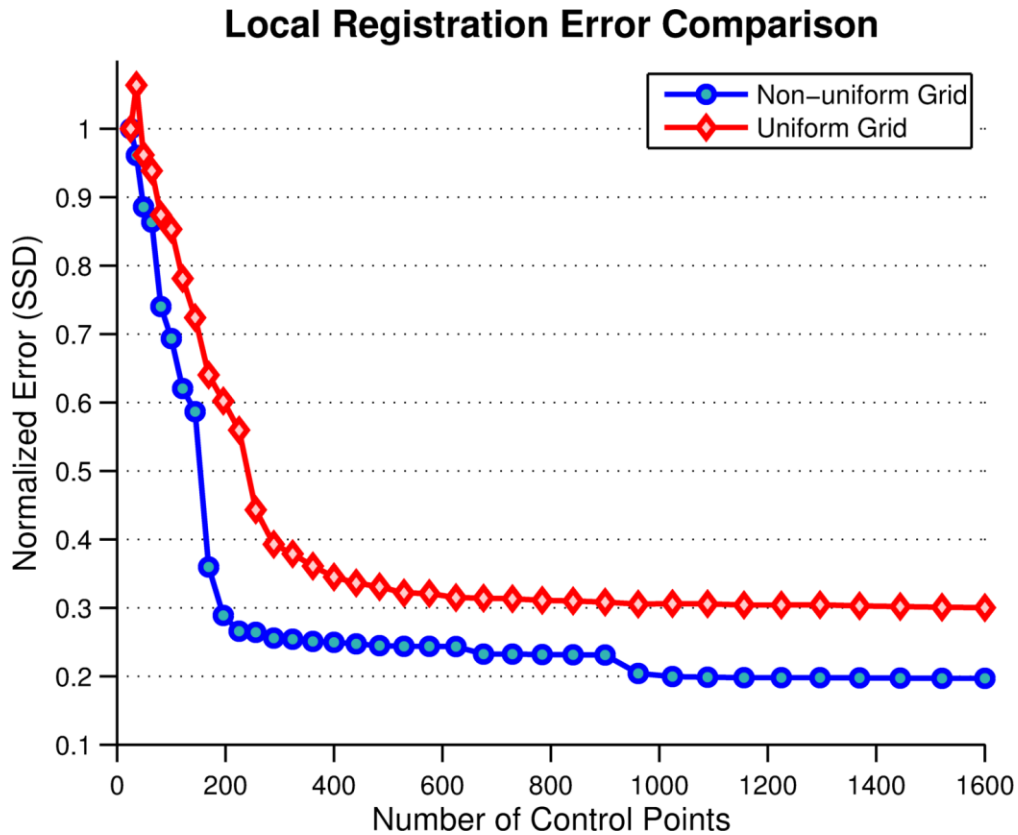
**Figure 81:** Comparison of the non-uniform (upper) and uniform (lower) registration errors with a 10 by 10 control point grid.



**Figure 82:** Comparison of non-uniform (upper) and uniform (lower) registration errors with 12 by 12 control point grid



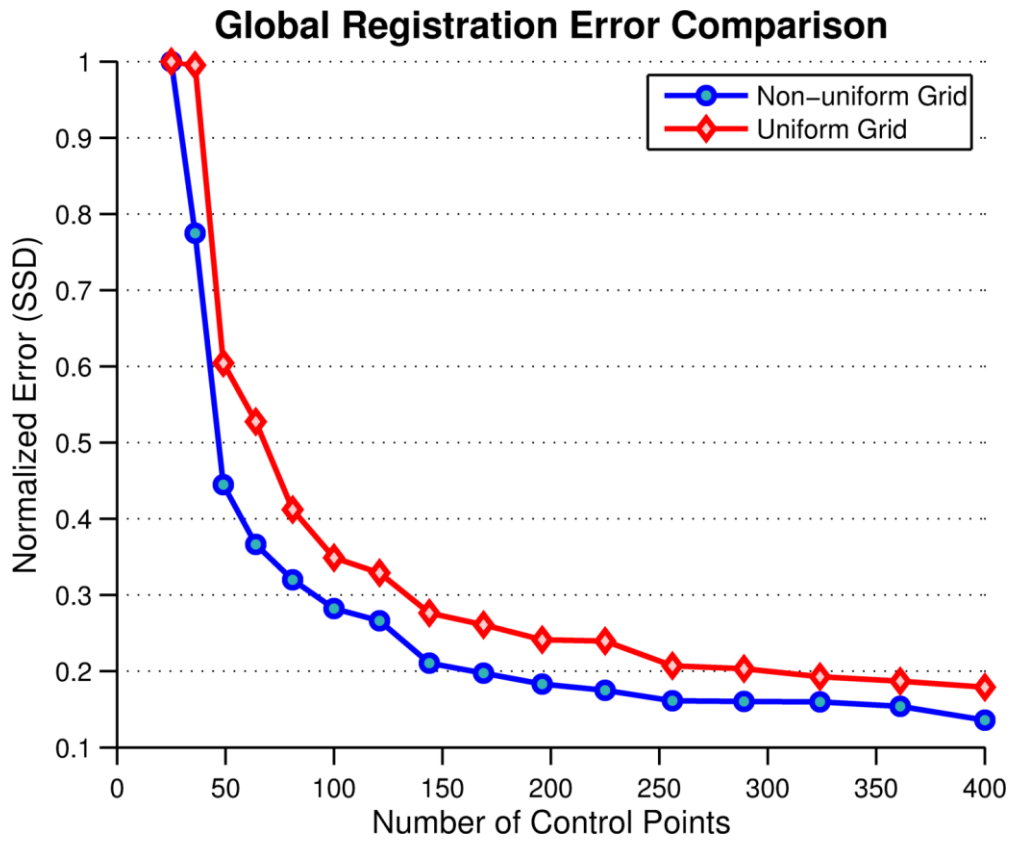
**Figure 83:** Comparison of non-uniform (upper) and uniform (lower) registration errors with 15 by 15 control point grid



**Figure 84:** Non-uniform refinement results in a reduction of large local errors in the ROI.

The same general trend was observed for the global error, as shown in Figure 85.

Again, the diminishing return on registration error begins with the 14 by 14 non-uniform control point grid and the uniform grid does not surpass that accuracy until 20 by 20. This confirms that preferentially placing control points in regions of anatomical motion and away from static regions (where they would be redundant) decreases large local errors without adversely affecting the accuracy of the global registration.



**Figure 85:** Non-uniform refinement results in an overall reduction of registration error.

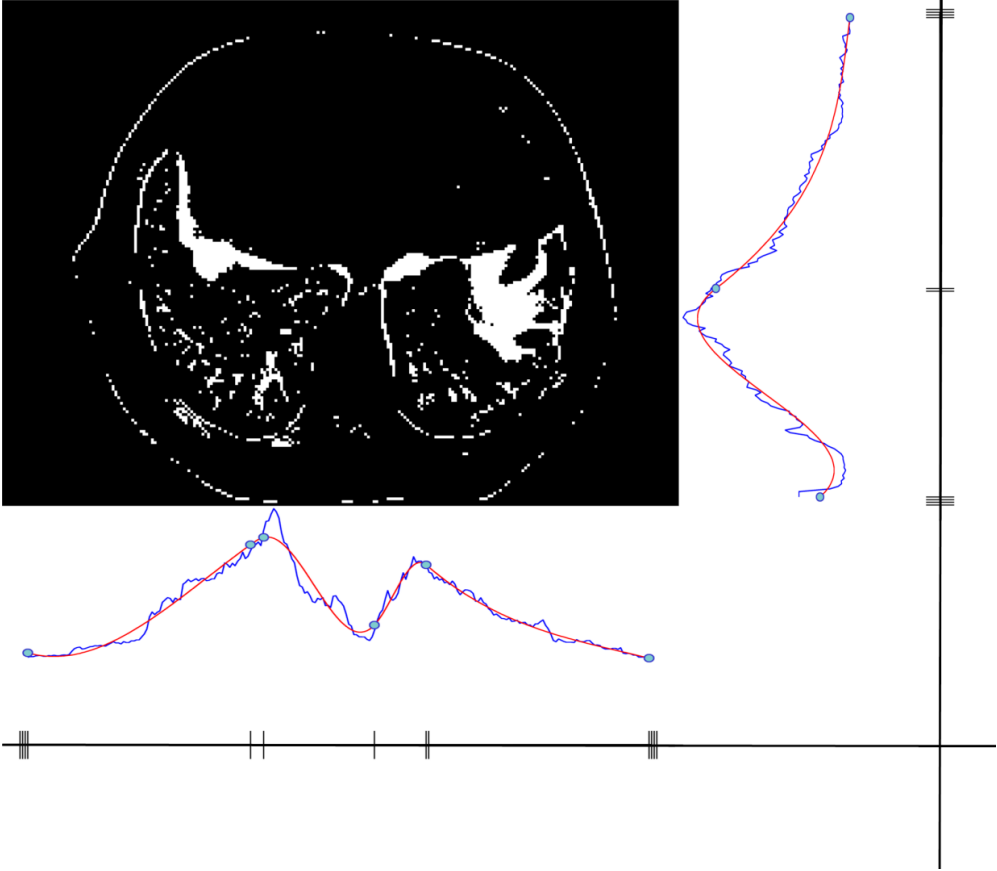
## 5.2 3D DIR

The previous section demonstrated that the non-uniform B-spline fitting methods can be integrated into the 2D DIR process in order to decrease registration errors. This section builds upon that proof of concept by modifying the algorithm to the 3D case. It was observed in Section 4.3 that the straightforward extension of the non-uniform knot placement methods from 2D to 3D resulted in a modest decrease in fitting error as compared to the uniform method. In essence, condensing a 3D error to three 1D error profiles obscured the true 3D distribution of errors. Similarly, the restriction of the rectilinear knot grid inhibited the ability of the knots to adequately mimic the 3D error distribution. These limitations, as well as other lessons learned in Chapter 4, were taken into consideration when developing the 3D non-uniform B-spline DIR method.

