

**ENHANCEMENT AND SMOOTHING METHODS FOR
EXPERIMENTAL DATA: APPLICATION TO PIV MEASUREMENTS
OF A LAMINAR SEPARATION BUBBLE**

**M.Sc. Thesis by
Hakkı Ergün ÇEKLİ, Eng.**

Department : Mechanical Engineering

Programme: Thermo-Fluids

JANUARY 2007

**ENHANCEMENT AND SMOOTHING METHODS FOR
EXPERIMENTAL DATA: APPLICATION TO PIV MEASUREMENTS
OF A LAMINAR SEPARATION BUBBLE**

**M.Sc. Thesis by
Hakkı Ergün ÇEKİ, Eng.**

(503031112)

Date of submission : 25 December 2006

Date of defence examination: 30 January 2007

SUPERVISOR (CHAIRMAN): Assoc.Prof. Dr. Hasan GÜNEŞ

Members of the Examining Committee Prof.Dr. İsmail TEKE (Y.T.Ü.)

Assist. Prof.Dr. Murat ÇAKAN

JANUARY 2007

İSTANBUL TEKNİK ÜNİVERSİTESİ « FEN BİLİMLERİ ENSTİTÜSÜ

**DENEYSEL VERİLERİN FİLTASYONU VE ÇÖZÜNÜRLÜĞÜNÜN
ARTTIRILMASI METODLARI: LAMİNER BALONCUK AYRILMASI
PIV ÖLÇÜMLERİNE UYGUNMASI**

YÜKSEK LİSANS TEZİ

Mak. Müh. Hakkı Ergün CEKLİ

Tezin Enstitüye Verildiği Tarih : 25 Aralık 2006

TEZ DANIŞMANI : Doç.Dr. Hasan GÜNEŞ
Diğer Jüri Üyeleri Prof.Dr. İsmail TEKE (Y.T.Ü.)
Yard.Doç.Dr. Murat ÇAKAN

OCAK 2007

ACKNOWLEDGMENTS

I would like to extend my sincere thanks to a number of people without whose help and support this thesis would never have been possible.

First of all, I would like to thank Dr. Hasan Gunes for being a great advisor. I could not have imagined having a better advisor for my M.Sc. thesis.

I specifically thank Dr. Ulrich Rist for all of his absolutely invaluable help and for all brain storming sessions we had together. It has been a great pleasure working with and learning from him.

My thanks go to all of the staff of mechanical engineering faculty of Istanbul Technical University for all of their support.

Additional thanks are due to people at the Institute of Aerodynamics and Gasdynamics (IAG), Stuttgart University. It was a great collaboration and pleasure to work with all of those people and to benefit from their knowledge.

I would like to thank Julich and Tubitak that support my visit to IAG.

Additional thanks due to my friends and colleagues for the great time I had in our group. I enjoyed the atmosphere, their friendship, and their support.

Special thanks goes to my special friends Unsal Matkap, Goksel Senol and Anke Neuber for all of their support in my life.

Last but not least, the most special thanks go to my family for their support, understanding, patience and enduring my absence because of my long studies.

Istanbul, December 2006

HAKKI ERGUN CEKLI

TABLE OF CONTENTS

ACKNOWLEDGMENTS	ii
TABLE OF CONTENTS	iii
ABBREVIATIONS	v
LIST OF TABLES	vi
LIST OF FIGURES	vii
LIST OF SYMBOLS	xi
ABSTRACT	xii
ÖZET	xiii
1. INTRODUCTION	1
1.1. Thesis Organization	4
2. OVERVIEW OF PROPER ORTHOGONAL DECOMPOSITION, KRIGING INTERPOLATION, AND DIGITAL FILTERS	5
2.1. Proper Orthogonal Decomposition	5
2.2. Kriging Interpolation	7
2.2.1. Variogram and its Parameters	8
2.2.2. Selection of Variogram Model	12
2.2.3. Kriging Estimation Procedure	16
2.2.4. Inverse-distance Weighting (IdW) Interpolation	18
2.3. Digital Filtering	18
2.3.1. The z-Transform	23
2.3.2. Rational z-Transforms	24
2.3.3. Concept of Filtering	25
2.3.4. The Transfer Function	26
2.3.5. Types of Transfer Functions	26
2.3.6. Finite Impulse Response (FIR) and Infinite Impulse Response (IIR) Filters	28
2.4. Particle Image Velocimetry (PIV-Measurements)	31
3. DATA SMOOTHING	36
3.1. Data Smoothing via POD modes	36
3.2. Data Smoothing by Kriging Interpolation	42
3.2.1. A comparison of Kriging and POD smoothing	46
3.2.2. Effect of zone number on Kriging interpolation	47
3.2.3. 2D Line Kriging	48
3.3. Data Smoothing via FIR/IIR Digital Filters	50
3.4. Smoothing of test data	52
3.4.1. Creating, disturbing and smoothing test data	53

4. ESTIMATION OF MISSING PLANES	61
4.1. Estimation via POD	61
4.2. Estimation via Kriging interpolation	61
4.3. Estimation via Inverse-distance Weighting Interpolation	64
4.4. Calculation of the cross-flow velocity component	65
5. RECONSTRUCTION OF “BLACK ZONES”	68
5.1. Black zone in test data	68
5.2. Black zone in PIV data	71
6. APPLICATION OF KRIGING INTERPOLATION FOR PROBLEMS	75
6.1. Simulated Annealing	75
7. CONCLUSION	82
REFERENCES	84
APPENDIX	86
BIOGRAPHY	94

ABBREVIATIONS

CCD	–	: Charge Coupled Device
CFD	–	: Computational Fluid Dynamics
CMOS	–	: Complementary metal-oxide-semiconductor
DNS	–	: Direct Numerical Simulation
DTFT	–	: Discrete Time Fourier Transform
FFT	–	: Fast Fourier Transform
FIR	–	: Finite Impulse Response
IAG	–	: Institute für Aerodynamic und Gasdynamic
IdW	–	: Inverse-distance Weighting
IIR	–	: Infinite Impulse Response
LHS	–	: Latin Hypercube Sampling
LTI	–	: Linear Time-Invariant
MRI	–	: Magnetic Resonance Imaging
PIV	–	: Particle Image Velocimetry
POD	–	: Proper Orthogonal Decomposition
RWTH	–	: Rheinisch-Westfälischen Technischen Hochschule Aachen
SA	–	: Simulated Annealing
rms	–	: root mean square

LIST OF TABLES

	<u>Page</u>
Table 2.1: Common descriptive statistics for test data (Source: Golden Software Tutorials).....	10
Table 3.1: Number of modes and cumulative energy contribution.	39
Table 5.1: The <i>rms</i> error values for reconstruction of different size of black zones.	71

LIST OF FIGURES

	<u>Page</u>
Figure 2.1: Experimental and model variogram.....	9
Figure 2.2: A variogram with its components.....	10
Figure 2.3: Histograms of Data A (left) and Data B (right).....	11
Figure 2.4: The contours of Data A (left) and Data B (right).	11
Figure 2.5: Variograms of Data A (on the left) and Data B (on the right).	12
Figure 2.6: Different variogram models.	13
Figure 2.7: A variogram and the related correlogram.	15
Figure 2.8: Predicted point from available data.	16
Figure 2.9: (a) Input signal, (b) output of a low-pass filter with a cut-off at 80 Hz, (c) output of a high-pass filter with a cut-off at 150 Hz, (d) output of a band-pass filter with cut-offs at 80 Hz and 150 Hz, (e) output of a band-stop filter with cut-offs at 80 Hz and 150 Hz..	20
Figure 2.10: (a) A continuous-time signal, (b) a discrete-time signal.	21
Figure 2.11: Schematic representation of an LTI system.	23
Figure 2.13: The frequency responses of four types of ideal digital filters, (a) low-pass filter, (b) high-pass filter, (c) band-pass filter, (d) band-stop filter.	27
Figure 2.14: A sketch of a PIV system.	32
Figure 2.15: A photograph of airfoil SD7003.	33
Figure 2.16: A sketch of experimental set up and a photograph of the whole system.	33
Figure 2.17: A photograph of laser sheet scanning system.....	33
Figure 2.18: 3D view of the experimental data.	34
Figure 2.19: (a) 2D constant x-plane, (b) 2D constant y-plane, (c) 2D constant z-plane.....	35
Figure 3.1: Spatio-temporal expansion coefficients, symbols denote known planes and solid lines are curve fits used for interpolation.....	37
Figure 3.2: Spatio-temporal expansion coefficients $a_k(z,t)$	38
Figure 3.3: The kinetic energy content of each eigenmode.	39
Figure 3.4: Eigenmodes (non-smoothed) of flow field, from left to right: 2 nd , 6 th , 7 th , 10 th and 15 th modes, respectively.	40
Figure 3.5: Smoothed eigenmodes of flow field, from left to right: 2 nd , 6 th , 7 th , 10 th and 15 th modes, respectively.	40
Figure 3.6: The contours of the streamwise velocity ($z = 0.75$) for POD with different numbers of modes. From left to right: PIV measurements, reconstructed with 4, 6, 10, 20 original modes.	41
Figure 3.7: The contours of the streamwise velocity ($z = 0.75$) for POD with different numbers of modes. From left to right: PIV measurements, reconstructed with 4, 6, 10, 20 kriged modes, respectively.	42

Figure 3.8: The contour plots of the streamwise velocity. From left to right: The original PIV data, smoothed PIV data by exponential model, spherical model, spline model, and the Gaussian model, respectively. (Note that all plots are for $z = 0.75$ and for the first snapshot).	43
Figure 3.9: The streamwise velocity distribution on $x = 33$ ($z = 0.75$) for different variogram models.	43
Figure 3.10: The streamwise velocity distribution on $x = 33$ ($z = 0.75$) for the the Gaussian correlation model with different correlation parameters. (See the enlarged view for selected region).....	44
Figure 3.11: The contours of the streamwise velocity ($z = 0.75$) for the Gaussian model with different correlation parameters. From left to right: PIV measurements, the Gaussian smoothing with correlation parameters $\theta = 6, 3, 0.5, 0.1$, respectively.	45
Figure 3.12: The Gaussian variogram models for different correlation parameters..	46
Figure 3.13: The contours of the streamwise velocity. (a) $z = 0.75$, (b) $z = 3.75$, from left to right: original PIV data, smoothed by Kriging interpolation, smoothed by POD.	47
Figure 3.14: The contours of the streamwise velocity ($z = 0.75$) for the Gaussian model with different number of zones. From left to right: PIV measurements, 2 zone-smoothing, 4 zone-smoothing and 8 zone-smoothing, respectively. Note that red dashed lines show boundaries between zones.	48
Figure 3.15: The contours of the streamwise velocity ($z = 0.75$) for 2D line Kriging by the Gaussian model with different correlation parameters. From left to right: PIV measurements, 2D line Kriging smoothing with $\theta = 10, 5, 2$ and 0.5 , respectively.	49
Figure 3.16: The streamwise velocity distribution on $x = 30$ line ($z = 0.75$) for 2D line Kriging by the Gaussian correlation model with different correlation parameters.	50
Figure 3.17: The contours of the streamwise velocity ($z = 0.75$) for 2D IIR Butterworth filter with different cut-off frequencies. From left to right: PIV measurements, smoothed data using $f_c = 0.5, 0.3, 0.2$, and 0.1 , respectively.	51
Figure 3.18: The streamwise velocity distribution on $x = 30$ ($z = 0.75$) for low-pass IIR Butterworth filter with different cut-off frequencies.....	51
Figure 3.19: Amplitude response of the second order low-pass IIR Butterworth filter for different values of cut-off frequency.	52
Figure 3.20: Amplitude response of low-pass IIR Butterworth filter ($f_c = 0.3$) for different values of filter order.	52
Figure 3.21: (a) Actual data, (b) disturbed data, (c) smoothed data by the Gaussian model ($\theta = 0.5$).	54
Figure 3.22: (a) Actual data, (b) disturbed data ($rms = 0.05$), (c) smoothed data by the Gaussian model ($\theta = 0.1$) ($rms = 0.035$), (d) smoothed data by the Gaussian model ($\theta = 0.5$) ($rms = 0.016$), (e) smoothed data by the Gaussian model ($\theta = 1$) ($rms = 0.02$).	55
Figure 3.23: Data smoothing (a) on $y = -0.6$, $\theta = 0.5$, (b) on $y = 1.35$, $\theta = 0.5$	56

Figure 3.24: (a) Actual data, (b) disturbed data ($rms = 0.05$), (c) smoothed data by 2D line Kriging with the Gaussian model ($\theta = 0.1$) ($rms = 0.023$), (d) smoothed data by 2D line Kriging with the Gaussian model ($\theta = 0.5$) ($rms = 0.019$), (e) smoothed data by 2D line Kriging with the Gaussian model ($\theta = 1$) ($rms = 0.023$), (f) smoothed data by 2D line Kriging with the Gaussian model ($\theta = 2$) ($rms = 0.029$).	57
Figure 3.25: Data smoothing (a) on $y = -0.6$, $\theta = 0.5$, (b) on $y = 1.35$, $\theta = 0.5$	58
Figure 3.26: (a) Actual data, (b) disturbed data ($rms = 0.05$), (c) smoothed data by Kriging with the Gaussian model ($\theta = 0.5$) ($rms = 0.035$), (d) smoothed data by IIR Butterworth filter $f_c = 0.1$ ($rms = 0.085$), (e) smoothed data by IIR Butterworth filter $f_c = 0.3$ ($rms = 0.0154$), (f) smoothed data by IIR Butterworth filter $f_c = 0.5$ ($rms = 0.0227$).	59
Figure 3.27: Comparison of 2D Butterworth filter and Kriging interpolation. . (a) $y = -0.6$, (b) $y = 1.35$	60
Figure 4.1: The contours of the streamwise velocity (a) $z = 1.125$ plane, ... (b) $z = 3.375$ plane. From left to right: linear interpolation, Kriging interpolation, POD “interpolation”.	62
Figure 4.2: The contours of the streamwise velocity for $y = 5$ for the first snapshot, from top to bottom: original PIV data, enhancement of PIV data by Kriging interpolation, enhancement of PIV data by POD “interpolation”, respectively.	63
Figure 4.3: The contours of the streamwise velocity for $x = 10$. From top to bottom: original PIV data, enhanced data by Kriging interpolation, enhanced data by POD “interpolation”.	64
Figure 4.4: The contours of the streamwise velocity for $y = 10$. From top to bottom: linear interpolation; Inverse-distance Weighting interpolation ($p = 3$) and Kriging interpolation.	65
Figure 4.5: The contours of the cross-flow velocity for $y = 5$. From left to right: calculated cross-flow velocity from the original PIV data, enhanced cross-flow velocity by Kriging interpolation, enhanced cross-flow velocity by POD “interpolation”.	66
Figure 4.6: In plane velocity vectors (u, w) on plane $y = 5$. Colours denote the spanwise velocity component (v). From top to bottom: original PIV data, Kriging interpolation and POD “interpolation”.	67
Figure 5.1: Kriging interpolation for a black zone. The Gaussian variogram model is used. (a) original data, (b) Kriging interpolated data (compare the dashed black-zone), ($rms = 1.48 \cdot 10^{-3}$).	69
Figure 5.2: The original and the reconstructed function values on a selected constant line ($x = -1,05$).	69
Figure 5.3: Kriging interpolation results employing the Gaussian variogram model for black-zone, (a) original data, (b) Kriging interpolated data, (compare the dashed black zone), ($rms = 6.177 \cdot 10^{-2}$).	70
Figure 5.4: Kriging interpolation results employing the Gaussian variogram model for black-zone, (a) original data, (b) Kriging interpolated data, (compare the dashed black-zone), ($rms = 0.134$).	70
Figure 5.5: Kriging interpolation results for a black zone with noisy data. The Gaussian model is used. a) actual data, b) disturbed data with black zone, c) reconstructed data.	71

Figure 5.6: Kriging interpolation results for the contours of the streamwise velocity component, employing the Gaussian variogram with different correlation parameter for black-zone in experimental data, a) original PIV data, b) $\theta = 0.1$ c) $\theta = 0.2$, d) $\theta = 0.3$, e) $\theta = 0.5$	72
Figure 5.7: Smoothing the outside of black zone. (a) Original PIV data, b) smoothed by the Gaussian model $\theta = 0.5$	73
Figure 5.8: Kriging interpolation results for the contours of the streamwise velocity component, employing the Gaussian variogram with different correlation parameter for black-zone on the smoothed experimental data, a) original PIV data, b) smoothed PIV data c) $\theta = 0.1$ d) $\theta = 0.2$, e) $\theta = 0.3$, f) $\theta = 0.5$	74
Figure 6.1: Temperature of system.	78
Figure 6.2: Probability values, a) $T_0 = 1$, $c = 1$, b) $T_0 = 1$, $c = 2$, c) $T_0 = 10$, $c = 1$	79
Figure 6.3: Optimization process, a) $T_0 = 1$, $c = 1$, b) $T_0 = 1$, $c = 2$, c) $T_0 = 10$, $c = 1$	80
Figure A.1: Stencil for numerical derivation and integration.	86
Figure A.2: The contours of the actual and the final integrated data, continuous line denotes the actual data and dashed line denotes the final integrated data, (a) low-resolution, (21x21 grid points), (b) high-resolution, (41x41 grid points).	88
Figure A.3: Analytical and numerical calculated values of variable and its derivative on constant line $y = -0.6$, (a) low-resolution, (21x21 grid points), (b) high-resolution, (41x41 grid points). left: variable, right: derivative.....	88
Figure A.4: The Gaussian variogram model for different correlation parameters.	90
Figure A.5: Sinus1 variogram model for different correlation parameters.	90
Figure A.6: Sinus2 variogram model for different correlation parameters.	90
Figure A.7: The contours of the variable of the actual, disturbed and smoothed data with the Gaussian model for different correlation parameter, (a) actual data, (b) disturbed data, ($rms = 11.09\%$), (c) $\theta = 3$, ($rms = 4.24\%$), (d) $\theta = 1$, ($rms = 3.46\%$), (e) $\theta = 0.5$, ($rms = 2.87\%$), (f) $\theta = 0.2$, ($rms = 2.49\%$).....	91
Figure A.8: The contours of the variable of the actual, disturbed and smoothed data with the sinus1 model for different correlation parameter, (a) actual data, (b) disturbed data, ($rms = 11.09\%$), (c) $\theta = 0.1$ ($rms = 2.44\%$).	92
Figure A.9: The contours of the variable of the actual, disturbed and smoothed data with the sinus2 model for different correlation parameter, (a) actual data, (b) disturbed data, ($rms = 11.09\%$), (c) $\theta = 0.1$, ($rms = 2.2\%$).	92
Figure A.10: The contours of the variable of the actual, disturbed and smoothed data with different variogram models for higher rms value, (a) actual data, (b) disturbed data ($rms = 18.4\%$), (c) The Gaussian model, $\theta = 0.01$, ($rms = 4.48\%$), (d) sinus1 model, $\theta = 1$, ($rms = 6.03\%$), (e) sinus2 model, $\theta = 1$ ($rms = 5.297\%$).....	93

LIST OF SYMBOLS

$\mathbf{a}$	: Range
$\mathbf{a}_k$	: Spatio-temporal coefficients
$\mathbf{b}$	: Coefficient
$\mathbf{c}$	: Cooling parameter
$\mathbf{c}_0$	: Nugget
$\mathbf{c}_1$	: Sill
$\mathbf{E}_T$	: Total energy of flow
$\mathbf{e}$	: The best value
$\mathbf{e}_n$	: The last estimated value
$\mathbf{f}_c$	: Cut-off frequency
$\mathbf{G}(\mathbf{z})$	: z-transform of a sequence
$\mathbf{h}$	: Distance
$\mathbf{h}(\mathbf{t})$	: Impulse response
$\mathbf{i}$	: Sample location
$\mathbf{L}$	: Number of pairs
$\mathbf{N}$	: Number of the most energetic POD modes
$\mathbf{n}$	: Number of available points
$\mathbf{P}$	: Probability function
$\mathbf{P}(\mathbf{z})$	: Numerator
$\mathbf{p}$	: Inverse-distance exponent
$\mathbf{S}$	: Selected number of snapshots
$\mathbf{t}$	: Time
$\mathbf{u}$	: The streamwise velocity
$\mathbf{v}$	: The spanwise velocity
$\mathbf{w}$	: The cross-flow velocity
$\mathbf{W}_i$	: Weight at point I
$\alpha_{k,i}$	: Eigenvalue vector
δ	: Discrete delta function
$\Delta \mathbf{t}$	: Time difference
$\Delta \mathbf{x}$	: Particle displacement
ϕ_k	: POD modes
γ	: Variance
λ	: Eigenvalues
ρ	: Langrange multiplier
θ	: Correlation parameter

SUBSCRIPTS

$\mathbf{P}$	: Estimated
$\mathbf{A}$	: Actual

ENHANCEMENT AND SMOOTHING METHODS FOR EXPERIMENTAL DATA: APPLICATION TO PIV MEASUREMENTS OF A LAMINAR SEPARATION BUBBLE

ABSTRACT

Methods for data reconstruction and spatial enhancement/smoothing of experimental data for a transitional boundary layer with laminar separation bubble are investigated.

First, Proper Orthogonal Decomposition (POD) is applied to PIV data to extract the POD modes, which are projected onto the experimental data in order to obtain model coefficients. These model coefficients are then used to reconstruct and interpolate the experimental data employing the ‘smoothed’ POD modes.

Next, we investigate the possibility of enhancing/smoothing of PIV data using Kriging interpolation. We have demonstrated that both POD and Kriging methods can be used effectively to enhance/smooth the experimental data, thus a noise-free reconstruction of PIV data can be achieved. The performance of several correlation functions (e.g., the Gaussian, exponential, spherical, etc.) in applying Kriging interpolation is studied in detail. It has been shown that, with the appropriate selection of the correlation function and its parameters, one can control the ‘degree’ of smoothness in a robust way.

Furthermore, digital filters which are widely used in signal processing have been investigated in order to compare with POD and Kriging methods. In this respect, it has been found that implementation of a Line Kriging method closely compares with the well-known digital filters (e.g., Butterworth filter).

It has also been shown that Kriging can be a viable ingredient in constructing effective optimization algorithms. Possibilities of using Kriging interpolation in reconstruction of black zones (large continuous regions of missing data) are also discussed.

DENEYSEL VERİLERİN FİLTRASYONU VE ÇÖZÜNÜRLÜĞÜNÜN ARTTIRILMASI METODLARI: LAMİNER BALONCUK AYRILMASI PIV ÖLÇÜMLERİNE UYGULANMASI

ÖZET

Deneyssel olarak elde edilen sınır tabaka geçiş bölgesi içindeki laminar baloncuk ayrılmasının incelenmesi için yapılan deneylerde elde edilen deneyssel verilerin konumsal olarak zenginleştirilmesi, veri geri kazanımı ve parazitlerinin giderilmesi için çeşitli yöntemler incelenmiştir.

İlk olarak Uygun Ortogonal Ayrıklaştırma (Proper Orthogonal Decomposition POD) metodu model katsayılarının belirleneceği POD modlarının çıkarılması için PIV ölçümlerinden elde edilen veri setine uygulandı. Bu model katsayıları deneyssel verinin geri kazanımı ve interpolasyonu için ‘düzgünleştirilmiş’ POD modları ile kullanılmıştır.

Kriging interpolasyonunu kullanarak PIV den elde edilen verilerin zenginleştirilmesi ve düzgünleştirilmesi olanakları araştırılmıştır. Çalışmaların sonucunda görülmüştür ki POD ve Kriging metodlarının her ikisi de deneyssel verilerin zenginleştirilmesi ve düzgünleştirilmesi için etkin olarak kullanılabilirler, bu sayede parazitsiz bir veri seti elde edilebilecektir. Kriging interpolasyonunda çeşitli korelasyon fonksiyonlarının performansları ayrıntılı olarak incelenmiştir. Korelasyon fonksiyonunun ve parametresinin uygun bir şekilde seçilmesi ile veri düzgünlüğünün derecesinin etkin bir şekilde kontrol edilebileceği görülmüştür.

Ayrıca, sinyal işleme analizlerinde sıklıkla kullanılan sayısal filtreler POD ve Kriging metodlarının karşılaştırılması için incelenmiştir. Bu çalışmalar sırasında, sayısal filtreler ile yakın olarak örtüşen Çizgisel Kriging uygulaması geliştirilmiştir.

Ek olarak, Kriging interpolasyonunun etkin optimizasyon algoritmalarının geliştirilmesinde kullanılabileceği görülmüştür. Ayrıca Kriging interpolasyonunun kara delikler olarak adlandırılan geniş sürekli kayıplı veri alanlarının geri kazanılması için kullanılabiliirliği değerlendirilmiştir.

1. INTRODUCTION

With the recent development of state-of-the-art quantitative data measurement techniques such as particle image velocimetry (PIV) and magnetic resonance imaging (MRI), experimentalists now obtain flow (image) data with increasingly well resolved spatio-temporal accuracy, while ever expanding computational methods and hardware resources provide the computing community with even more highly resolved spatio-temporal numerical data compared to quantitative measurement techniques. These developments in experimental techniques and simulations open the possibility for integrating seamlessly simulation and experiment. At the heart of this integration is the ability to reconstruct flow fields from a finite number of experimental measurements at a controlled level of accuracy. Even for simple experimental measurements, information may be missing due to the shadowing of an obstructed view or proximity to the boundaries or the frequency of the measurements may be below the required spatial or temporal resolution for a reliable flow data visualisation and analysis. Therefore, it is necessary to work with so-called “gappy data” to find the best estimate of the missing data points and/or improve the accuracy of the spatio-temporal resolution of measurements for further post processing (e.g., visualization, feature extraction and quantification) of the data.