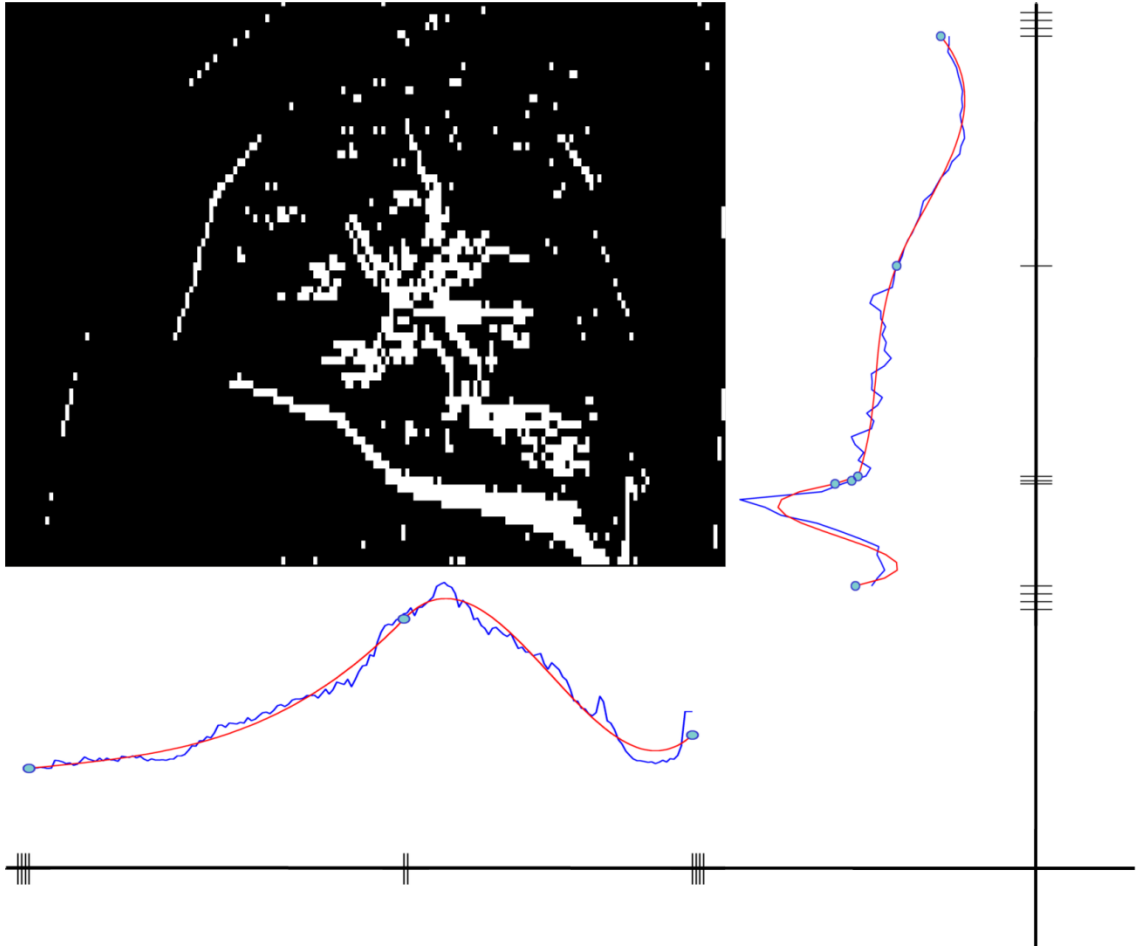
The added complexity of 3D DIR as compared to both 3D B-spline fitting and 2D DIR required numerous modifications to previous algorithms. The first major modification is to the heuristic rule for knot placement. In the 2D DIR method, the 2D difference image was projected into two 1D error profiles. However, projecting the difference volume in an analogous way resulted in a loss of structural information. The spatial distribution of the difference in voxel intensity between two CT volumes varies greatly from slice to slice. When directly applying the 1D projection method, the differences seen in individual slices were obfuscated by noise and by averaging, thus diminishing the efficacy of the knot insertion method. In order to reduce this effect, a 3D map of the errors larger than two standard deviations from the mean voxel intensity difference was created. This mapping was used instead of the actual difference volume to compute the three 1D error profiles. As a result, the 1D error profiles were more smoothly varying, had larger relative peaks, and retained the structural information of the real anatomical



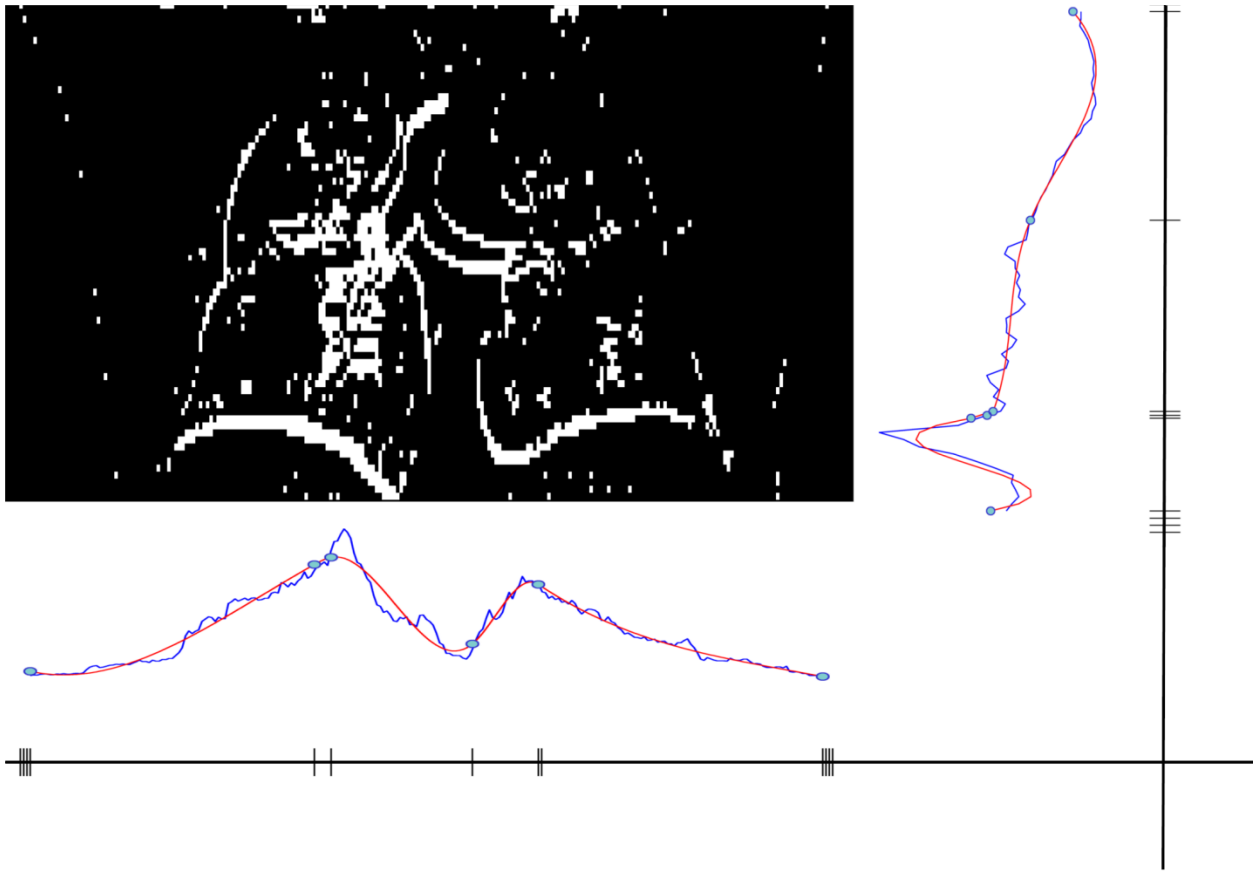
changes between the two CT volumes. Error maps and the associated 1D error profiles created in the axial, coronal, and sagittal planes can be seen in figures 86, 87, and 88, respectively.



**Figure 86:** Axial error map with 1D projections and the resultant knot vectors.



**Figure 87:** Sagittal error map with 1D projections and the resultant knot vectors.

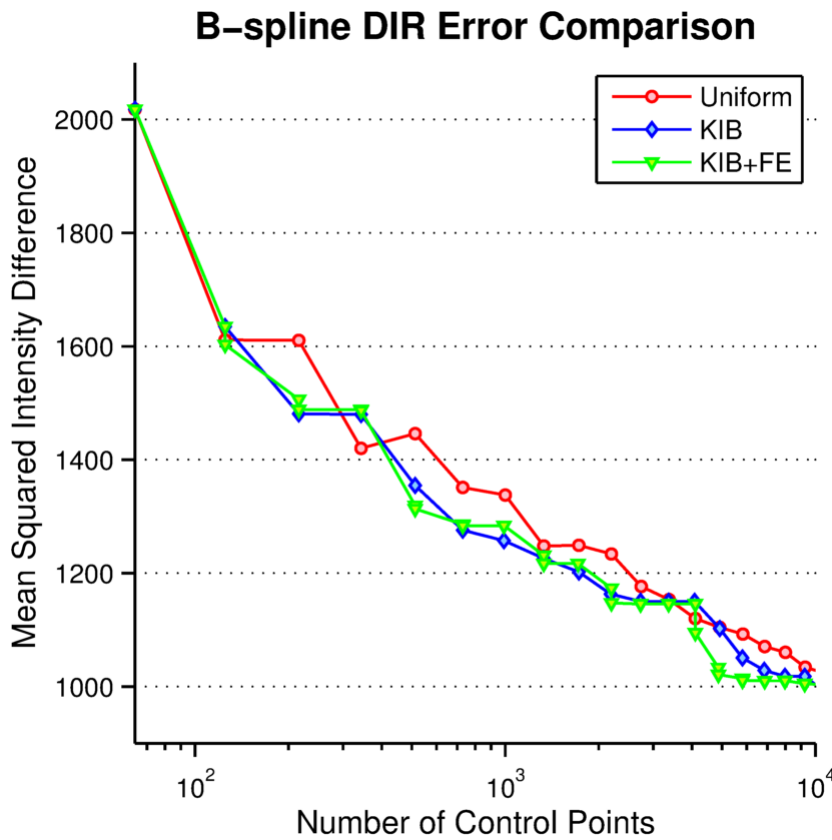


**Figure 88:** Coronal error map with 1D projections and the resultant knot vectors.

### 5.2.1 Incremental Knot Insertion DIR Accuracy

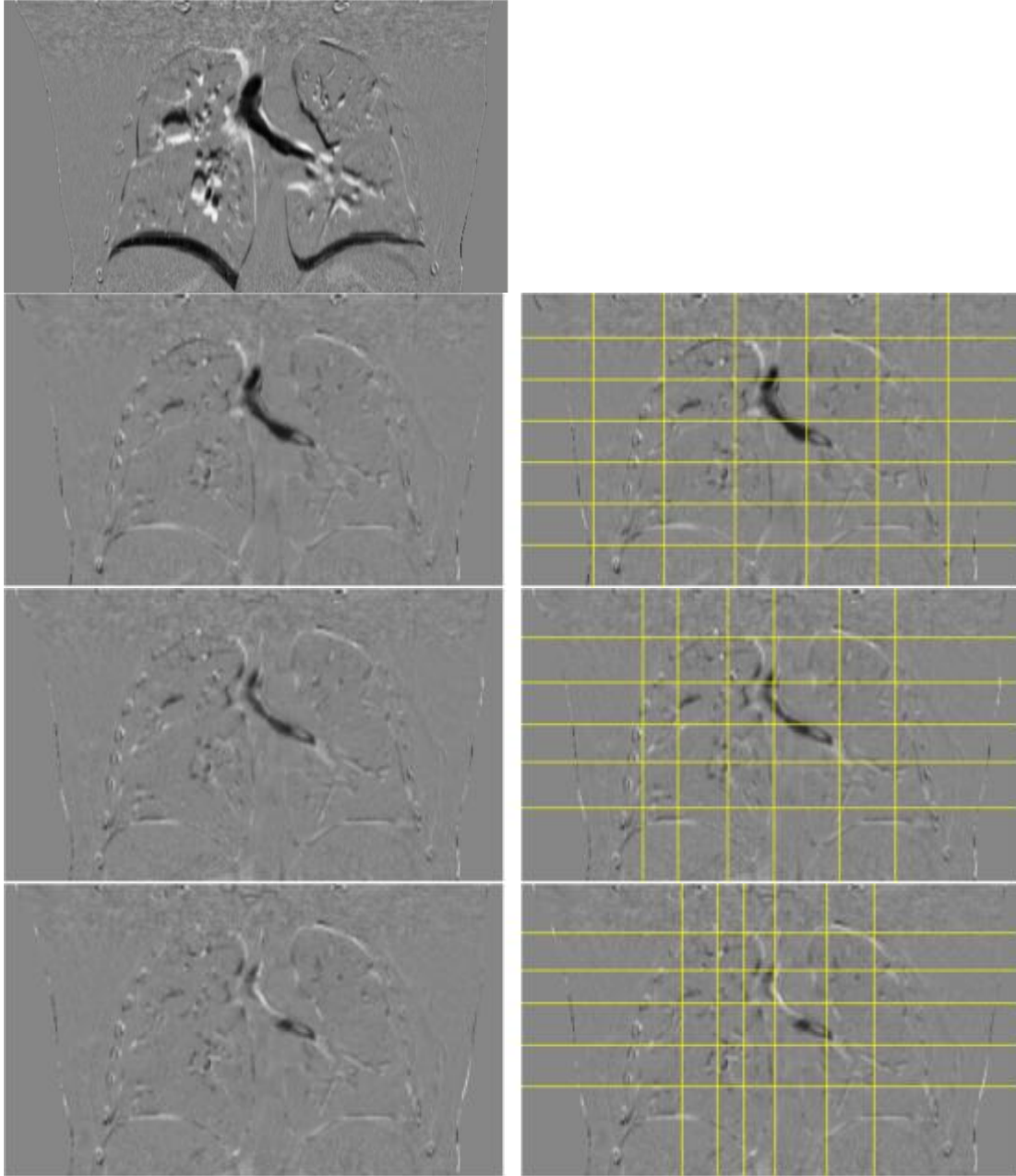
A target CT is created by applying the Demon's-generated DVF from the POPI dataset to the source CT. For this purpose, the DVF mapping the end-inhalation phase to the end-exhalation phase was used. The CT was scaled to match the dimensions of the DVF, resulting in a 235 by 176 by 141 voxel image dataset. Creating a synthetic DVF made the two image sets noise consistent and also allowed for a comparison between the DVF generated by the various algorithms and the applied DVF. As a first approach, B-spline DIR using knots determined by the uniform, KIB, and KIB+FE methods were compared. Up to this point, the non-uniform methods have all incremented the number of knots uniformly in each direction. This is