When compared with the numerical data, an important feature of experimental data is that it contains some undesired background noise and turbulence. In three dimensional complex flows, structure extraction and vortex identification methods are of particular importance in order to visualize and understand the flow physics. In addition, many of the data visualization methods involve the evaluation of the derivatives of the field variables, which further increase the noise level of the experimental data due to inevitable numerical differentiation errors. Therefore, utilization of various filtering as well as resolution enhancement techniques is usually required on the experimental data (**Linnick & Rist, 2004**). And, an integration of such techniques and experiments is necessary.

In this thesis, we investigate the ability of Kriging and Proper Orthogonal Decomposition (POD) techniques to smooth and enhance of resolution of experimental data, in detail. The first method, Kriging is an effective statistical estimation procedure which has been named after D.G. Krige, a South African mine engineer who developed the procedure in order to predict mine ore ground water reserves more accurately from multi-point measurements. Kriging is a statistical tool useful in many disciplines such as geology, thermo-fluid systems, process engineering, environment and medicine. Kriging is used to estimate unknown values from data observed at known locations. A variogram is usually constructed to account for the spatial variation of known data. Kriging gives an unbiased estimate for the unknown value using a weighted linear combination of the available data (**Davis, 2002**). Recently, Kriging interpolation has been successfully applied to data recovery and reconstruction of randomly generated laminar gappy flow fields of uniform flow past a circular cylinder (**Venturi & Karniadakis, 2004**), (**Gunes et al., 2006**). A recent study showed that Kriging interpolation can be used successfully for resolution enhancement and for reconstruction of large spatial gappiness for a mixed convection data (**Cekli & Gunes, 2006**). Kriging interpolation has been also applied to large spatial gappiness or for flow fields with black zones with a considerable success for relatively smooth data (**Piskin & Gunes, 2005**).

The second method that will be investigated is Proper Orthogonal Decomposition introduced by **Lumley (1967)**. It has long been popular as a means of extracting the most energetic eigenfunctions (coherent structures) from a flow (or temperature) field data. It has been proven mathematically by **Sirovich (1987)** that POD eigenfunctions are optimal in terms of kinetic energy of the flow compared to any other basis. The POD eigenfunctions also capture the essential dynamics of the flow and provide insight into the nature of the flow and its instabilities. POD is applied to many fields in diverse areas such as thermo-fluids, meteorology, ocean sciences, chemical sciences, physiology and biometrics.

In this thesis we have applied the aforementioned methods to smooth and increase the resolution of a data set obtained from scanning Particle Imaging Velocimetry (PIV) measurements of a laminar separation bubble on an airfoil SD7003. These measurements have been performed at the Institute of Aerodynamics, RWTH

Aachen University, Aachen, Germany (**Burgmann et al., 2006**). Scanning PIV is applied to the laminar turbulent transition zone in a laminar separation bubble to investigate the spanwise structure and dynamics of the roll-up of vortices within the bubble. The laminar flow separation with turbulent reattachment is studied on the suction side of the airfoil (chord length $c = 20$ cm) at Reynolds number 20000 in a Göttingen type water tunnel which has a 30cm x 25cm cross section. For detailed information about the experimental setup and results, we refer to **Burgmann et al., (2006)**. Because of practical restrictions of PIV measurements, it was not possible to obtain a highly resolved data set in all flow directions. Particularly, the PIV data consisted of 49 grid points in the streamwise direction (x), 104 grid points in the spanwise direction (y), and only 5 grid points in the cross-flow (wall normal) direction (z). It can be said that, in x and y -directions we have enough grid points for further post processing and for visualization methods. However, we obviously need more grid points in z -direction (wall normal direction) in order to visualize the flow field faithfully and to reduce the numerical errors for further data analyses, such as to implement vortex identification and traction methods.

In this thesis, we have investigated Kriging and POD techniques in detail and assessed their effectiveness. In addition, in order to compare with other techniques, several digital filters (e.g., Butterworth or FFT-based filters), frequently used in signal processing phenomena have been used and also applied to smooth noisy data. It is clear that these filters can only be used for smoothing of noisy data. It is, of course, not possible to use them to enhance the resolution of a data set. In this thesis, with respect to Kriging and POD procedures, we also introduce some new concepts (e.g., Line Kriging, pseudo-snapshots, simulated annealing optimization via Kriging) and investigate for data smoothing and resolution enhancement.

Kriging has been also applied to reconstruct so-called “black zones” (large continuous regions of missing data) in flow fields. Furthermore, an effective meta-algorithm for optimization problems which is a combination of Kriging and Simulated Annealing techniques has been briefly investigated. Simulated Annealing (SA) uses nature’s own minimization algorithm and is good for global optimization problems in the presence of many local minima and maxima.

In these investigations it was not possible to compare and determine the capabilities of techniques based on the PIV measurements in all cases. Therefore, we additionally used synthetic data sets (test data) with known analytical solution. Thus, we could perform detailed analyses in a well-controlled manner in order to assess the effectiveness of each procedure.

1.1. Thesis Organization

This thesis is organized as follows. Chapter 2 gives a general overview of POD and Kriging techniques and also basic concepts of digital filters. In Chapter 3 we present the results and comparison of these methods for data smoothing problem. Further, results and comparison for resolution enhancement phenomena are given in Chapter 4. In Chapter 5, we give the application of Kriging to reconstruct the black-zones. Chapter 6 is dedicated to investigation of using Kriging for optimization problems. We conclude in Chapter 7 with a brief summary and possible directions for future work. Finally, in Appendix, the aforementioned procedures are applied for a synthetic test data.

2. OVERVIEW OF PROPER ORTHOGONAL DECOMPOSITION, KRIGING INTERPOLATION, AND DIGITAL FILTERS

2.1. Proper Orthogonal Decomposition

The proper orthogonal decomposition (POD) is a powerful method for system identification to obtain low-dimensional descriptions for multidimensional systems. The POD provides a basis for the modal decomposition of a system of functions, usually data obtained from measurements or numerical simulations. The basis functions retrieved are called POD modes. It provides an efficient way of capturing the dominant components of a multidimensional system and representing it to the desired precision by using the relevant set of modes, thus reducing the order of the system. Thus, POD can be used for low-dimensional representation, modeling and control of a flow field. In addition, POD can be used for natural noise filtering and data enhancement of experimental data.

POD is a powerful tool in data analysis in that it sorts the data by the energy, with the largest eigenvalue corresponding to the eigenvector representing the highest energy of the system. It is a way to represent most of the total energy associated with a large set of data using only a fraction of the data set, by partitioning the variance amongst eigenvectors. This can allow the analyst to examine the data in terms of where the most variability occurs. POD is a robust technique used for recognizing or identifying key dominant features in a data set. When applied to flow, this technique has the ability to identify structures that contribute most to the energy of the flow. The decomposition of the flow by this technique provides a set of modes that represent flow structures containing most of the energy. These modes can be used to reconstruction or interpolation of flow data.

POD is widely used in thermo-fluids systems, meteorology, heat transfer, aerodynamics, turbomachinery, biometrics (e.g., face recognition), and reconstruction of gappy data.

POD was introduced by **Lumley (1967)** to define coherent structures in turbulent flows in the field of fluid mechanics. However, because of the lack of highly resolved flow fields the application of POD has not been developed for years. Recent developments in quantitative measurement systems like Particle Image Velocimetry (PIV) and Direct Numerical Solutions (DNS) yield highly resolved data sets, thus application of POD for complicated flow fields in two and three dimensions is possible now. Additionally, **Sirovich (1987)** introduced the snapshot version of POD that became an effective procedure for two- and three-dimensional flow fields.

The POD modes can be determined by linear combination of time-dependent velocity flow field as follows (**Sirovich, 1987**),

$$\phi_k^\circ(x, y) = \sum_{i=1}^S \alpha_{k,i} \dot{V}(x, y, t_i) \quad (2.1)$$

where, S denotes the selected number of snapshots, $V(x, y, t_i)$ is the i^{th} snapshot of velocity field and $\alpha_{k,i}$ is the k^{th} eigenvalue vectors elements of the correlation matrix $C:(S \times S)$ of the eigenvalue problem is given below,

$$C\alpha = \lambda\alpha \quad (2.2)$$

The matrix C in Eq.(2.2) can be determined by using the velocity field as follows (**Sirovich, 1987**),

$$C_{ij} = \frac{1}{S} \iint \dot{V}(x, y, t_i) \dot{V}(x, y, t_j) dx dy \quad (2.3)$$

Since the matrix C is symmetric and positive definite the eigenvalues are real non-negative numbers. Thus, eigenvalues of matrix C can be sorted as $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \dots \geq \lambda_N \geq 0$.

Here, $\sum_{i=1}^S \lambda_i = E_T$ is the total energy of the flow and each eigenvalue represents a contribution to total flow energy to the corresponding eigenvalue function (**Berkooz et al., 1993**).

Having obtained eigenmodes, the velocity field $V(x, y, t_i)$ can be represented by a series expansion with normalized POD modes as follows,

$$\overset{\circ}{V}(x, y, t) = \sum_{k=1}^N \overset{\circ}{a}_k(t) \overset{\circ}{\phi}_k(x, y) \quad (2.4)$$

where N denotes the number of most energetic POD modes and generally $N \ll S$. With the orthogonal property of the temporal coefficients of POD modes $a_k(t)$ can be determined by Eq.(2.5),

$$a_k(t) = \iint V(x, y, t) \phi(x, y) dx dy \quad k = 1, 2, \dots, N \quad (2.5)$$

Eq.(2.5) is called the reconstruction formula.

2.2. Kriging Interpolation

Kriging is an estimation technique which has been named after its inventor Prof. Dr. D.G. Krige in early 1950s who is a South African mining engineer who worked in the field of geostatistics. His empirical work to evaluate mineral resources (**Krige, 1951**) was formalised in the 1960s by French engineer Prof. Georges Matheron at the Centre de Morphologie Mathematique in Fontainebleau, France (**Matheron, 1962**).

For the past several decades, Kriging has been used with success in geology and environmental engineering to interpolate the “regionalized” data. Recently, Kriging interpolation has been successfully applied to data recovery and reconstruction of randomly generated laminar gappy flow fields of uniform flow past a circular cylinder (**Venturi & Karniadakis, 2004**), (**Gunes et al., 2006**). Kriging

interpolation has been also applied to large spatial gappiness or for flow fields with black zones with a considerable success for relatively smooth data.

Kriging is an unbiased estimation procedure which uses known values and a variogram to determine unknown values. Based on the variogram, optimal weights are assigned to known values in order to calculate the data at unknown points. Since the variogram changes with distance, the weights depend on the known sample distribution.

A key concept of Kriging is that of the ‘regionalized variable’, which has properties intermediate between a truly random variable and one that is completely deterministic. Unlike random variables, regionalized variables have continuity from point to point, but the changes in the variable are so complex that they cannot be described by any tractable deterministic function **(Davis, 2002)**. Even though a regionalized variable is spatially continuous, it is not usually possible to know its value everywhere. Instead, its values are known only through a sample of observations that are taken at specific locations.

First of all, in order to estimate by Kriging interpolation, the spatial relations among known points have to be determined by a variogram.

2.2.1. Variogram and its Parameters

The variogram characterizes the spatial continuity or roughness/smoothness of a data set **(Davis, 2002)**. The variogram analysis consists of constructing an experimental variogram from the data and fitting a variogram model to the experimental variogram. The experimental variogram is calculated by averaging one half the difference squared of the values over all pairs of observations with the specified separation distance h and possible direction,

$$\gamma(h) = \frac{1}{2L} \sum_i^L (z_i - z_{i+h})^2 \quad (2.6)$$

where, z_i is the value of the variable at point i , and z_{i+h} is the value of the variable at h separation distance away from point i . γ is called the variance. Once variances are calculated for different values of h , results can be plotted in the form of a variogram.

This plot is usually called an experimental variogram. Next, an appropriate curve is fitted to the experimental results. An experimental and its model variogram are shown in Figure 2.1.

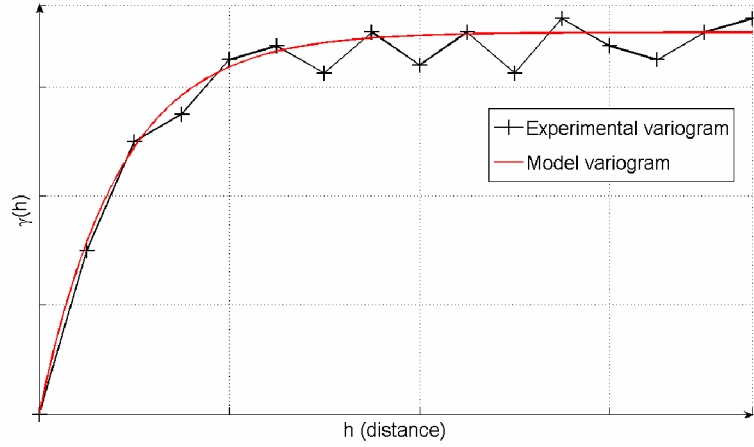


Figure 2.1: Experimental and model variogram.

The variogram model is usually chosen from a set of mathematical functions that describe the spatial relationship. The appropriate model is chosen by matching the shape of the curve of the experimental variogram to the shape of the curve of the mathematical function (i.e., polynomial, exponential, the Gaussian, etc.) **(Davis, 2002), (Deutsch & Journel, 1992), (Isaaks & Srivastava, 1989).**

When we look at the variogram, if distance h between sample points is zero, it means the point is being compared itself, the variance is zero. If distance h is a small number, it means the sample points are closed to each other, the points being compared tend to be very similar, and the variance will be a small number. As the distance h increased, the points being compared are less and less closely related to each other and their differences become larger, resulting in larger values of γ . At some distance the points being compared are so far apart that they are not related to each other, and their squared differences become equal in magnitude to the variance around an average value. The variogram no longer increases and it develops a flat region called a sill. The distance at which the variance approaches the sill value is referred to as the range of the regionalized variable, and defines a neighbourhood within which all locations are related to no another. In other words, small values of distance tend small values of γ , and a small value of γ means a higher correlation. On the other hand, large values of distance give higher γ and it means lower correlation. While building model for estimation processes there may be small errors even for

known values. This is accounted into variogram model by a new parameter called “nugget effect”. With nugget effect, the variogram model does not start from origin, i.e., when h is zero γ value is a portion of sill value. In Figure 2.2 variogram and its components such as sill, range and nugget effect are shown.

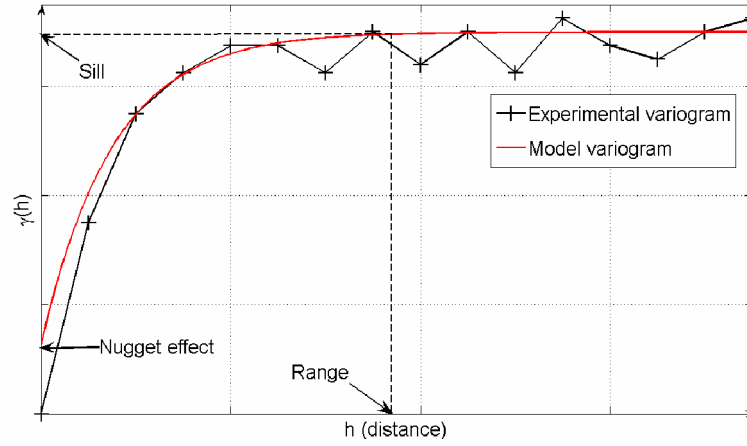


Figure 2.2: A variogram with its components.

In order to represent distinctive feature of a variogram we analyze statistical properties of two different synthetic data sets which are named Data A and Data B (Data A and B are taken from **Golden Software Tutorials**). Some common descriptive statistics for these two data sets are given in Table 2.1. The histograms for these two data sets are represented in Figure 2.3. The contour plots of these data sets are given in Figure 2.4.

Table 2.1: Common descriptive statistics for test data (**Source: Golden Software Tutorials**).

	Data A	Data B
Data Points	15251	15251
Average	100	100
Standard Deviation	20	20
Median	100.35	100.92
10 Percentile	73.89	73.95
80 Percentile	125.62	124.72

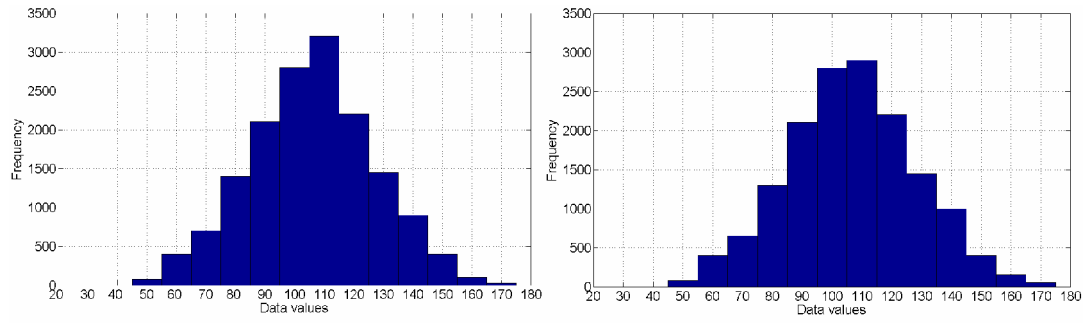


Figure 2.3: Histograms of Data A (left) and Data B (right).

By looking these statistical properties and histograms, it can be said that these two data sets are almost identical. However, as it can be seen by comparing their contour plots these two data sets are significantly different. Note that the difference cannot be captured by the common descriptive statistics and histograms. Note that we can not say that Data A is more variable than Data B, since the standard deviations for the two data sets are the same, as are the magnitudes of highs and lows. Data B is more continuous than Data A and changes in Data A are more rapid than the ones in Data B.

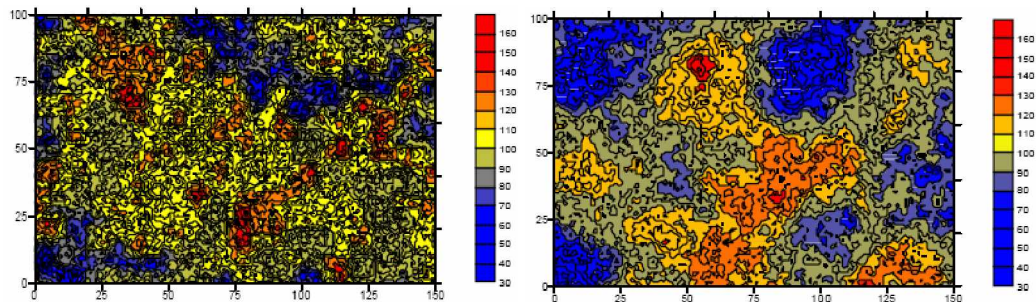


Figure 2.4: The contours of Data A (left) and Data B (right), (Source: Golden Software Tutorials).

The variograms for these two data sets are given in Figure 2.5. When comparing the variograms of these two data sets, the difference can be seen clearly. First of all, we have to look initial slopes of model variograms. The curve of variogram of Data A is more vertical than Data set B. It means that the first variogram gives higher γ values in a small distance that suggest lower correlation. It is agreeable when variograms of Data A and Data B compared with their contour plots. The gradients are higher in Data A so its variogram is more vertical.

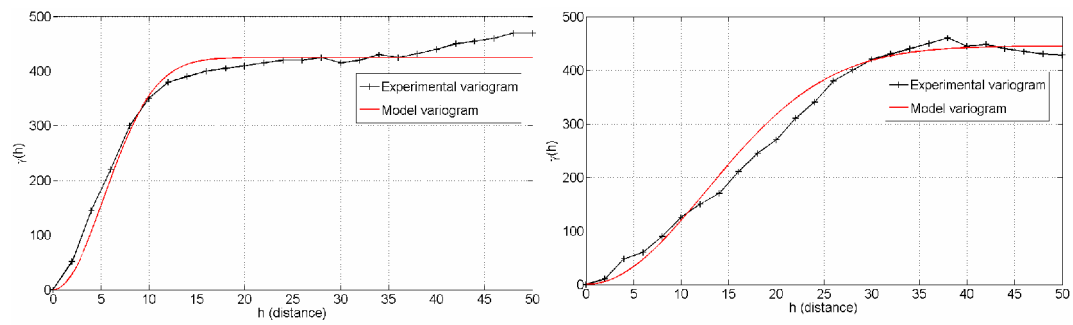


Figure 2.5: Variograms of Data A (on the left) and Data B (on the right), (Source: Golden Software Tutorials).

The variogram is used to determine the spatial-dependence between values of known points and also between a new unknown point and known points. The selection of a suitable variogram model is a crucial step in Kriging procedure because it has an important effect on the weights and estimation error.

2.2.2. Selection of Variogram Model

After an experimental variogram has been built, the next step is to fit a model variogram to it. The variogram model is usually chosen from a set of mathematical functions such as the Gaussian, exponential, spherical, etc. The appropriate model is chosen by matching the shape of the curve of the experimental variogram to the shape of the curve of the mathematical function. Selection of a variogram model is the most important step for Kriging interpolation, because it has direct effect on estimation results. By selecting different variogram models estimation results can be controlled. Selection of a certain variogram model should depend on physical reasoning and needs experience. The importance and effect of the variogram model will be discussed in the next sections in detail.

There are different kinds of model variograms. The most common ones, the spherical, exponential and the Gaussian model are shown for comparison in Figure 2.6.

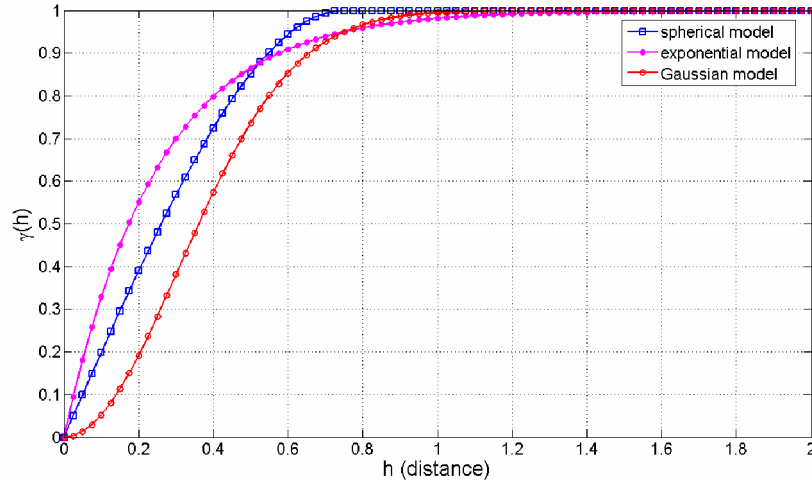


Figure 2.6: Different variogram models.

In this thesis in Figure 2.6 and next variogram plots to make better comparison γ and h values have been normalized to 1 and 2, respectively.

As can be shown in Figure 2.6 that the spherical and exponential models have linear behaviour near to origin, whereas the Gaussian model has a parabolic behaviour. This behaviour near to origin has a considerable effect on Kriging because small distances are very important to distribute correlations to neighbouring points. The spherical model reaches to sill when distance equals to range; on the other hand, the exponential and the Gaussian models reach to sill only asymptotically. The spherical and exponential models have similar shapes (i.e., linear behaviour) from origin to sill; however, the Gaussian model has a different shape and an inflection point. As it can be seen in the Figure 2.6 the Gaussian model gives more correlation to larger distance. To make it clear; for example, at distance 0.2 the Gaussian model gives $\gamma = 0.2$, the spherical and exponential models give $\gamma = 0.4$ and $\gamma = 0.55$, respectively. Because of larger γ means lower correlation it can be said that the Gaussian model gives more correlation for point far away. And this is the idea behind smoothing phenomena which will be analyzed in next chapters.

Mathematical functions of these models are given in Eqs.(2.7)-(2.20).

Exponential model:

$$\gamma(h) = c_0 + c_1 \left[1 - \exp\left(-3 \frac{h}{a}\right) \right] \quad (2.7)$$

Spherical model:

$$\gamma(h) = c_1, \text{ if } \frac{h}{a} > 1 \quad (2.8)$$

$$\gamma(h) = c_0 + c_1 \left[1.5 \frac{h}{a} - 0.5 \left(\frac{h}{a} \right)^3 \right], \text{ if } \frac{h}{a} \leq 1 \quad (2.9)$$

Gaussian model:

$$\gamma(h) = c_0 + c_1 \left[1 - \exp \left(-3 \left(\frac{h}{a} \right)^2 \right) \right] \quad (2.10)$$

Note that, where c_0 is nugget effect, c_1 is sill, and a is range; and $c_0 \geq 0, c_1 \geq 0, a \geq 0$.

There are also some other variogram models, and can be used for different aims. If necessary also there may add some different models as well.

Linear model:

$$\gamma(h) = 0, \text{ if } h = 0 \quad (2.11)$$

$$\gamma(h) = c_0 + b \times h \quad (2.12)$$

$$b \geq 0 \quad (2.13)$$

Power model:

$$\gamma(h) = 0, \text{ if } h = 0 \quad (2.14)$$

$$\gamma(h) = c_0 + b \times h^\Lambda \quad (2.15)$$

$$0 \leq \Lambda \leq 2 \quad (2.16)$$

Cubic model:

$$\gamma(h) = c_1, \text{ if } \frac{h}{a} > 1 \quad (2.17)$$

$$\gamma(h) = c_0 + c_1 \left[7 \left(\frac{h}{a} \right)^2 - \frac{35}{4} \left(\frac{h}{a} \right)^3 + \frac{7}{2} \left(\frac{h}{a} \right)^5 - \frac{3}{4} \left(\frac{h}{a} \right)^7 \right], \text{ if } \frac{h}{a} \leq 1 \quad (2.18)$$

Cosines model:

$$\gamma(h) = c_0 + \left[1 - e^{-\left(\frac{h}{a} \right)^2} \times \cos \left(\frac{h}{a} \right) \right] \quad (2.19)$$

Sinus model:

$$\gamma(h) = c_0 + \left[1 - \frac{\sin \left(\frac{h}{a} \right)}{\left(\frac{h}{a} \right)} \right] \quad (2.20)$$

Here we should also mention that one can equally use the correlogram (C) instead of the variogram, γ . The correlogram is typically defined as $C = 1 - \gamma$. However, our numerical experience reveals that the evaluation of the coefficients is more robust via the correlogram as the coefficient matrix is less prone to singularities. In Figure 2.7 a variogram and the related correlogram are given.