unnecessary, and, as was observed in Section 4.2, is sub-optimal with respect to the fitting error per number of knots. Instead, after the initial fit, the knot grid is updated by bisecting the three knot segments containing the largest number of errors greater than two standard deviations from the mean error, irrespective of the axis in which they reside. That is, these three knot segments may all exist along the same directional axis. This results in knot vectors that are non-uniform in both knot spacing and number of knots per axis. In this experiment, rather than iteratively adding a single knot in each direction, the non-uniform knot methods placed three knots to bisect the three largest error segments according to the 1D error profiles. Figure 89 plots the DIR error versus the log of the number of control points for the three methods. The non-uniform methods exhibited slightly lower mean squared intensity differences as compared to the uniform method.

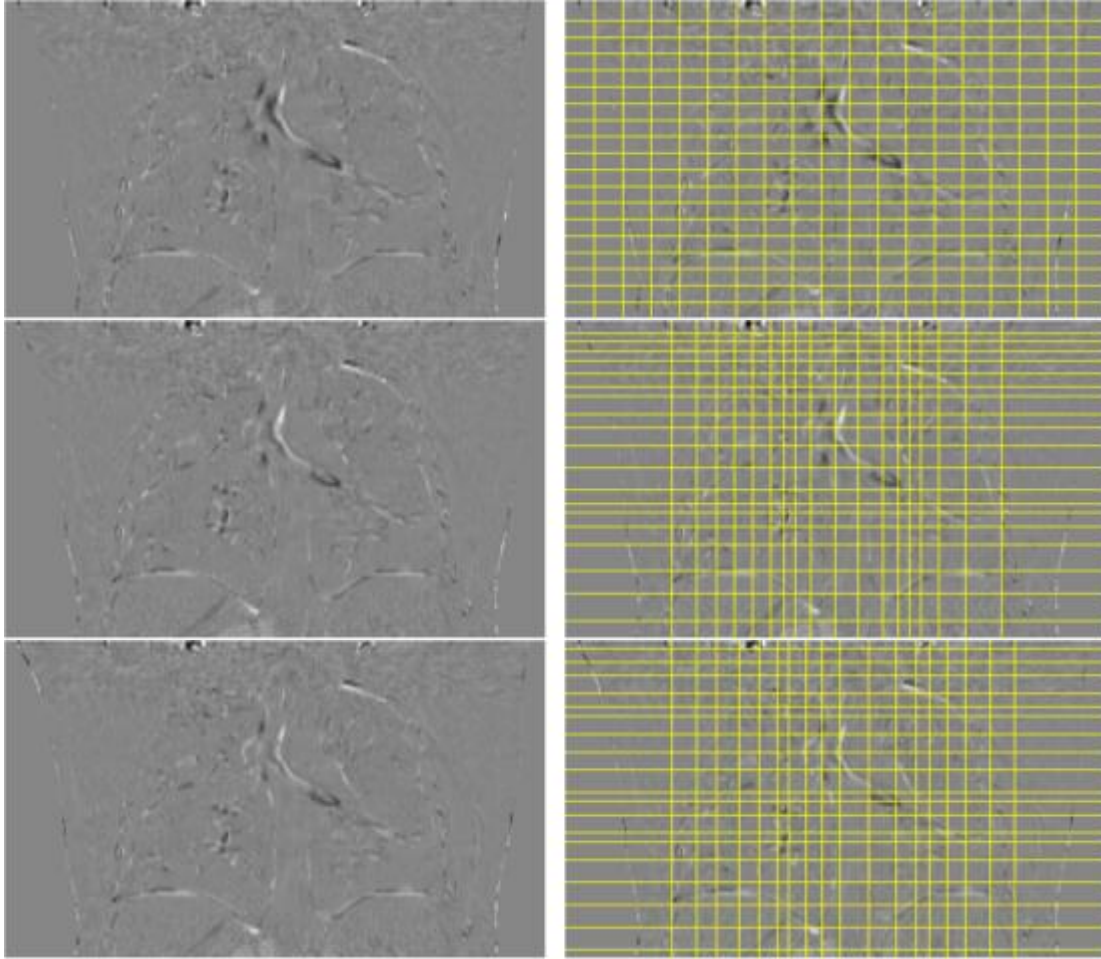


**Figure 89:** Error comparison of the three incremental B-spline DIR methods. Error is plotted as a function of the log of the number of control points.

Figure 90 compares the initial difference image in the coronal plane with the difference images after registration via the uniform (second row), KIB+FE (third row), and KIB (fourth row) knot placement methods. The uniform method had a 10 by 10 by 10 control point grid for a total of 1000 control points while the KIB and KIB+FE methods each heuristically selected 11 by 10 by 9 control point grids. The right hand column displays an overlay of the knot positions in yellow. There is a visually discernable difference in the quality of the registrations, particularly in the main airways. However, with the use of 10,000 control points (figure 91), the three methods are qualitatively indiscernible from one another. Figure 89 confirms that with 10,000 control points the three methods have nearly identical mean squared intensity differences.



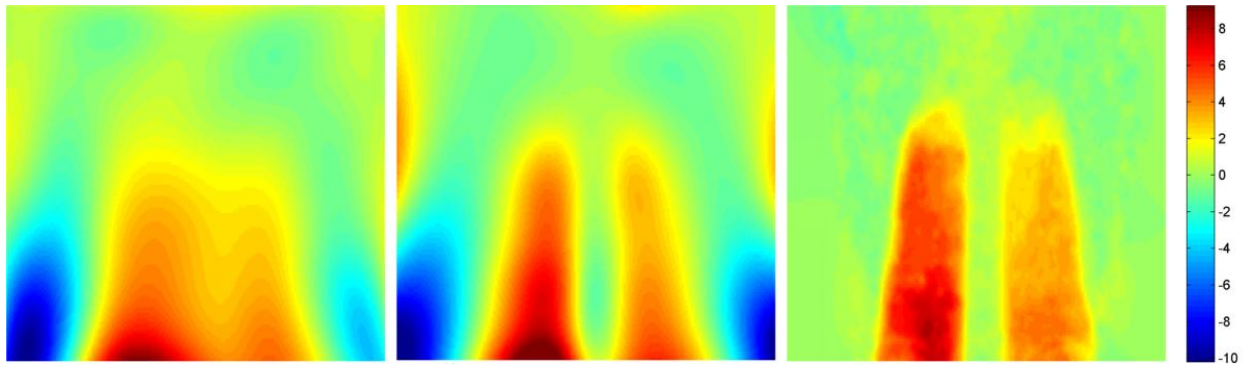
**Figure 90:** Difference images in a coronal slice of the CT pair before registration (top panel), and after registration with the three methods (uniform, KIB+FE, KIB). The frames on the right overlay the knot distribution with 6 free knots. The uniform method had a 10 by 10 by 10 control point grid, while both non-uniform methods selected a control point grid of 11 by 10 by 9.



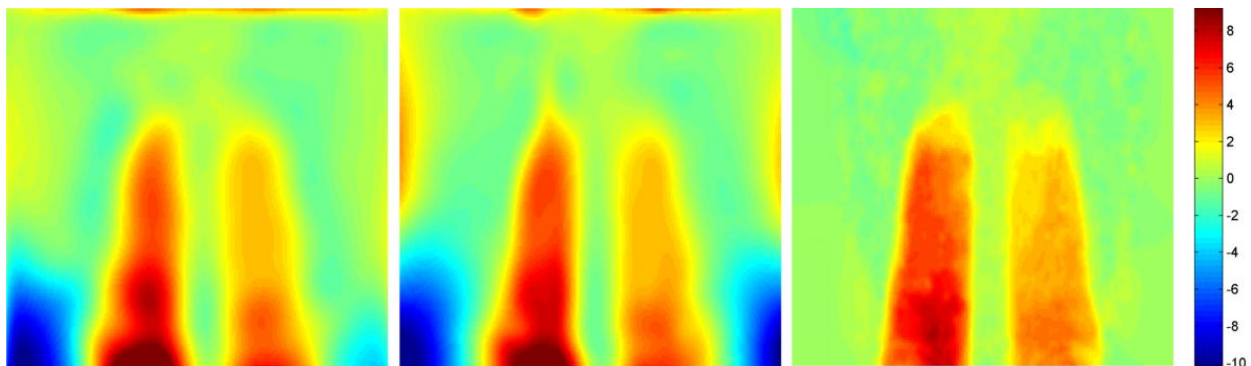
**Figure 91:** Coronal difference images resulting from registration using uniform, KIB+FE, and KIB methods with approximately 10,000 control points.

A second qualitative investigation of the DIR methods is made by comparing the resultant DVFs with the Demon's DVF used to generate the target CT. Once again, the difference between the uniform and non-uniform B-spline DVFs is apparent in the case with fewer control points. Specifically, the added shape control gained from the use of non-uniform knot placement allows the DIR to resolve the two lobes of the lung with fewer control points than the uniform method (figure 92). In general, it is observed that the non-uniform method results in a B-spline DVF with a higher level of fine detail, though neither can exactly model the underlying non-regularized Demon's generated DVF. This can be seen in figures 96 and 100,

which contain DVF slices from the sagittal and axial planes, respectively. It is worth noting that a quantitative comparison between the DVFs (e.g. a sum of squared differences), will not necessarily yield a result indicative of overall the DIR accuracy. This is because the value of the B-spline DVF may differ from the underlying target DVF in regions with little or no image contrast without adversely affecting the sum of squared differences of the pixel intensities. Furthermore, the non-uniform method is expected to result in coarser grid spacing and shape control in those regions. This can be seen on the periphery of the DVF volumes in all three planar views.



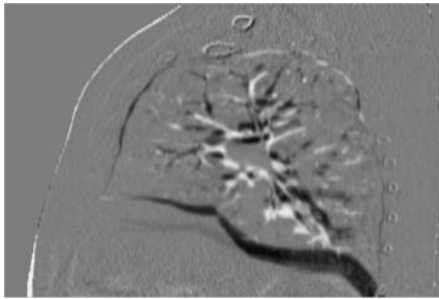
**Figure 92:** Contour plots of a coronal plane containing the z-component of the DVF. The left panel is the DVF generated by the uniform B-spline method with 1,000 control points, the middle panel is the DVF generated by the non-uniform B-spline method with 990 control points, and the right panel is the target DVF generated by the Demon's algorithm.