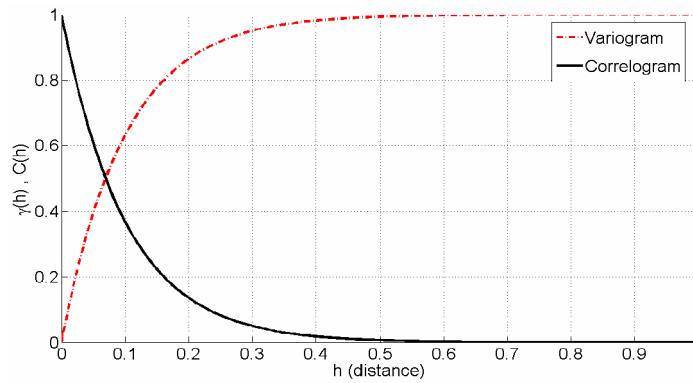


Figure 2.7: A variogram and the related correlogram.

2.2.3. Kriging Estimation Procedure

Once a variogram has been constructed, Kriging estimates a value for any desired new (unknown) point. Kriging gives a weight for each known point to estimate a new point as in Eq.(2.21), and these weights depend on the spatial dependence of the data set (i.e., on the variogram). In Figure 2.8 a symbolic way of prediction is illustrated.

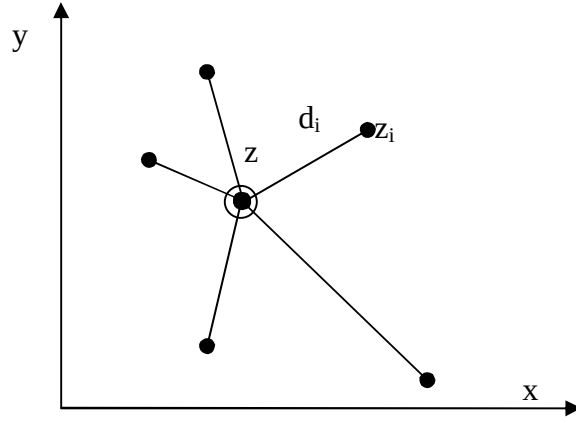


Figure 2.8: Predicted point from available data.

$$z(x, y) = \sum_{i=1}^n W_i z_i \quad (2.21)$$

It is apparent that a point with a small distance from estimated point has a large weight; in contrast, a point with a large distance has a small weight. z_p is the estimated point and z_i are the known points (design). Employing the variogram, the weights $W_1, W_2, \dots, W_n$ can be found by solving the following system of linear equations,

$$\begin{bmatrix} \gamma(h_{11}) & \gamma(h_{12}) & \dots & \gamma(h_{1n}) & 1 \\ \gamma(h_{21}) & \gamma(h_{22}) & \dots & \gamma(h_{2n}) & 1 \\ \vdots & \vdots & \dots & \vdots & \vdots \\ \gamma(h_{n1}) & \gamma(h_{n2}) & \dots & \gamma(h_{nn}) & 1 \\ 1 & 1 & \dots & 1 & 0 \end{bmatrix} \begin{bmatrix} W_1 \\ W_2 \\ \vdots \\ W_n \\ \rho \end{bmatrix} = \begin{bmatrix} \gamma(h_{1p}) \\ \gamma(h_{2p}) \\ \vdots \\ \gamma(h_{np}) \\ 1 \end{bmatrix} \quad (2.22)$$

where $\gamma(h_{ij})$ is the variance between points i and j , and $\gamma(h_{ip})$ is the variance between point i and the estimated point p . The linear system above has to be solved to obtain the weights. $\gamma(h_{ij})$ and $\gamma(h_{ip})$ can be taken from the variogram or calculated from its mathematical function. As $h_{ij} = h_{ji}$, the coefficient matrix is symmetric and diagonal terms are equal to zero. The coefficient matrix depends only on the design data points. Thus, once the coefficient matrix has been built it can be used to predict any desired points. Note that in Eq.(2.22), we introduce a new variable called the Lagrange multiplier ρ , to impose a minimum estimation error (Isaaks & Srivastava, 1989). This, in turn, allows us to have a constraint on the weights as follows,

$$\sum_{i=1}^n W_i = 1 \quad (2.23)$$

This version of the Kriging is often called ordinary Kriging. Without this constraint on weights the method is called the simple Kriging.

One of the advantages of the Kriging procedure is that the estimation error can be calculated. The estimated value is different from its actual value as in Eq.(2.24),

$$\varepsilon = (z_p - z_A) \quad (2.24)$$

where z_p is the estimated value and z_A is the actual value.

The estimation variance is,

$$s_p^2 = W_1\gamma(h_{1p}) + W_2\gamma(h_{2p}) + \dots + W_n\gamma(h_{np}) \quad (2.25)$$

and as its square root is the standard error of the estimate.

The important aspects of Kriging method are that it gives a minimum estimation variance as well as an unbiased estimation, and an error value for the estimation. Kriging is therefore called B.L.U.E (best linear unbiased estimator) as reported by

(Davis, 2002), (Deutsch & Journel, 1992), (Isaaks & Srivastava, 1989). Kriging is the best estimation method because it gives a minimum estimation error.

2.2.4. Inverse-distance Weighting (IdW) Interpolation

A disadvantage of Kriging is that it is relatively slow and may lead to computer memory problems if the number of design (known) points is large. Therefore, we also investigated if Inverse-distance Weighting can be an alternative to Kriging as it does not require the construction of a coefficient matrix and therefore it is computationally fast and requires little memory. That is, in IdW interpolation, the weights depend only on distance, independent of the spatial distribution of function. The weights for IdW can be calculated as,

$$z(x, y) = \sum_{i=1}^n W_i z_i \quad (2.26)$$

$$W_i = \frac{\left(\frac{1}{d_i}\right)^p}{\sum_{i=1}^n \left(\frac{1}{d_i}\right)^p} \quad (2.27)$$

where, d_i is the distance between estimated and i^{th} sample location, and p is the inverse-distance exponent typically between ($2 < p < 5$). In contrast to Kriging the choice of the weights W_i is fully empirical and does not vary with function. The numerical investigations showed that Inverse-distance Weighting Method cannot be an alternative to Kriging, as it is discussed in next chapters.

2.3. Digital Filtering

In this thesis, digital filtering procedures have been applied for data smoothing process. These filtering procedures are widely used in signal processing analyses. Filtering is used to pass certain frequency components in a signal through the system without any distortion and to block other frequency components. The range of frequencies that is allowed to pass through the filter is called the pass band, and the

range of frequencies that is blocked by the filter is called the stop band. There are several types of filters, and each one has different properties. In most cases, the filtering operation for analogue signals is linear and is described by the convolution integral **(Mitra, 2001)**,

$$y(t) = \int_{-\infty}^{\infty} h(t-\tau)x(\tau)d\tau \quad (2.28)$$

where, $x(t)$ is the input signal and $y(t)$ is the output of the filter characterized by an impulse response $h(t)$.

A type of filter which is called low-pass filter passes low-frequency components below a certain specified cut-off frequency f_c and blocks all high-frequency components of a signal above f_c . A high-pass filter passes all high-frequency components above a certain cut-off frequency f_c and blocks of low-frequency components below f_c . A band-pass filter passes all frequency components between two cut-off frequencies f_{c1} and f_{c2} and blocks all other frequency components. A band-stop filter blocks all frequency components between two cut-off frequencies f_{c1} and f_{c2} and passes all of other frequency components **(Mitra, 2001)**. In Figure 2.9 outputs of different kinds of filters can be seen. In Figure 2.9 (a) a signal composed of the sinusoidal components of frequencies 50 Hz, 110 Hz, and 210 Hz, that to be filtered is shown. In Figure 2.9 (b-e) different types of filter outputs are shown **(Mitra, 2001)**.

In general, signals can be classified into two types. It depends on the nature of the independent variables and value of the function defining the signal. If independent variable is continuous then signal is called continuous-time signal. On the other hand, if the independent variable is discrete, then signal is called the discrete-time signal. A continuous-time signal is defined at every instant of time, however, a discrete-time signal is defined at discrete instants of time. Figure 2.10 illustrates continuous-time and discrete-time signals.

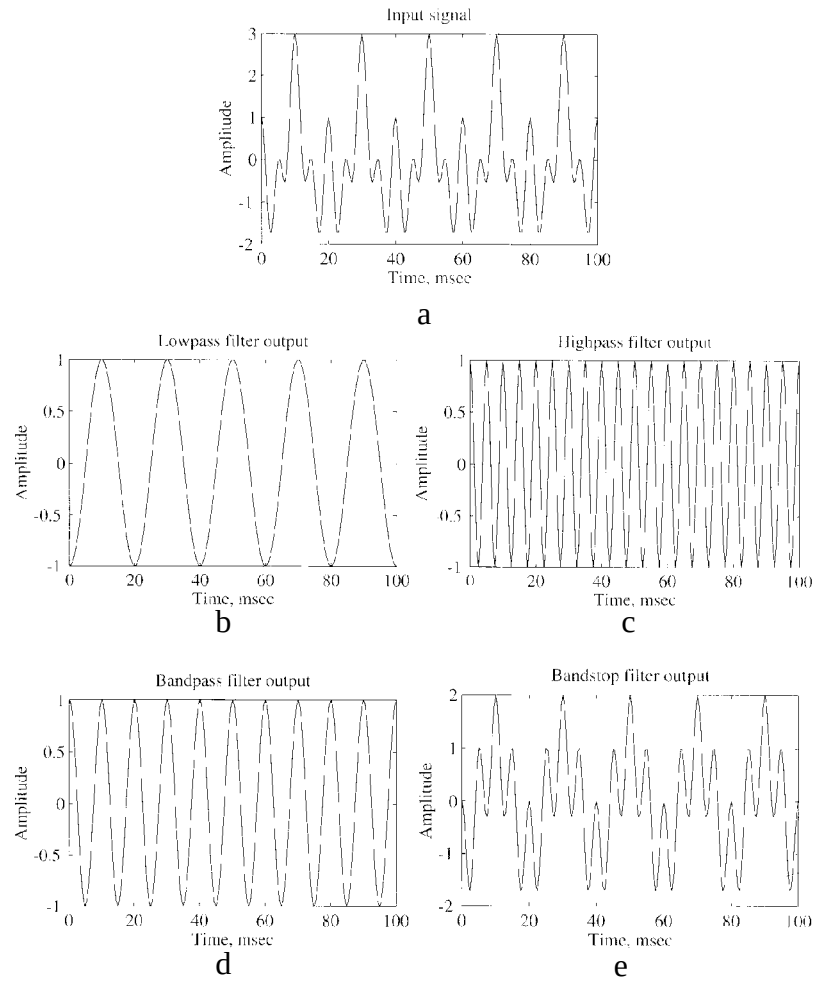


Figure 2.9: (a) Input signal, (b) output of a low-pass filter with a cut-off at 80 Hz, (c) output of a high-pass filter with a cut-off at 150 Hz, (d) output of a band-pass filter with cut-offs at 80 Hz and 150 Hz, (e) output of a band-stop filter with cut-offs at 80 Hz and 150 Hz. (This figure is taken from **Mitra, 2001**).

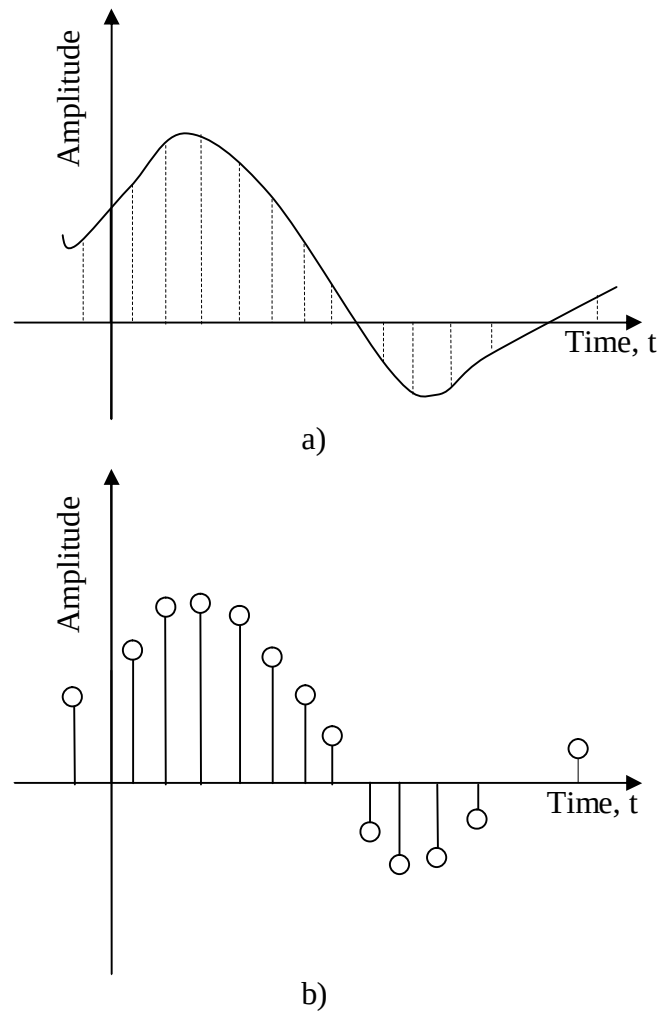


Figure 2.10: (a) A continuous-time signal, (b) a discrete-time signal.