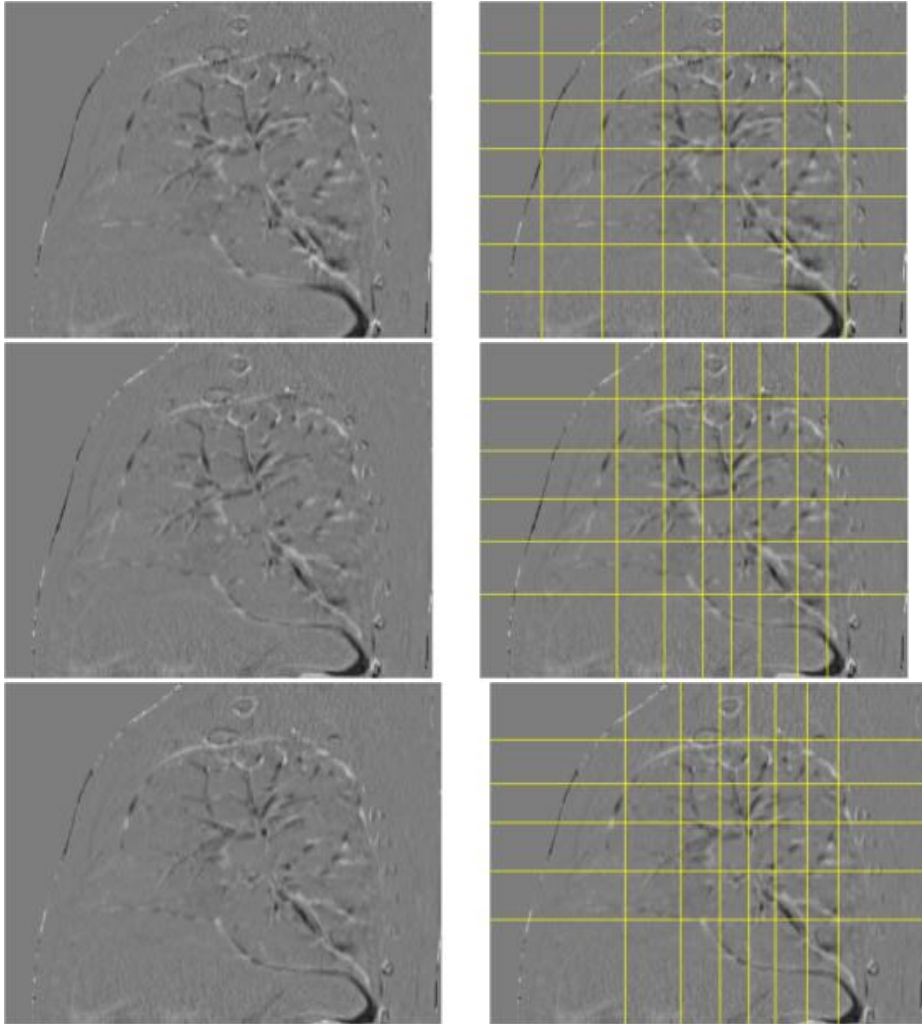


**Figure 93:** Contour plots of a coronal plane containing the z-component of the DVF. The left panel is the DVF generated by the uniform B-spline method with 10,000 control points, the middle panel is the DVF generated by the non-uniform B-spline method with 990 control points, and the right panel is the target DVF generated by the Demon's algorithm.

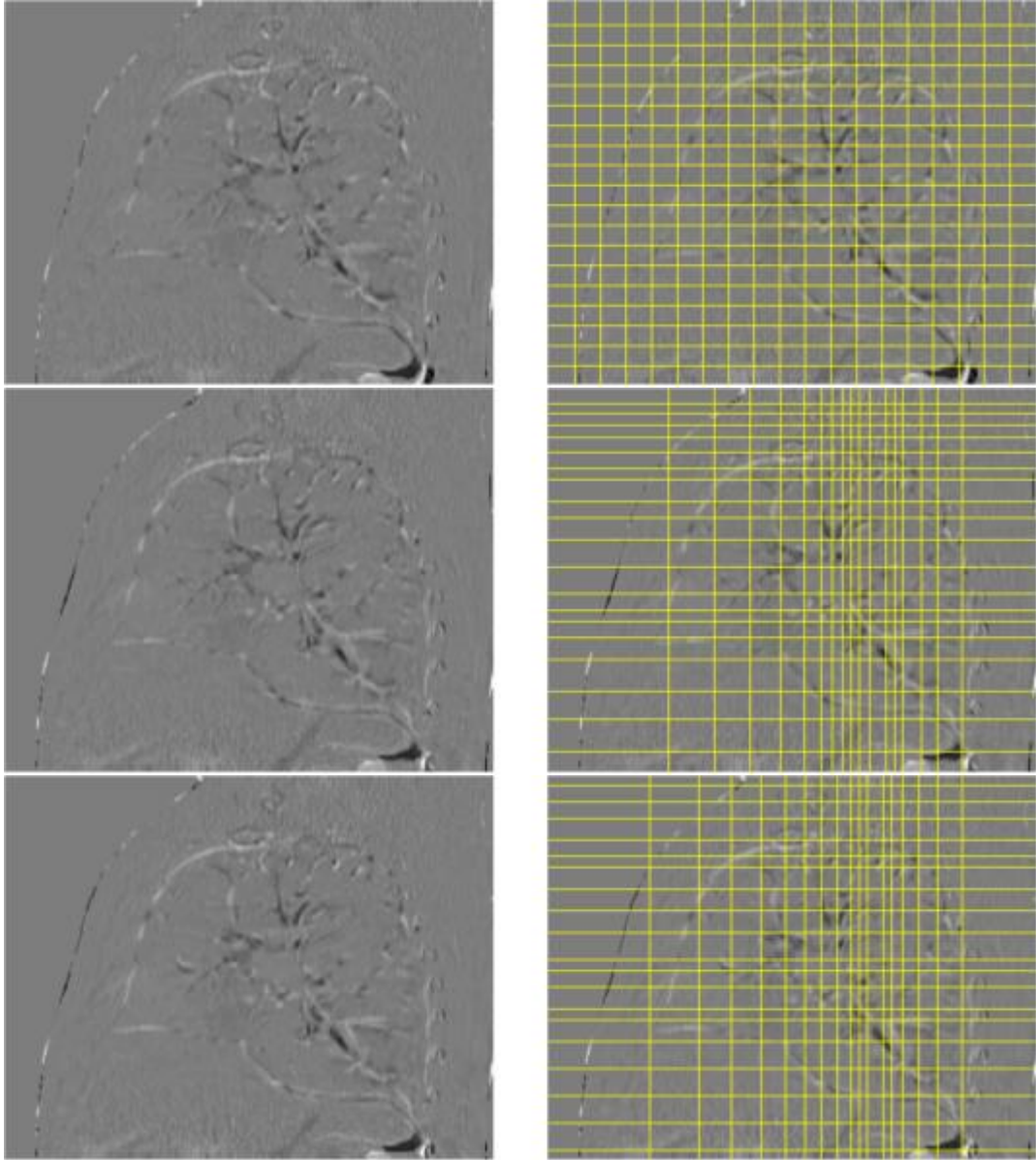


non-uniform B-spline method with 990 control points, and the right panel is the target DVF generated by the Demon's algorithm.

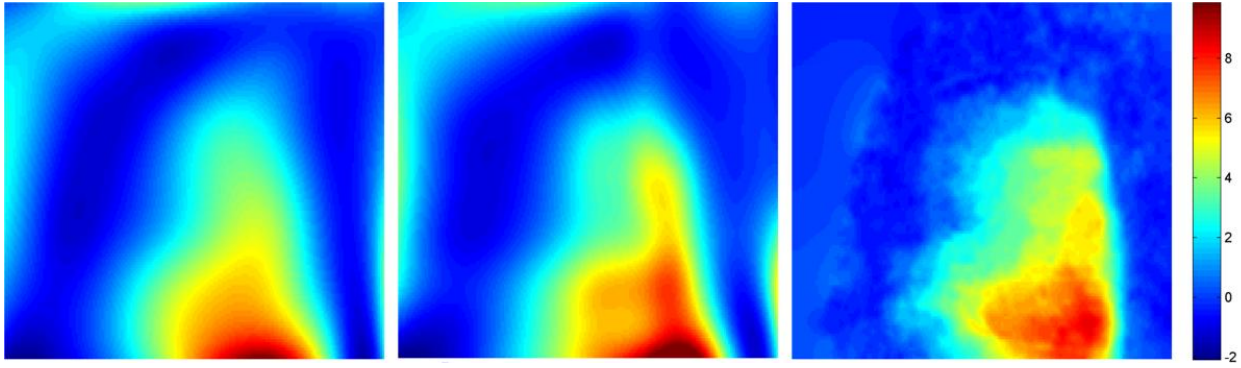




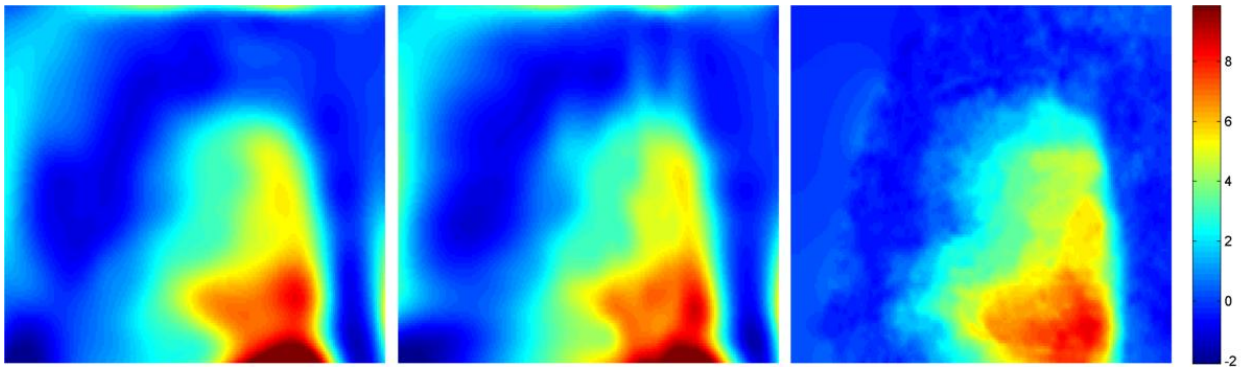
**Figure 94:** Difference images in a sagittal slice of the CT pair before registration (top panel), and after registration with the three methods (uniform, KIB+FE, KIB). The frames on the right overlay the knot distribution with 6 free knots. The uniform method had a 10 by 10 by 10 control point grid, while both non-uniform methods selected a control point grid with dimensions of 11 by 10 by 9.