In signal processing, linear, time-invariant system theory investigates the response of a linear, time-invariant (LTI) system to an arbitrary input signal. Though the standard independent variable is time, it could just as easily be space or some other coordinates. Linearity and time invariance are defining properties of a linear time-invariant system. **(Mitra, 2001)**

Linearity means that the relationship between the input and the output of the system satisfies the scaling and superposition properties. The linearity makes it easy to compute the response to a complicated sequence that can be decomposed into weighted combination of some simple sequences, such as the unit sample sequences or the complex exponential sequences. In this case, the desired output is given by a similarly weighted combination of the outputs to the simple sequences **(Mitra, 2001)**.

The second condition; time invariance means that whether we apply an input to the system now or Δt seconds later from now, the output will be identical, except for a time delay of Δt seconds. More specifically, an input affected by a time delay should effect a corresponding time delay in the output, hence time-invariant. The time-invariance property ensures that for a specified input, the output of the system is independent of time the input is being applied.

An LTI system can be characterized by impulse response function of the system. The impulse response of a system is its output when presented with a very brief signal, an impulse. The output of the system is simply the convolution of the input to the system with the system's impulse response. This method of analysis is often called the time domain point-of-view. The same result is true of discrete-time linear shift-invariant systems, in which signals are discrete-time samples, and convolution is defined on sequences. A schematic representation of an LTI system is given in Figure 2.11.

On the other hand, it is possible to characterize an LTI system in the frequency domain by the transfer function of the system, which is the z-transform or discrete-time Fourier transform of the system's impulse response. A transfer function is a mathematical representation of the relation between the input and output of a system. As a result of the properties of these transforms, the output of the system in the frequency domain is the product of the transfer function and the transform of the input. In other words, convolution in the time domain is equivalent to multiplication in the frequency domain (**Mitra, 2001**).

For all LTI systems, the eigenfunctions, and the basis functions of the transforms, are complex exponentials. This is, if the input to a system is the complex waveform αe^{st} for some complex amplitude α and complex frequency s , the output will be some complex constant times the input, i.e., βe^{st} for some new complex amplitude β . The ratio β / α is the transfer function at frequency s .

Many important systems can be described via LTI theory. Especially, it is a common way to model electrical circuits with resistors, inductors and capacitors, also for spring-mass-damper systems. Any system that can be modelled as a linear

homogeneous differential equation with constant coefficients is an LTI system (Mitra, 2001).

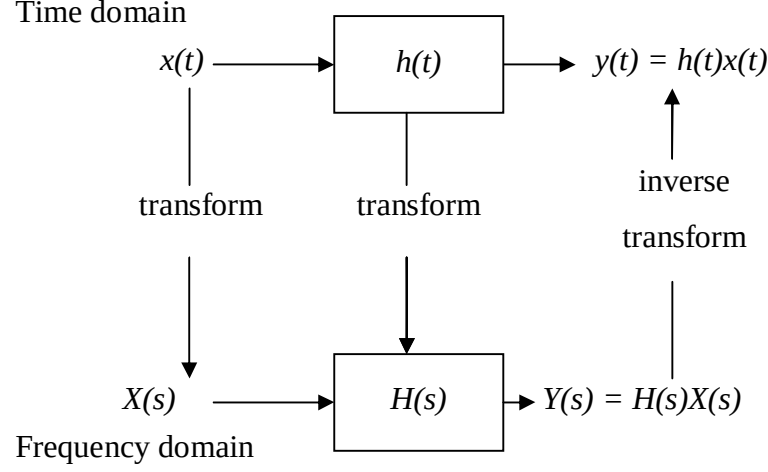


Figure 2.11: Schematic representation of an LTI system.

2.3.1. The z-Transform

A frequency-domain representation of discrete-time signals and LTI systems can be provided by the discrete-time Fourier transform. The discrete-time Fourier transform of a sequence may not exist because of convergence condition. In these cases it is not possible to make use of such frequency-domain characterization. A generalization of the discrete-time Fourier transform leads to the z-transform, which may exist for many sequences for which the discrete-time Fourier transform does not exist. Moreover, the use of z-transform techniques permits simple algebraic manipulations. Consequently, z-transform has become an important tool in the analysis and design of digital filters (Mitra, 2001).

z-transform $G(z)$ of a given sequence $g[n]$ is defined as in follows,

$$G(z) = Z\{g[n]\} = \sum_{n=-\infty}^{\infty} g[n]z^{-n} \quad (2.29)$$

where $z = \text{Re}(z) + j\text{Im}(z)$ is a complex variable. If we let $z = re^{j\omega}$, then,

$$G(re^{j\omega}) = \sum_{n=-\infty}^{\infty} g[n]r^{-n}e^{-j\omega n} \quad (2.30)$$

It can be interpreted as the discrete-time Fourier transform of the modified sequence $g[n]r^{-n}$. If $r = 1$, it means $|z| = 1$, the z-transform of $g[n]$ reduces to its discrete-time Fourier transform.

2.3.2. Rational z-Transforms

In the case of LTI discrete-time systems all pertinent z-transforms are rational functions of z^{-1} , are ratios of two polynomials in z^{-1} ,

$$G(z) = \frac{P(z)}{D(z)} = \frac{p_0 + p_1z^{-1} + \dots + p_{M-1}z^{-(M-1)} + p_Mz^{-M}}{d_0 + d_1z^{-1} + \dots + d_{N-1}z^{-(N-1)} + p_Nz^{-N}} \quad (2.31)$$

where the degree of the numerator polynomial $P(z)$ is M and that of denominator polynomial $D(z)$ is N . The z-transform $G(z)$ for $z = re^{j\omega}$ is merely the Fourier transform of the modified sequence $g[n]r^{-n}$. Accordingly, by the inverse Fourier transform relation,

$$x[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(e^{j\omega})e^{j\omega n}d\omega \quad (2.32)$$

we have,

$$g[n]r^{-n} = \frac{1}{2\pi} \int_{-\pi}^{\pi} G(e^{j\omega})e^{j\omega n}d\omega. \quad (2.33)$$

The input-output relation of an LTI discrete-time system with an impulse response is given by the convolution sum,

$$y[n] = \sum_{k=-\infty}^{\infty} h(k)x(n-k) \quad (2.34)$$

If the input $x[n]$ is a complex exponential sequence,

$$x[n] = e^{j\omega n} \quad (2.35)$$

Then the output,

$$y[n] = \sum_{k=-\infty}^{\infty} h[k] e^{j\omega(n-k)} = \left(\sum_{k=-\infty}^{\infty} h[k] e^{-j\omega k} \right) e^{j\omega n} \quad (2.36)$$

$$y[n] = H(e^{j\omega}) e^{j\omega n} \quad (2.37)$$

$$H(e^{j\omega}) = \sum_{k=-\infty}^{\infty} h[k] e^{-j\omega k} \quad (2.38)$$

Here, the quantity $H(e^{j\omega})$ is called the frequency response of the LTI discrete-time system and it provides a frequency-domain description of the system. $H(e^{j\omega})$ is also a complex function of ω with period 2π and can be expressed in terms of real and imaginary parts or its magnitude and phase.

$$H(e^{j\omega}) = H_{re}(e^{j\omega}) + jH_{im}(e^{j\omega}) = |H(e^{j\omega})| e^{j\theta(\omega)} \quad (2.39)$$

where, $H_{re}(e^{j\omega})$ and $H_{im}(e^{j\omega})$ are, the real and imaginary parts of $H(e^{j\omega})$, respectively, and,

$$\theta(\omega) = \arg\{H(e^{j\omega})\} \quad (2.40)$$

$|H(e^{j\omega})|$ is called the magnitude response and $\theta(\omega)$ is called the phase response of the LTI discrete-time system (**Mitra, 2001**).

2.3.3. Concept of Filtering

An application of an LTI discrete-time system passes certain frequency components in a signal through the system without any distortion and blocks other frequency

components. Such systems are called digital filters. The key to the filtering process is the inverse discrete-time Fourier transform which expresses an arbitrary input sequence as a linear weighted sum of an infinite number of exponential sequences.

The nonzero phase response of the system causes some delay on the output signal $y[n]$ of a frequency-selective LTI discrete-time system relative to the input signal $x[n]$.

2.3.4. The Transfer Function

As stated above, the frequency response function $H(e^{j\omega})$ of a system gives important information about characteristics of a system. A generalization of the frequency response function leads to the concept of transfer function. An LTI digital discrete-time system with input and output sequences $x[n]$ and $y[n]$ respectively, and impulse response $h[n]$ has been given as follows,

$$y[n] = \sum_{k=-\infty}^{\infty} h[k]x[n-k] \quad (2.41)$$

$Y(z)$, $X(z)$ and $H(z)$ denote the z-transform of $y[n]$, $x[n]$ and $h[n]$, respectively. Taking the z-transform of both sides of Eq. (2.41) we obtain input-output relation of the filter in the z-domain as follows,

$$Y(z) = H(z)X(z) \quad (2.42)$$

The z-transform of the impulse response sequence $h[n]$ of the filter $H(z)$, is called the transfer function or the system function, **(Mitra, 2001)**.

$$H(z) = \frac{Y(z)}{X(z)} \quad (2.43)$$

2.3.5. Types of Transfer Functions

The time-domain classification of a digital transfer function based on the length of its impulse response sequence leads to the finite impulse response (FIR) and infinite

impulse response (IIR) transfer functions. In the case of digital transfer functions with frequency-selective frequency responses, one classification is based on the shape of the magnitude function or the form of the phase function. So, four types of ideal filters can be defined, ideal low-pass, ideal high-pass, ideal band-pass and ideal band-stop filters. A digital filter which is designed to pass signal components of certain frequencies should have a frequency response of value equal to one at these frequencies and should have a frequency response value equal to zero to all other frequencies to block signal components those frequencies. Where the frequency response is equal to one that region is called pass band, and the region where the frequency response is equal to zero is called stop band of filter. In Figure 2.13 the frequency responses of four types of ideal digital filters are shown.

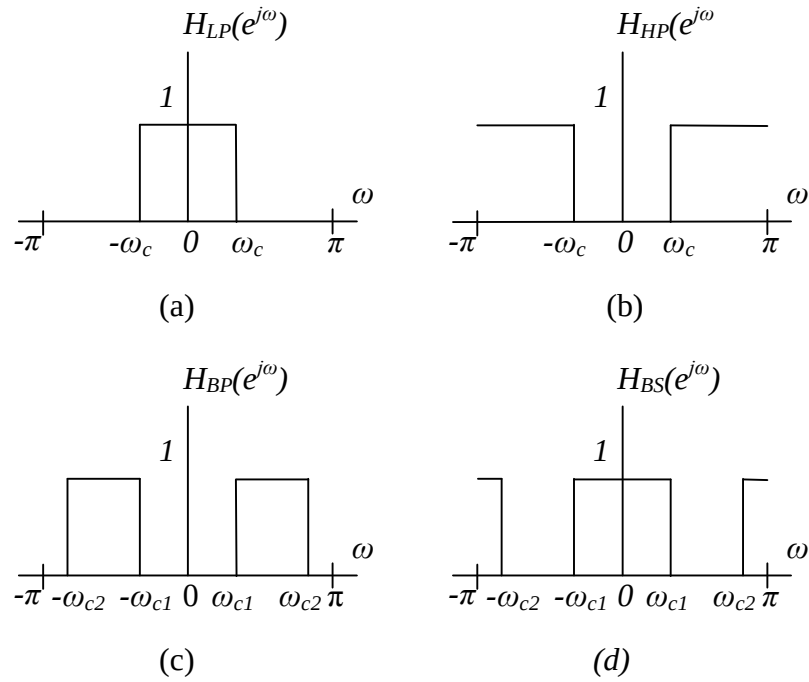


Figure 2.12: The frequency responses of four types of ideal digital filters, (a) low-pass filter, (b) high-pass filter, (c) band-pass filter, (d) band-stop filter (Mitra, 2001).