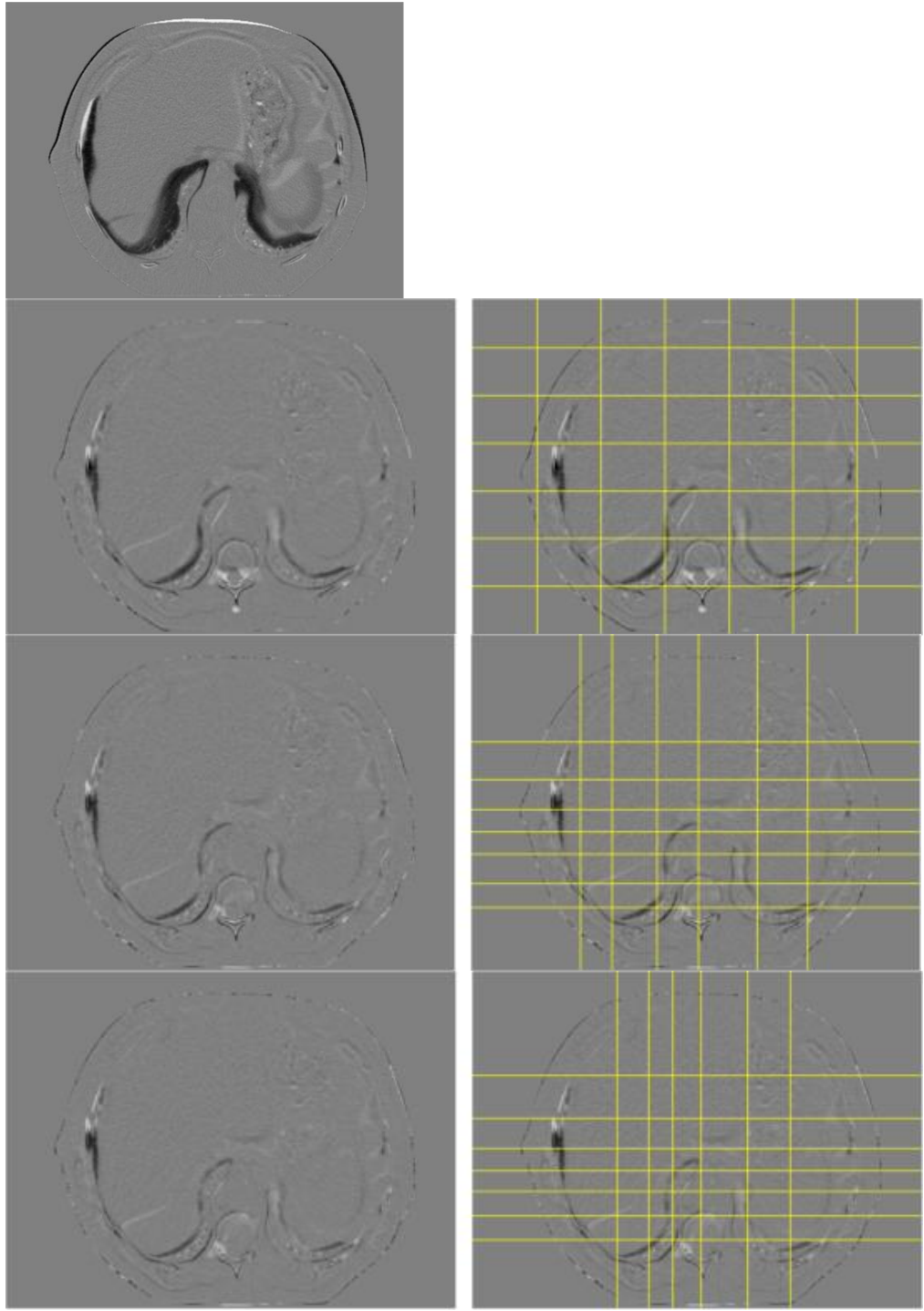
**Figure 95:** Sagittal difference images resulting from registration using uniform, KIB+FE, and KIB methods with approximately 10,000 control points.



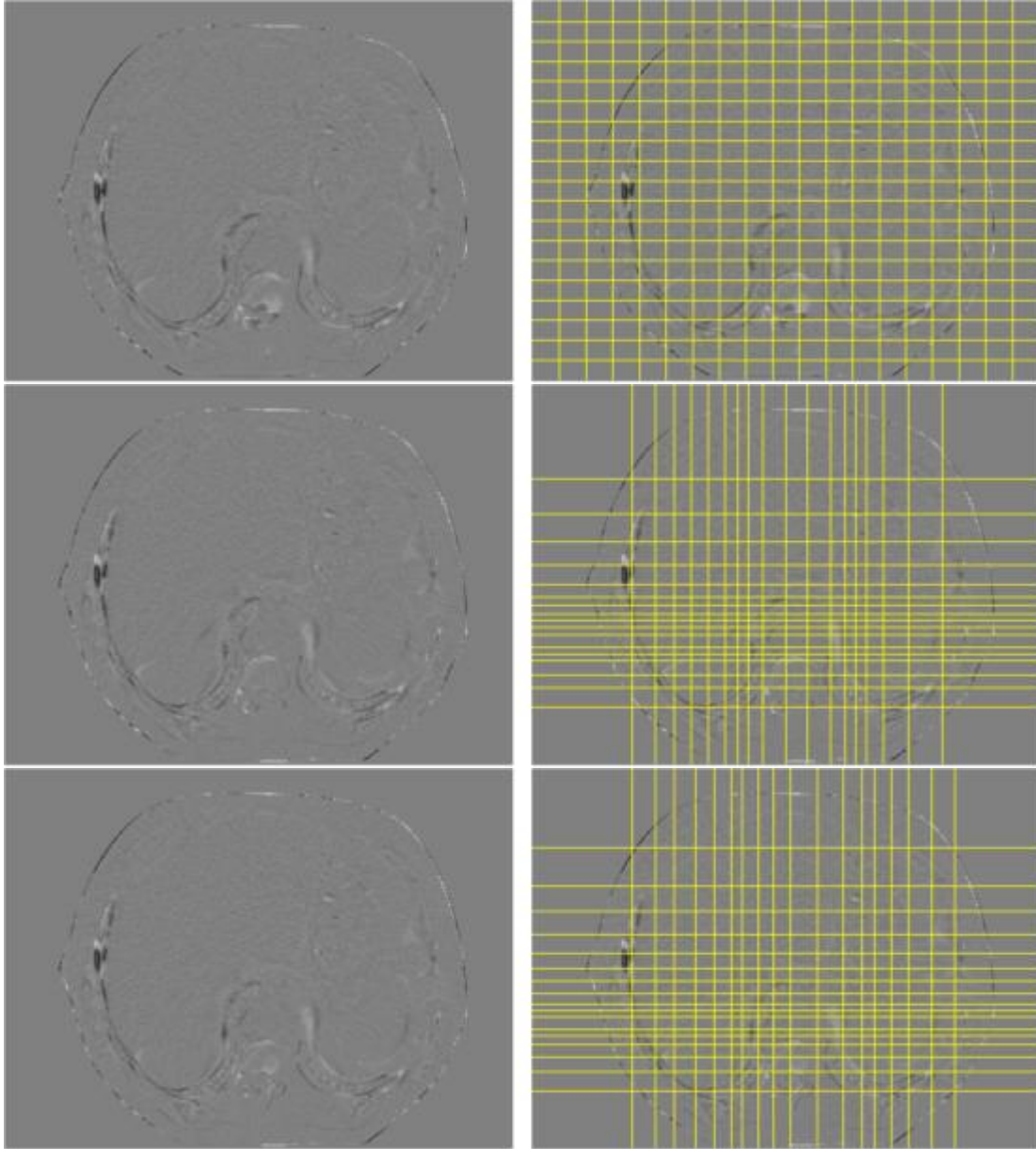
**Figure 96:** Contour plots of a sagittal plane containing the z-component of the DVF. The left panel is the DVF generated by the uniform B-spline method with 1,000 control points, the middle panel is the DVF generated by the non-uniform B-spline method with 990 control points, and the right panel is the target DVF generated by the Demon's algorithm.



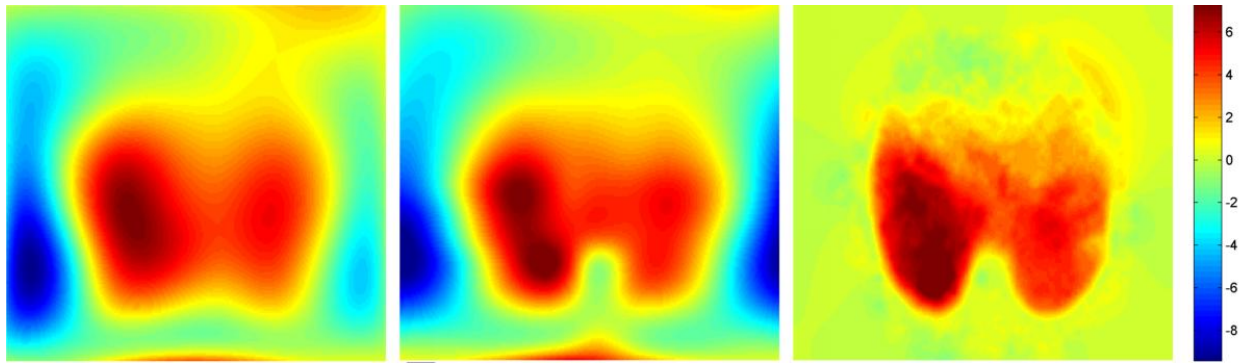
**Figure 97:** Contour plots of a sagittal plane containing the z-component of the DVF. The left panel is the DVF generated by the uniform B-spline method with 10,000 control points, the middle panel is the DVF generated by the non-uniform B-spline method with 990 control points, and the right panel is the target DVF generated by the Demon's algorithm.



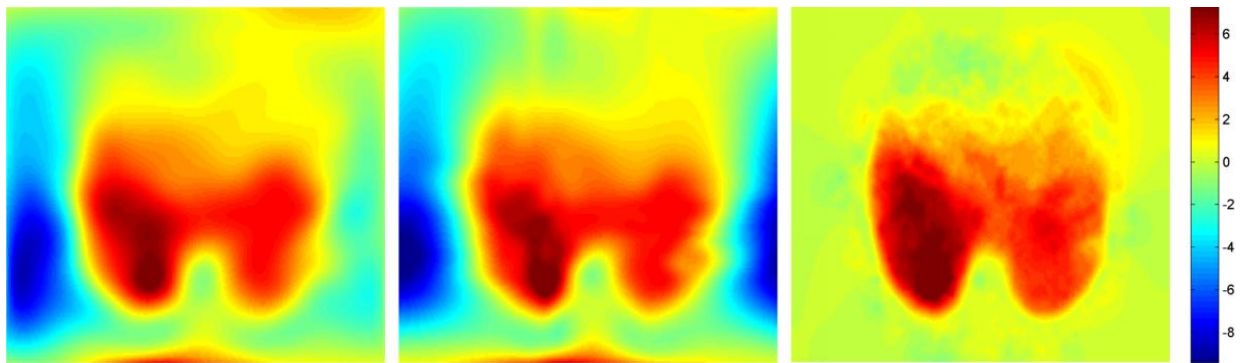
**Figure 98:** Difference images in a axial slice of the CT pair before registration (top panel), and after registration with the three methods (uniform, KIB+FE, KIB). The frames on the right overlay the knot distribution with 6 free knots. The uniform method had a 10 by 10 by 10 control point grid, while both non-uniform methods selected 11 by 10 by 9 control point grid.



**Figure 99:** Axial difference images resulting from registration using uniform, KIB+FE, and KIB methods with approximately 10,000 control points.



**Figure 100:** Contour plots of an axial plane containing the z-component of the DVF. The left panel is the DVF generated by the uniform B-spline method with 10,000 control points, the middle panel is the DVF generated by the non-uniform B-spline method with 990 control points, and the right panel is the target DVF generated by the Demon's algorithm.



**Figure 101:** Contour plots of an axial plane containing the z-component of the DVF. The left panel is the DVF generated by the uniform B-spline method with 10,000 control points, the middle panel is the DVF generated by the non-uniform B-spline method with 990 control points, and the right panel is the target DVF generated by the Demon's algorithm.

### 5.2.2 Incremental Knot Insertion DIR Computation Time

The time to compute the DIR with respect to the number of control points is displayed in Figure 102. The difference in the computation time is a consequence of the way that the knot grids are incremented. In the uniform method, a new knot vector is computed which subdivides the parameter space into equal segments. On this new knot vector domain, the previously computed control points are no longer valid. Instead, the control points associated with the new knot vector are computed by fitting the new B-spline to the previous B-spline (i.e. solving equation 3.5). The time required to solve this equation (matrix inversion) scales with the total number of control points. The KIB method, on the other hand, inserts a single knot  $\bar{k}$  in the  $s$ th knot span (i.e.  $\bar{k} \in [k_s, k_{s+1})$ ) without disturbing the position of the pre-existing knots  $k_i$ . Due to the local support property of B-splines, only  $p+1$  control points need to be recomputed per dimension, where  $p$  is the degree of the B-spline. This is illustrated in Figure 102. This powerful property of the B-spline is well-known and exploited in computer aided design, but has not been explored in the context of DIR. As shown by Piegl [Piegl, 1997], the new control point set  $\mathbf{Q}$  can be solved from the old control point set  $\mathbf{P}$  by

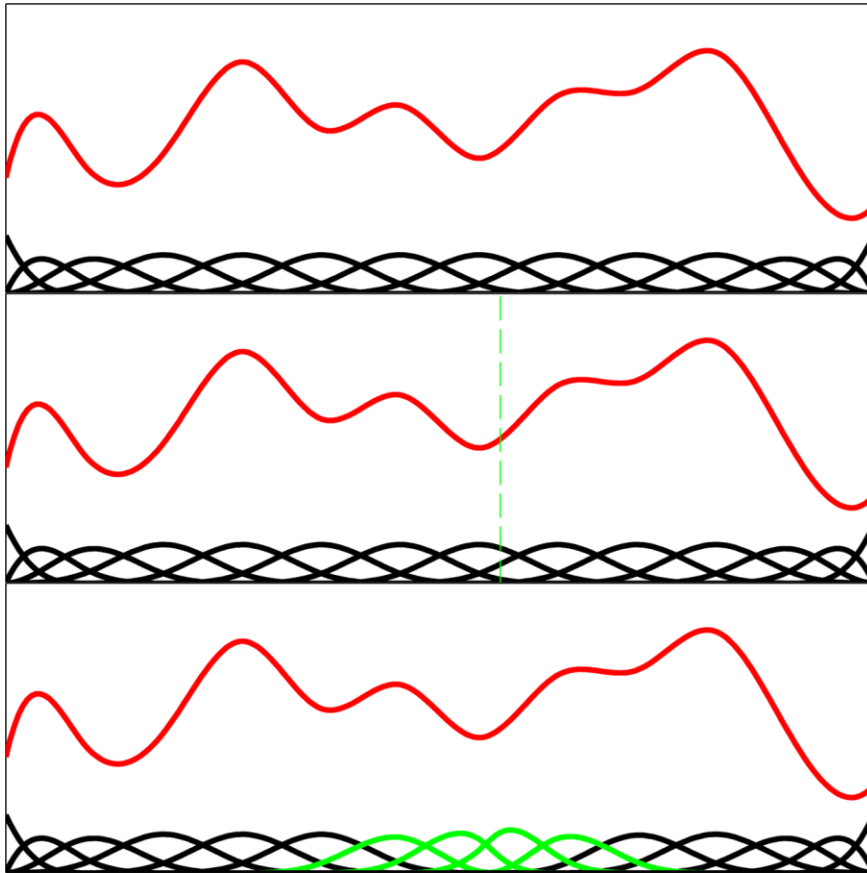
$$\mathbf{Q}_i = \alpha_i \mathbf{P}_i + (1 - \alpha_i) \mathbf{P}_{i-1} \quad (5.4)$$

$$\alpha_i = \begin{cases} 1, & i \leq s - p \\ \frac{\bar{k} - k_i}{k_{i+p} - k_i}, & s - p + 1 \leq i \leq k \\ 0, & i \geq s + 1 \end{cases} \quad (5.5)$$

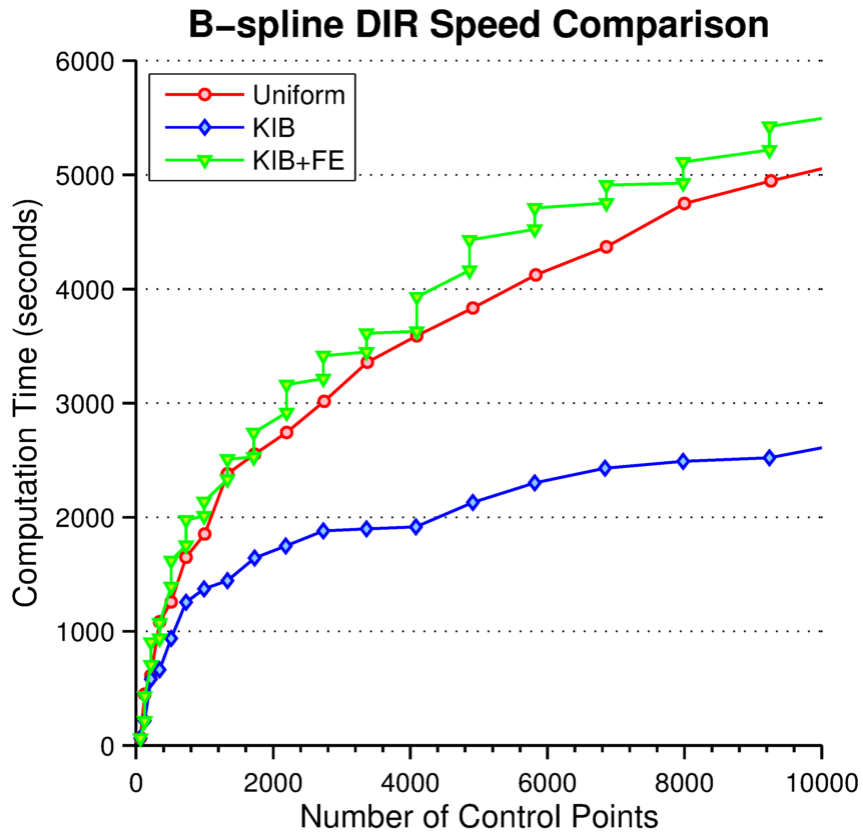
This algorithm allows for a single knot to be added with very little computational burden. It can also be used recursively to add a large number of knots. When called recursively to bisect each knot segment in the uniform B-spline, the computation time for the KIB method was reduced by



a factor of two compared to the uniform incremental method. Because the KIB+FE method allows all of the knot values to change in its force equilibration step, it requires that all the control points be recalculated. In addition, it involves two optimizations per control point iterations (see Figure 79). Consequently, it requires the longest computation time.



**Figure 102:** Adding a knot to the B-spline curve will only affect  $p+1$  basis functions. Because of this, it is only necessary to compute  $p+1$  new control point values



**Figure 103:** Comparison of the total time required to compute the three B-spline methods. The uniform B-spline method requires nearly twice as long to compute as the KIB non-uniform method. Note that from Figure 99, the error upon completion of the registration with 10,000 control points produces very nearly the same error.

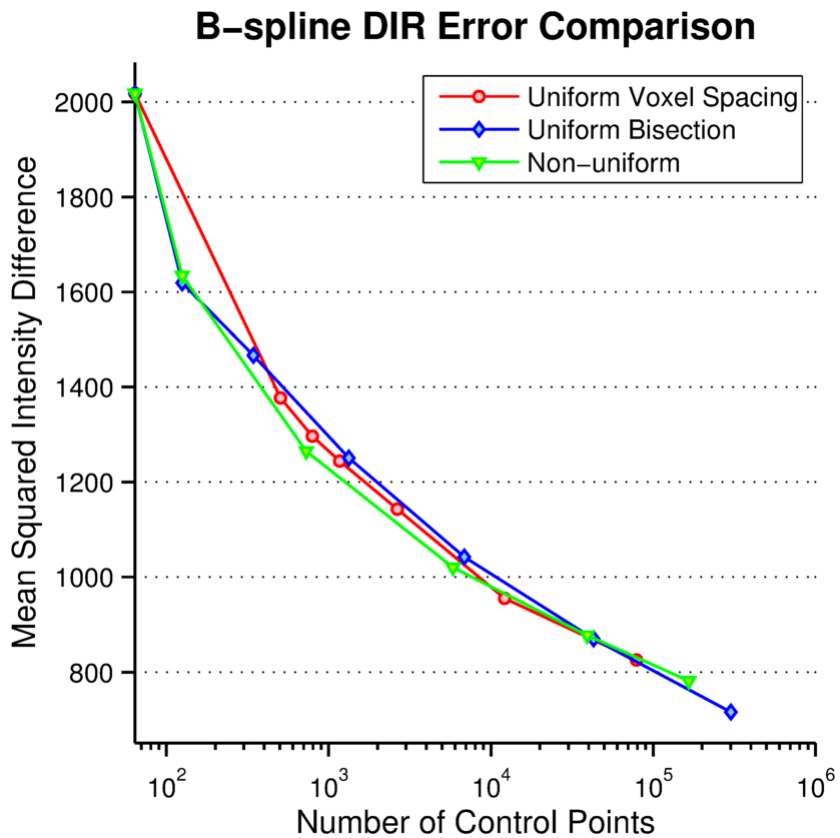
### 5.2.3 Multiple Knot Insertion DIR

Increasing the number of control points by adding a few knots per iteration is useful for comparing the accuracy of uniform and non-uniform B-spline DIR on a per control point basis, but it is an impractical strategy for the purpose of fast DIR. Incrementing the number of knots in small steps results in redundant optimization of the control points. For example, inserting non-uniform knots, even with the force equilibration step, was often not enough to dislodge the optimization from a local minimum (see Figure 89). A common approach to correct for this shortcoming involves grid refinement with large steps, often doubling the control point grid resolution at each refinement step. This approach is referred to as multiscale or multilevel B-spline DIR. This section develops methods for multiple knot insertion DIR.

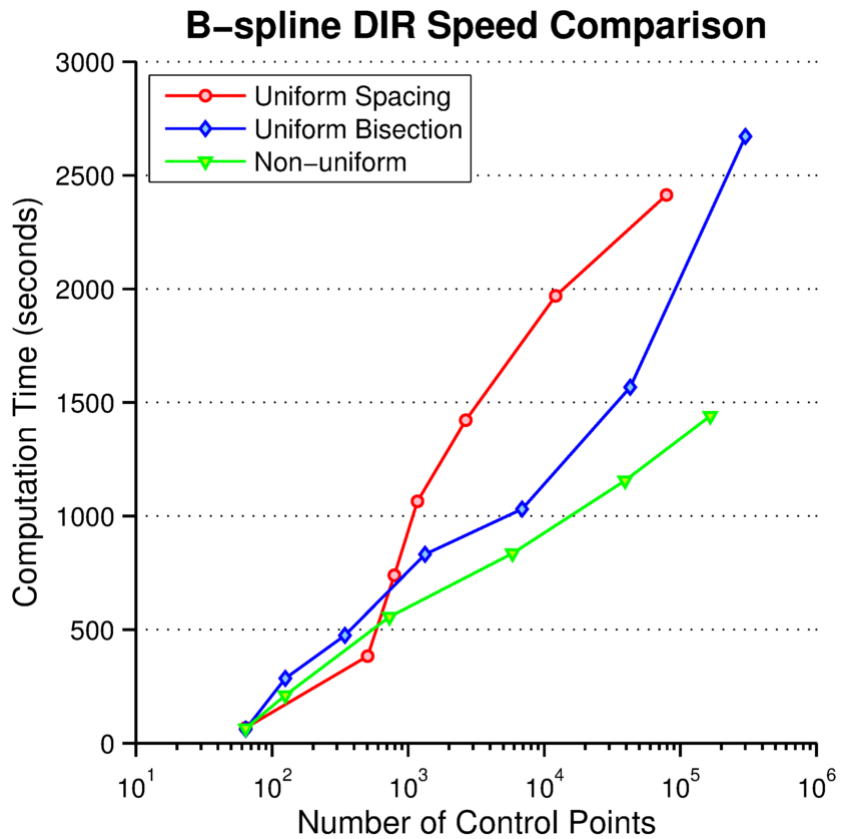
Two uniform methods and one non-uniform method are compared. The first uniform method refines the grid by decreasing the knot spacing from 40 to 10 voxels in increments of 10 voxels, and down to 5 on a final refinement step. The second uniform method starts with zero free knots and bisects every non-zero knot segment each iteration. The initialization of control points to the previous estimate of the DVF is computed by invoking equation 5.4 after each bisection. Similarly, the non-uniform method adds  $3n^2$  knots per iteration, where  $n$  is the iteration number. This relationship was chosen based on the resulting exponential increase in control points, which mimics the effect of bisecting every non-zero knot segment. Again, equation 5.4 is used recursively after each knot insertion to solve for the initial control point values belonging to the new knot grid.