In many applications it is important to ensure that digital filter does not distort to phase of the input signal components. The way to avoid any phase distortions is to make the frequency response of the filter real and nonnegative, i.e., design the filter with a zero phase characteristic. However, sometimes it is not possible to design a filter with a zero phase. Another way to avoid phase distortion is processing the input data in both the forward and reverse directions. After filtering in the forward

direction, reverse the filtered sequence and run it back through the filter. The resulting sequence has precisely zero-phase distortion and double the filter order.

2.3.6. Finite Impulse Response (FIR) and Infinite Impulse Response (IIR) Filters

Traditionally, digital filters have been classified into two large families; first those whose transfer function does not have a denominator, and second those whose transfer function has a denominator. Since the filters of the first family admit a realization where the output is a linear combination of a finite number of input samples, they are sometimes called non-recursive filters. For these systems, it is more customary and correct to refer to the impulse response, which has a finite number of non-null samples, thus calling them Finite Impulse Response (FIR) filters. On the other hand, the filters of the second family admit only recursive realizations, thus meaning that the output signal is always computed by using previous samples of itself. The impulse response of these filters is infinitely long, thus justifying their name as Infinite Impulse Response (IIR) filters.

An FIR filter is a linear combination of a finite number of samples of the input signal. FIR filters are also known as non-recursive filters. Finite impulse response (FIR) filters get their name from the following property, an impulse signal fed into a FIR filter results in an output which decays to zero after a finite number of iterations. The Eq.(2.44) describes how the output y of a finite impulse response filter is calculated from the input x . This equation simply says that the n^{th} output is a weighted average of the most recent N inputs. The mathematical expression of a FIR filter is,

$$y[n] = \sum_{m=0}^N h[m]x[n-m] \quad (2.44)$$

Since the time extension of the impulse response is $N+1$ samples, we say that the FIR filter has length $N+1$. The transfer function is obtained as the z-transform of the impulse response and it is a polynomial in the powers of z^{-1} ,

$$H(z) = \sum_{m=0}^N h_m z^{-m} = h_0 + h_1 z^{-1} + \dots + h_N z^{-N} \quad (2.45)$$

Since such polynomial has order N , we also say that the FIR filter has order N . The IIR filter is the same as the FIR filter, only with an additional summation term which feeds back previous outputs. This kind of filters are also known as recursive filters. These filters can produce superior results with much less computational cost, but they are harder to design, and can suffer from stability problems if improperly designed. When an IIR filter is said to be unstable, it means that the feedback is too intense, and the output of the filter is out of control, producing either wild oscillations or outputs which diverge to infinity exponentially. In software, this can cause a crash due to floating point overflow. IIR filter is represented by a difference equation where the output signal at a given instant is obtained as a linear combination of samples of the input and output signals at previous time instants. Moreover, an instantaneous dependency of the output on the input is also usually included in the IIR filter. The difference equation that represents an IIR filter is,

$$y[n] = \sum_{m=1}^N a_m y[n-m] + \sum_{m=0}^M b_m x[n-m] \quad (2.46)$$

The array a holds weighting coefficients for feeding back the previous N outputs into the current output value. Note that a FIR filter is really just an IIR filter with $N = 0$. While the impulse response of FIR filters has a finite time extension, the impulse response of IIR filters has, in general, an infinite extension. The transfer function is obtained by application of the z -transform to the above equation. The result is the rational function $H(z)$ that relates the z -transform of the output to the z -transform of the input,

$$H(z) = \frac{b_0 + b_1 z^{-1} + \dots + b_M z^{-M}}{1 + a_1 z^{-1} + \dots + a_N z^{-N}} \quad (2.47)$$

The filter order is defined as the degree of the polynomial in z^{-1} that is the denominator of the above equation (**Oppenheim & Schafer, 1989**).

The impulse response is infinite because there is feedback in the filter; if you put in an impulse, theoretically an infinite number of non-zero values will come out.

FIR filters do not use feedback, so for a FIR filter with N coefficients, the output always becomes zero after putting in N samples of an impulse response. IIR filters can achieve a given filtering characteristic using less memory and calculations than a similar FIR filter.

When we compare FIR filters with IIR, FIR filters have following advantages, they can easily be designed to be linear phase. Put simply, linear-phase filters delay the input signal, but do not distort its phase. They are simple to implement. They are suited to multi-rate applications. Multi-rate means either decimation (reducing the sampling rate), interpolation (increasing the sampling rate), or both. Whether decimating or interpolating, the use of FIR filters allows some of the calculations to be omitted, thus providing an important computational efficiency. In contrast, if IIR filters are used, each output must be individually calculated, even if it that output will discarded so the feedback will be incorporated into the filter. They have desirable numeric properties. In practice, all digital filters must be implemented using finite-precision arithmetic, that is, a limited number of bits. The use of finite-precision arithmetic in IIR filters can cause significant problems due to the use of feedback, but FIR filters have no feedback, so they can usually be implemented using fewer bits, and the designer has fewer practical problems to solve related to non-ideal arithmetic.

Compared to IIR filters, FIR filters sometimes have the disadvantage that they require more memory and calculation to achieve a given filter response characteristic. Also, certain responses are not practical to implement with FIR filters.

In this thesis the digital filters to smooth noisy data have been designed by Matlab routines. These Matlab routines are ready to use and built on digital filtering concept of digital signal processing subject, and they give any desired digital filter for certain cut-off frequency, filter order, for any type of digital filters like low-pass, high-pass, band-pass and band-stop filters. A filter can be built for finite impulse response or infinite impulse response transfer function, also the discrete-time Fourier transform

or z-transform can be used to design a digital filter by Matlab routines. Once filter designed its characteristics can be analyzed with graphs.

2.4. Particle Image Velocimetry (PIV-Measurements)

In the last two decades a multi-point technique, called Particle Image Velocimetry (PIV), was developed that is one of the most powerful tools for the investigation of complex flow fields today. The Particle Image Velocimetry is a non-intrusive technique for measuring the spatial distribution of the velocity within a plane inside the flow, indirectly via the displacement of moving particle groups within a certain time (**Raffel, et al., 1998**). For this purpose the flow is homogeneously seeded with appropriate tracer particles whereby the concentration of the particles must be well adjusted with regard to the finest flow structures. In addition, the deviation of the particle velocity u_p from the real flow motion u must be negligible compared to the uncertainty of the imaging and recording system and to the uncertainty due to the evaluation procedure. After the required seeding concentration has been obtained, a desired plane inside the flow is illuminated twice by a thin laser light-sheet. The light scattered by the tracer particles at time t and t' in the direction of the recording optics is stored on individual frames of a single frame transfer CCD (charge coupled device) camera, whose optical axis is perpendicular to the light-sheet. It is obvious that the particle displacement Δx must be small relative to the finest flow scales, as only phenomena that occur over a time interval which is longer than $\Delta t = t - t'$ and that have a spatial extent larger than the absolute displacement Δx can be resolved, but the particle image displacement Δx , on the other hand, must be large for accurate measurements. The local displacement of the double exposed particles is finally determined from the two single exposed recordings by means of spatial cross-correlation techniques and afterwards divided by Δt and the magnification factor M of the imaging system to calculate the first order approximation of the velocity field (**Kahler, 2003**). A sketch of a PIV system has been given in Figure 2.14.

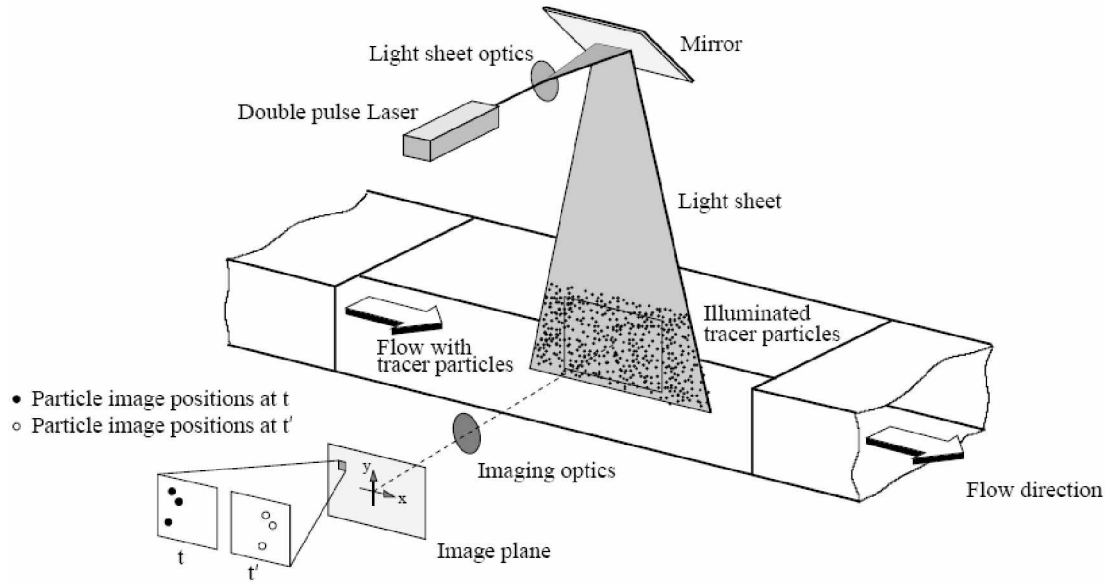


Figure 2.13: A sketch of a PIV system.

In this thesis smoothing and resolution enhancement processes implemented for a data set from scanning PIV measurements of a laminar separation bubble on an airfoil (SD7003). These measurements have been performed at the Institute of Aerodynamics, RWTH Aachen University, Aachen, Germany (**Burgmann et al., 2006**). Scanning PIV is applied to the laminar-separation transition zone in a laminar separation bubble to investigate the spanwise structure and dynamics of the roll-up of vortices within the bubble. The laminar flow separation with turbulent reattachment is studied on the suction side of an airfoil SD7003 (chord length $c = 20$ cm) at Reynolds number 20000 in a Göttingen type water tunnel which has a 30cm x 25cm cross section. A photograph of airfoil can be seen in Figure 2.15. The airfoil with a transparent segment in the spanwise center was mounted between the upper and lower wall of the tunnel. In Figure 2.16 a sketch of experimental set up and a photograph of the experimental system are shown. To characterize the spatial and temporal evolution in the separation region, the standard time-resolved PIV method was applied in the selected region of the separation bubble. The flow is recorded with a CMOS high-speed camera at frame rates between 308-922 Hz and a resolution of 1024 x 512 pixels. The light-sheet illumination system consisted of ten adjustable laser diodes each of which had a power of 50 W in continuous mode at a wavelength of 805 nm. All diodes could be adjustable and positioned independently. A special electronic control allowed to pulse the laser diodes in any desired sequence or simultaneously. The frame capture signal was synchronized with the

pulse control such that each image in the sequence corresponded to a pulse of a certain laser diode. In Figure 2.17 a photograph of laser sheet scanning system has been presented. Experiments were performed for a free stream velocity of $u_\infty = 0.1$ m/s and an angle of attack of $\alpha = 4^\circ$.



Figure 2.14: A photograph of airfoil SD7003.

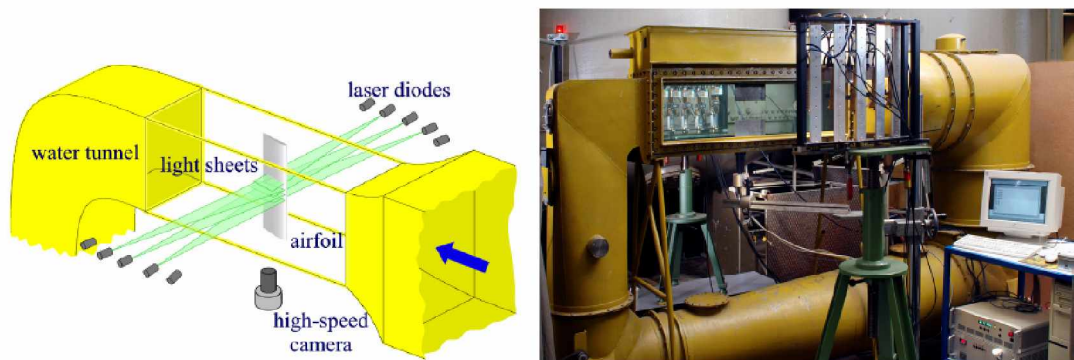


Figure 2.15: A sketch of experimental set up and a photograph of the whole system.

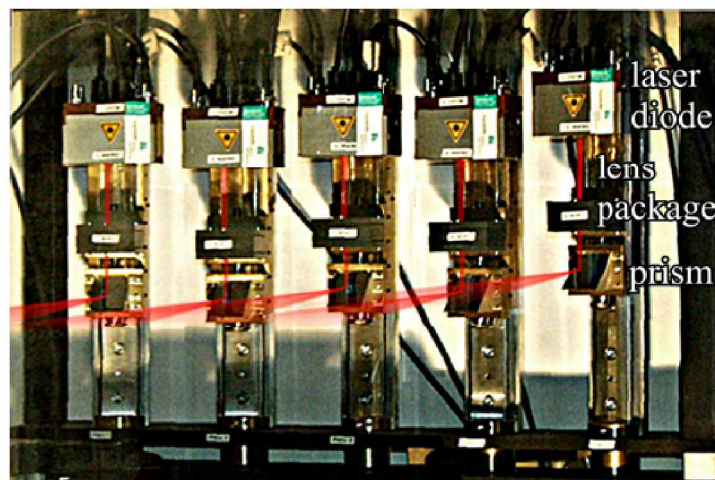


Figure 2.16: A photograph of laser sheet scanning system.