The multilevel B-spline DIR errors and the time to compute the DIR are compared in Figures 103 and 104, respectively. An important feature to note is that all three multilevel

techniques succeed in reducing the error at each refinement step. The large number of control points added per refinement avoids getting stuck in local minima on consecutive optimizations and ensures that fewer global control point optimizations are required to complete a registration with a larger number of control points. This is demonstrated in Figure 104. Despite computing an order of magnitude more control points, the multilevel techniques reduce the computation time by a factor of two.

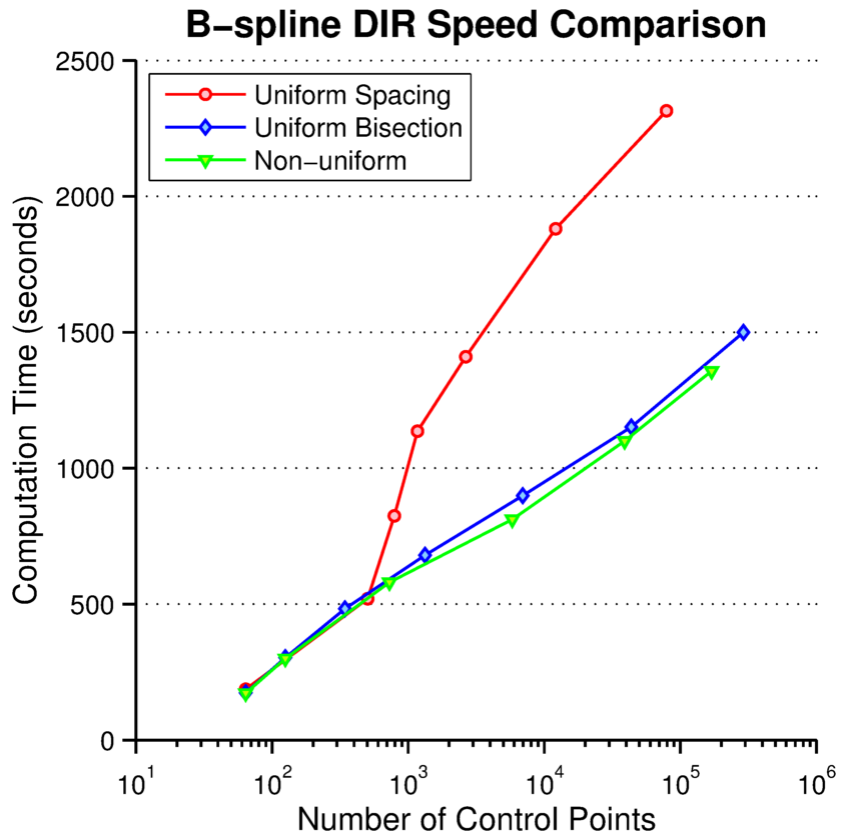


**Figure 104:** Comparison of the mean squared intensity difference after each of the three multilevel B-spline DIR methods. The non-uniform method slightly out-performs the uniform methods until after 100,000 control points.



**Figure 105:** Comparison of the total time required to compute the three B-spline multilevel methods as a function of the number of control points.

This experiment was repeated for each of the 9 breathing phases registered to the reference phase. The error, number of control points, and computation time were averaged over the breathing phases. Figure 105 plots the time to compute each method as a function of number of control points, and Figure 106 plots the mean squared intensity difference as a function of time. It was observed that the non-uniform multilevel method takes less time to produce the same (or better accuracy) than either of the uniform methods. Additionally, the uniform B-spline DIR algorithm's speed is greatly improved by the use of the knot bisection method over a grid refinement scheme which requires the direct solution of control points between every refinement step.

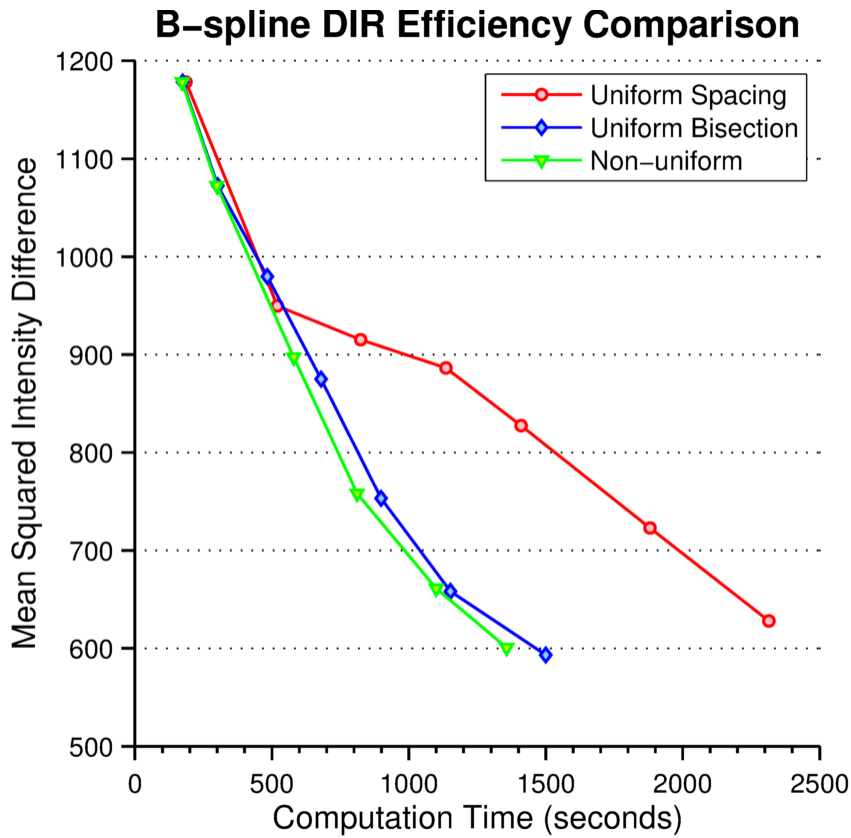


**Figure 106:** Comparison of the times to compute each multilevel registration, averaged over the 9 breathing phases.

### 5.3 Summary of Investigation

Chapter 5 incorporated the heuristic rules for knot placement into the DIR framework, first in 2D and then in 3D. Modifications were made to the algorithm so that the knots were placed preferentially in areas of large disagreement, as informed by the difference image computed at each refinement step. The benefit of non-uniform knots was demonstrated in 2D through use of the knot insertion technique with force equilibration. It was shown that the non-uniform method correctly placed knots in areas corresponding to large anatomical displacements, resulting in better DIR accuracy than was achievable with uniform knots. As was the case with non-uniform B-spline fitting, diminishing returns were observed in higher dimension non-uniform B-spline DIR. However, small improvements in DIR accuracy were demonstrated. In

addition, qualitative comparisons of the computed DVFs showed a higher level of fine detail in those resulting from the non-uniform B-spline method. Finally, a method for large, non-uniform, refinement steps was developed and its improved computational efficiency was demonstrated.



**Figure 107:** Comparison of the errors versus computation time, averaged over the 9 breathing phases.

## 6. Conclusions and Discussion

### 6.1 Conclusions

This work presented an investigation of a new approach to DIR based on NURBS. Despite being a more general and flexible formulation of B-splines, as well as a commonly implemented mathematical tool in CAD, no NURBS-based DIR algorithms have been documented in the literature to date. The motivation for the use of NURBS in the DIR framework was to use the additional free parameters associated with NURBS in an attempt to represent discontinuous DVFs, which are problematic for inherently smooth conventional B-splines. However, it was discovered early in the project that on a per-free parameter efficiency basis, the rational weights of NURBS offer no advantage over conventional B-spline control points when fitting DVFs. This discovery shifted the focus of the work to the creation of heuristic rules for knot placement of non-uniform tensor product B-splines (henceforth referred to as NUBS), which up until this study had also not been investigated in the context of DIR. In fact, the question of how to optimize NUBS knot placement has not been definitively answered even in the case of directly fitting curves and surfaces. It therefore became necessary to start from 1D function fitting and incrementally build up complexity to 3D DIR. The progression led to many challenges distinct from those addressed in the B-spline literature. For example, existing NUBS fitting algorithms rely upon information about the features of the target dataset in order to optimize knot placement, whereas in the DIR framework the underlying target function is unknown. This necessitated the creation of novel knot placement algorithms which are instead guided by the iterative error, irrespective of whether that error is defined as the difference between two



functional values or the difference in image intensity between a registration image pair. To this end, a set of heuristic rules was created, and the efficacy was demonstrated in 1-, 2-, and 3D NUBS fitting. These rules eventually formed the basis for a novel NUBS DIR scheme. In 2D, the NUBS DIR scheme with knots placed according to the KIB+FE algorithm was found to reduce the registration error compared to a uniform B-spline DIR with the same number of control points. This demonstrated proof of concept of a NUBS-based DIR framework for the first time. In 3D, large datasets and long computation times motivated the search for a more efficient grid refinement step in the registration process. A knot insertion method was implemented which allows the ability to add a knot at any point along the existing directional knot vectors without changing the value of the B-spline tensor product. When performed recursively, it enables the efficient addition of a large number of knots. This powerful B-spline algorithm is used for interactive design in CAD. However, to the best of the author's knowledge, it has not been explicitly mentioned in the medical imaging literature. Furthermore, prior to this work it had never been fully exploited to efficiently add knots in areas of deficient registration.

## **6.2 Discussion and Future Work**

NURBS were investigated as a way to model the discontinuities observed in the sliding motion of certain anatomical structures. Unfortunately, the rectilinear grid structure required by tensor product B-splines severely limited the efficacy of knot optimization. An ideal mathematical framework would allow the insertion of a single knot at an arbitrary location in the image space, thereby offering increased local DVF control without a large number of additional control points. There are existing methods that have been proposed to handle this problem. For

example, Shusharina and Sharp have investigated radial basis functions to produce “meshless” registration [Shusharina and Sharp, 2012]. Hierarchical B-splines have been developed which can overlay grids to produce nearly arbitrary topology. Additionally, T-splines, a variation of non-uniform B-splines which includes “T” shaped junctions in the knot grid, have been developed for use in CAD and CAE, with implications in finite element analysis [Sederberg et al, 2003]. At the time of writing and to the best of the author’s knowledge, T-splines have not been incorporated into a DIR framework, but it seems like a promising application for the new technique of B-spline shape control.

This paper used the sum of squared pixel intensity differences in order to assess registration accuracy, which can be problematic. As Castillo et al point out [Castillo et al, 2013], “the goal in applying DIR is to obtain accurate spatial registration of the underlying anatomy, and not simply quantitative image similarity.” However, pixel intensity difference does not necessarily correspond to the accurate registration of anatomy and can be affected by image noise and artifacts. Instead, the trend is to use the location of landmark features identified by experts as a standard for comparison of DIR methods. This study would need to be done to verify the clinical viability of the non-uniform knot method placement. In addition, using the pixel intensity difference in the cost function precludes the use of this method for the registration of images acquired by multiple modalities. This is required, for example, in the fusion of MRI and CT images, which can be performed by comparing the mutual information in the two images.

The paper focused on creating a fully automated image registration algorithm, but a semi-automated method with interactive insertion of knots to locally increase registration accuracy would be a useful tool. It was demonstrated that the insertion of knots without changing the values of the B-spline is both possible and efficient. Of course, re-optimization of the new

control point grid would remain computationally demanding, but Shackleford et al have shown orders of magnitude decrease in computation time by utilizing graphics processing units (GPU) [Shackleford et al, 2010]. Parallelizing the computation of the 3D B-spline DVF to take advantage of GPU speed increases would greatly reduce the computation time required by this algorithm. Combining the faster GPU computation with the knot insertion method could produce an interactive tool which could update the DIR in near real-time.

## 7. References

- Brown, L. (1992). "A survey of image registration techniques." *ACM Compute. Surv.* 24(4): 326-376.
- Beliakov, G. (2004). "Least squares splines with free knots: global optimization approach." *Applied Mathematics and Computation* 149: 783-798.
- Boehm, W. (1980). "Inserting new knots into B-spline curves." *Computer Aided Design* 12(4): 199-201.
- Brock, K.K, McShan, D.L., Ten Haken, R.K., Hollister, S.J., Dawon, L.A., Balter, J.M. (2003). "Inclusion of organ deformation in dose calculations." *Med. Phys.* 30(3): 290-295.
- Brock, K.K. (2010). "Results of a multi-institution deformable registration accuracy study (MIDRAS)." *Int J Radiat Oncol Biol Phys* 76(2): 583-596.
- Carlson, N. (2009). *NURBS Surface Fitting with Gauss-Newton*. Master's thesis, Lulea University of Technology.
- Castillo, R., Castillo, E., Fuentes, D., Ahmad, M., Wood, A., Ludwig, M., Guerro, T. (2013). "A reference dataset for deformable image registration spatial accuracy evaluation using the COPDgene study archive." *Phys Med Biol* 58(9): 2861-2877.
- Christensen G.E., Carlson, B., Chao, K.S.C., Yin, P., Grigsby, P.W., Nguyen, K., Dempsey, J.F., Lerma, F.A., Bae, K.T., Vannier, M.W., Williamson, J.F. (2001). "Registration of serial-imaging studies using deformable anatomic templates." *Int. J. Rad Oncol Bio Phys.* 51(1): 227-243.
- D'Angelo, E, Loring, S.H., Gioia, M.E., Pecchiari, M, Moscheni, C (2004). "Friction and lubrication of pleural tissues." *Respir. Physiol. Neurobiol.* 142(1): 55-68.
- Dierckx, P. (1995). *Curve and Surface Fitting with Splines*. Oxford, England, Oxford University Press.
- Farin, G. (1992). "From conics to NURBS." *IEEE Computer Graphics and Applications* 12: 78-86.
- Forsey, D.R., Bartels, R.H. (1995). "Surface fitting with hierarchical splines." *ACM Trans. Graphics* 14: 134-161.
- Goldenthal, R., Bercovier, M. (2003). "Spline curve approximation and design by optimal control over the knots using genetic algorithms." Paper presented at EUROGEN 2003 – Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems, Barcelona, Spain.
- Hill, D.L.G., Batchelor, G., Holden, M., Hawkes, D.J. (2001). "Medical image registration." *Phys. Med. Biol.* 46(3):1-45.

- Jacobson, T.J., Murphy, M.J. (2011). "Optimized knot placement for B-splines in deformable image registration." *Med. Phys.* 38(8): 4579-82.
- Jaffray, D., Kupelian, P., Djemil, T., Macklis, R.M. (2007). "Review of image-guided radiation therapy." *Expet. Rev. Anticancer Ther.* 7(1): 89-103.
- Javidrad, F., Nouri, R. (2013). "Optimum B-spline curve fitting using a modified simulated annealing method." *Int. J. of Computer Aided Engineering and Technology*: 5(4): 311-328.
- Jupp, D.L.B., (1978). "Approximation to data by splines with free knots." *SIAM Journal on Numerical Analysis* 15(2): 328-343.
- Kashani, R., Hub, M., Balter, J., Kessler, M., Dong, L., Zhang, L., Xing, L., Xie, Y., Hawkes, D., Schnabel, J., McClelland, J., Joshi, S., Chen, Q., Lu, W. (2008). "Objective assessment of deformable image registration in radiotherapy: A multi-institution study." *Med Phys* 35(12): 5944-5953.
- Kybic, J., Unser, M. (2003). "Fast parametric elastic image registration." *IEEE Trans Image Processing* 12(11): 1427-42.
- Laurent-Gengoux, P., Mekhilef, M. (1993). "Optimization of NURBS representation." *Computer-Aided Design* 25(11): 699-700.
- Leal, N., Leal, E., Branch, J.W. (2010). *Simple method for construction of NURBS surfaces from unorganized points*. Paper presented at the Proceedings of the 19<sup>th</sup> International Meshing Roundtable, Chattanooga, Tennessee.
- Li, W., Xu, S., Zhao, G., Goh, L.P. (2005). "Adaptive knot placement in B-spline curve approximation." *Computer-Aided Design* 37(8): 791-797.
- Ma, W., Kruth, J.P. (1998). "NURBS curve and surface fitting for reverse engineering." *Int J Adv Manuf Technol* 14: 918-927.
- Marquardt, D. (1963). "An algorithm for least-squares estimation of nonlinear parameters." *SIAM Journal on Applied Mathematics* 11(2): 431-441.
- Nelder, J.A., Mead, R. (1965). "A simplex method for function minimization." *Comput. J.* 7: 308-313.
- Park, H., Lee, J. (2007). "B-spline curve fitting based on adaptive curve refinement using dominant points." *Computer-Aided Design* 39: 439-451.
- Periaswamy S., Farid, H. (2006). "Medical image registration with partial data." *Medical Image Analysis* 10: 452-464.
- Piegl, L., Tiller, W. (1997). *The NURBS book*. Berlin, Germany: Springer-Verlag.
- Press, W.H., Flannery, B.P., Teukolsky, S.A., Vetterling, W.T. (1992). *Numerical recipes in C: The art of scientific computing*. Cambridge, England: Cambridge University Press.

- Razdan, A. (1999). Knot placement for B-spline curve approximation, Technical Report, Arizona State University, 1999,  
<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.33.8998&rep=rep1&type=pdf>
- Rogers, D.F. (2000). *An introduction to NURBS: With historical perspective*. Burlington, Massachusetts: Morgan Kaufmann.
- Rohde, G.K., Aldroubi, A., Dawant, B.M. (2003). “The adaptive bases algorithm for nonrigid image registration.” *IEEE Trans. Med. Imaging* 22: 1470-1479.
- Rueckert, R., Sonoda, L.I., Hayes, C., Hill, D.L., Leach, M.O., Hawkes, D.J. (1999). “Nonrigid registration using free-form deformations: application to breast MR images.” *IEEE Trans Med Imaging* 18(8): 712-21.
- Rueckert, R. (2001). *Nonrigid registration: Concepts, algorithms, and applications* (pp. 281-299). Danvers, MA: CRC Press LLC.
- Saleh-Sayah, N., Weiss, E., Salguero, F.J., Siebers, J.V. (2011). “A distance to dose difference tool for estimating the required spatial accuracy of a displacement vector field.” *Med. Phys.* 38(5): 2318-2323.
- Sarrut, D. (2006). “Deformable registration for image-guided radiation therapy.” *Med. Phys.* 16(4): 285-297.
- Schnabel, J.A., Rueckert, R., Quist, M., Blackall, J.M., Castellano-Smith, A.D., Hartkens, T., Penney, G.P., Hall, W.A., Liu, H., Truwit, C.L., Gerritsen, F.A., Hill, D.L.G., Hawkes, D.J. (2001). *Lecture notes in computer sciences: Medical image computing and computer-assisted intervention* (pp. 573-581). Berlin, Germany: Springer-Verlag.
- Sederberg, T.W., Zheng, J., Sewell, D., Sabin, M. (1998). “Non-uniform recursive subdivision surfaces.” *SIGGRAPH '90: Proceedings of the 25<sup>th</sup> annual conference on computer graphics and interactive techniques* (pp. 387-394). New York, NY: ACM.
- Sederberg, T.W., Zheng, J., Bakenov, A., Naris, A. (2003). “T-splines and T-nurcs.” *ACM Transactions on Computer Graphics*, 22(3): 477-484.
- Shackleford, J., Kandasamy, N., Sharp, G. (2010). “On developing B-spline registration algorithms for multi-core processors,” *Physics in Medicine and Biology* 52(19): 6329-6352.
- Shusharina, N., Sharp, G. (2012). “Image registration using radial basis functions with adaptive radius.” *Med. Phys.* 39(11): 6542-6549.
- Smith-Bindman, R., Miglioretti, D.L., Larson, E.B. (2008). “Rising use of diagnostic medical imaging in a large integrated health system.” *Health Affairs* 27(6): 1491-1502.
- Smith-Bindman, R., Miglioretti, D.L., Johnson, E., Lee, C., Feigelson, H.S., Flynn, M., Greenlee, R.T., Kruger, R.L., Hornbrook, M.C., Roblin, D., Solberg L.I., Vannerman, N. (2012). “Use of diagnostic imaging studies and associated radiation exposure for patients enrolled in large integrated health care systems.” *Journal of American Medical Association* 307(22): 2400-2409.

Szeliski, R., Coughlan, J. (1997). "Spline-based image registration." *Int. J. Comp. Vision* 22(3): 199-218.

Vandemeulebroucke, J., Rit, S., Kybic, J., Clarysse, P., Sarrut, D. (2011). "Spatiotemporal motion estimation for respiratory-correlated imaging of the lungs." *Med. Phys.* 38(1): 166-178.

Vandemeulebroucke, J., Bernard, O., Rit, S., Kybic, J., Clarysse, P., Sarrut, D. (2012). "Automated segmentation of a motion mask to preserve sliding motion in deformable registration of thoracic CT." *Med. Phys.* 39(2):1006-1015.

Wu, Z., Rietzel, E., Boldea, V., Sarrut, D., Sharp, G. (2008). "Evaluation of deformable registration of patient lung 4DCT with sub-anatomical region segmentations." *Med. Phys.* 35(2): 775-781.

Xie, H., Qin, H. (2001). "A novel optimization approach to the effective computation of NURBS knots." *Int. J. Shape Model* 7: 199

Xie, Y., Chao, M., Xiong, G. (2011). "Deformable image registration of liver with consideration of lung sliding motion." *Med. Phys.* 38(10): 5351-5361.

Xie, Z., Farin, G.E. (2004). "Image registration using hierarchical B-splines." *IEEE Trans. Vis. Comput. Graphics* 10(1): 85-94.

## Vita

Travis James Jacobson was born on March 21, 1984 in Roseville, California, and is an American citizen. He graduated from Airline High School, Bossier City, Louisiana in 2002. He received his Bachelor of Sciences in Physics and Mathematics from the Tulane University, New Orleans, Louisiana in 2006.

### Papers and Presentations:

T.J. Jacobson, M.J. Murphy, *Optimized knot placement for B-splines in deformable image registration*. Med. Phys., 2011 38(8): p. 4579-4582.

T.J. Jacobson, M.J. Murphy, *Non-Uniform B-Spline Deformable Image Registration*, 2012 Fall Meeting of the Mid-Atlantic Chapter of the American Association of Physicists in Medicine, Fredericksburg, Virginia.

T.J. Jacobson, M.J. Murphy, MO-F-BRA-03: *NURBS-based deformable image registration*. Med. Phys., 2012; 39(6), p. 3875. Oral Presentation at AAPM 54<sup>th</sup> Annual Meeting, Charlotte, North Carolina.

T.J. Jacobson, M.J. Murphy, TU-C-211-10: *Optimized knot placement in B-splines for Deformable Image Registration*. Med. Phys., 2011; 38(6), p. 3761. Oral Presentation at AAPM 53<sup>rd</sup> Annual Meeting, Vancouver, British Columbia.

T.J. Jacobson, M.J. Murphy, *Introduction of NURBS to deformable image registration*, 2010 Fall Meeting of the Mid-Atlantic Chapter of the American Association of Physicists in Medicine, Charlottesville, Virginia.

T.J. Jacobson, M.J. Murphy, SU-GG-I-117: *NURBS-based deformable image registration*. Med. Phys., 2010; 37(6), p. 3128. Poster at AAPM 52<sup>nd</sup> Annual Meeting, Philadelphia, Pennsylvania.

T.J. Jacobson, M.J. Murphy, *Introduction of NURBS to deformable image registration*, 2009 Fall Meeting of the Mid-Atlantic Chapter of the American Association of Physicists in Medicine, Baltimore, Maryland.