The airfoil PIV data consist of only 5 planes in the cross-flow velocity direction (z), and each plane has 49 grid points in the streamwise direction (x), and 104 grid points in the spanwise direction (y) (**Burgmann et al., 2006**). The available phase-averaged snapshot number is 11. In other words, there are 49 constant x -planes, 104 constant y -planes and only 5 constant z -planes. Constant z -planes are 0.75 mm, 1.5 mm, 2.25 mm, 3.0 mm and 3.75 mm away from airfoil surface. In Figure 2.18 and Figure 2.19(a-c), a general 3D view with constant z -planes and 2D constant x , y and z -planes are shown respectively. Because of the no-slip condition on the airfoil surface, an additional zero-velocity plane is assumed as the airfoil can be considered parallel to the measurement planes in the region of interest.

In Figure 2.18 the streamwise velocity is shown on the available planes for the first snapshot. In x and y -directions, we have enough grid points for further data post processing and for visualization methods. However, we obviously need more grid points in z -direction (wall normal direction) in order to visualize the flow field faithfully and to reduce the numerical errors in data analyses procedures, such as to implement vortex identification and traction methods.

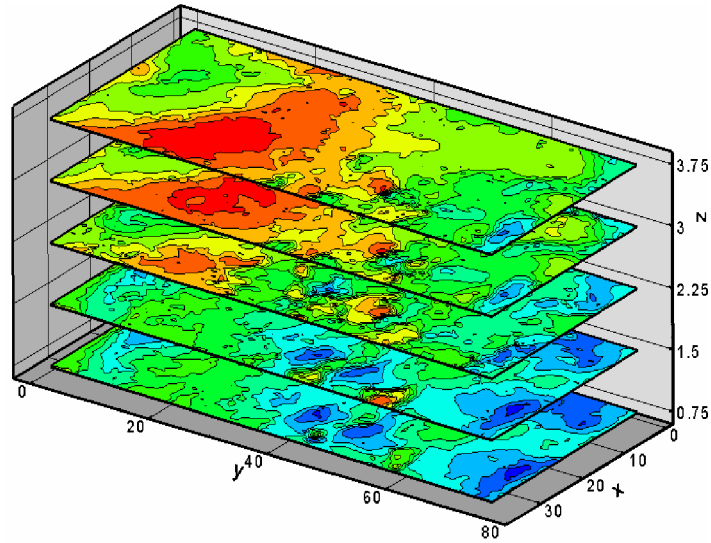


Figure 2.17: 3D view of the experimental data.

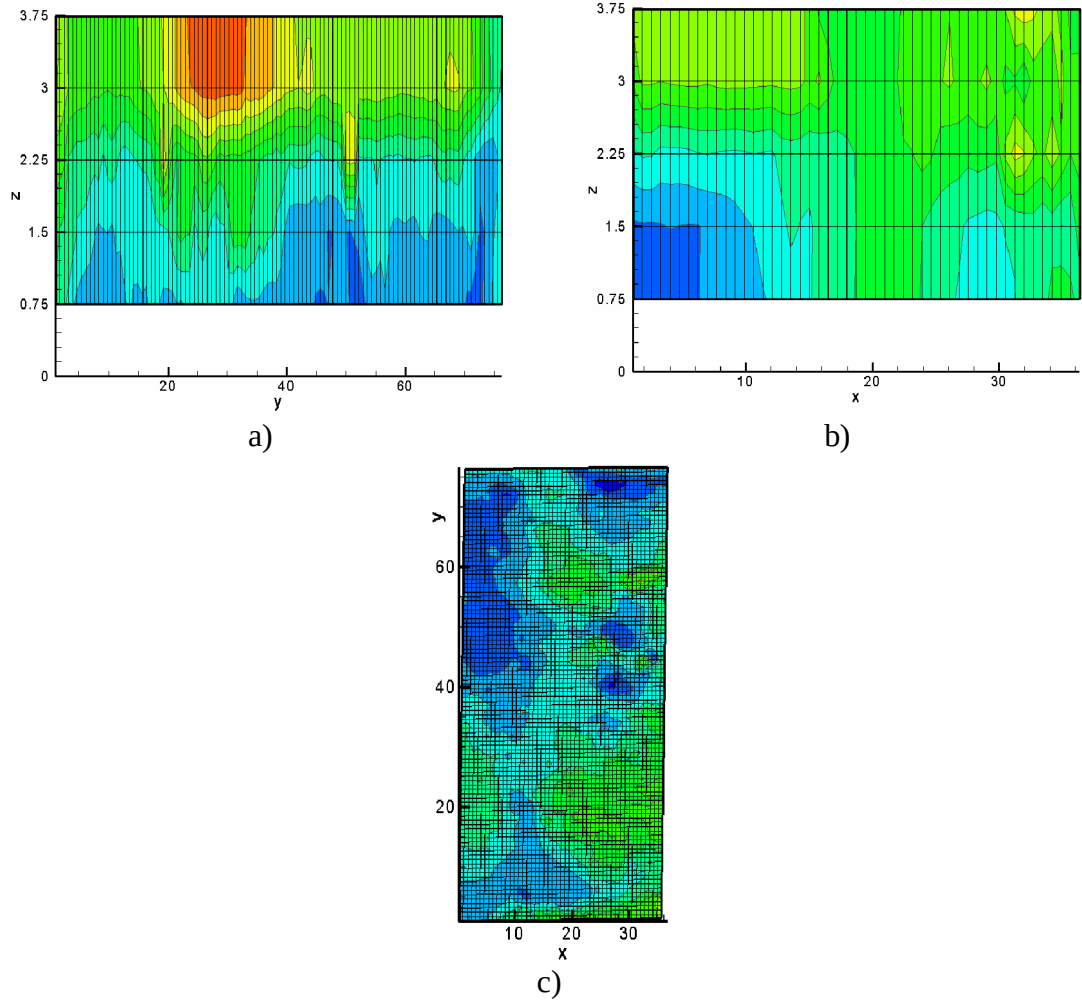


Figure 2.18: (a) 2D constant x -plane, (b) 2D constant y -plane, (c) 2D constant z -plane, (**Burgmann et al., 2006**).

3. DATA SMOOTHING

In practice, many experimental data are noisy and have to be smoothed for better investigations. The sources behind noise includes measurement errors and instabilities. Smoothing is a kind of filtering and nothing more than re-evaluation of design data set.

3.1. Data Smoothing via POD modes

As the airfoil PIV data consists of only 11 snapshots, the standard POD application includes extraction of 11 3-D POD modes for this three-dimensional flow field. With the standard POD application it is only possible to increase the temporal resolution of flow. However, here we propose a new idea based on a 2-D POD procedure using “pseudo-snapshots” in order to smooth and particularly enhance the spatial resolution of three-dimensional experimental data. The experimental data has 11 snapshots and 5 constant z-planes for each snapshot, so if we multiply snapshots with constant z-planes we come up with $S = 55$ pseudo-snapshots. Then we can construct a POD model using these 55 pseudo-snapshots. With these snapshots we can build a correlation matrix of the flow field as in follows

$$C_{i,j} = \frac{1}{S} \sum_{k=1}^K \sum_{l=1}^L \left[u(x_k, y_l, z_i, t_i) u(x_k, y_l, z_j, t_j) + v(x_k, y_l, z_i, t_i) v(x_k, y_l, z_j, t_j) \right] \quad (3.1)$$

Here, S is the number of pseudo-snapshots, u is the streamwise and v is the spanwise velocity, respectively. We have to mention that the multi-plane PIV measurements carried out by **Burgmann et al., (2006)** did not contain the wall normal component of the velocity, so it will be calculated later using the continuity equation after sufficient spatial resolution is achieved. Because the experimental data has the streamwise and the spanwise velocity, the POD model also contains these two components. Eigenfunctions of the flow field are written as follows,

$$\Phi^k = \phi_u^k(x, y)i + \phi_v^k(x, y)j \quad (3.2)$$

where,

$$\phi_u^k(x, y) = \sum_{i,j=1}^S \alpha_{k,ij} u(x, y, z_i, t_i) \quad (3.3)$$

$$\phi_v^k(x, y) = \sum_{i,j=1}^S \alpha_{k,ij} v(x, y, z_i, t_i) \quad (3.4)$$

In order to reconstruct and enhance PIV data by POD the “spatio-temporal” coefficients of the eigenmodes are calculated. In Figure 3.1 the spatio-temporal coefficients as a function of wall normal direction z are given.

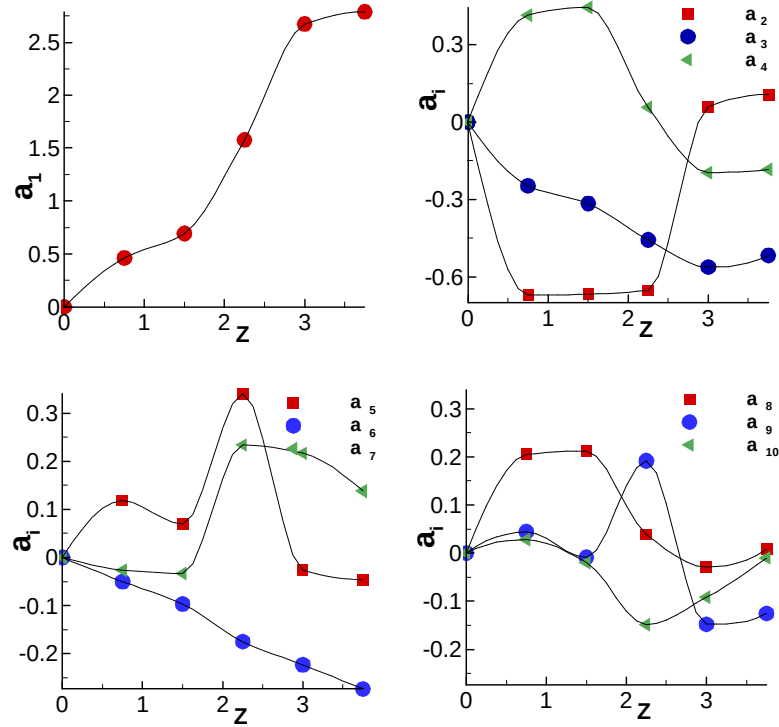


Figure 3.1: Spatio-temporal expansion coefficients, symbols denote known planes and solid lines are curve fits used for interpolation.

In Figure 3.1 symbols denote the coefficient at known z -planes. The curve fitted solid lines can be used to determine spatio-temporal coefficients to evaluate any desired missing (new) plane. In Figure 3.2 the contour plots (z versus time) of the first six spatio-temporal expansion coefficients are given.

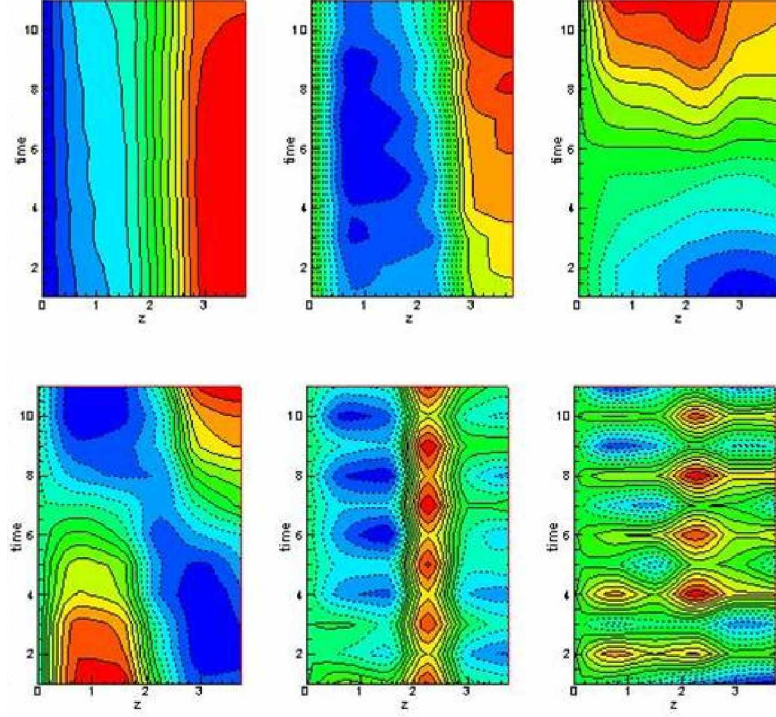


Figure 3.2: Spatio-temporal expansion coefficients $a_k(z, t)$.

Once spatio-temporal expansion coefficients are determined velocity components can be calculated as,

$$u(x, y, z, t) \cong \sum_{k=1}^N a_k(z, t) \phi_u^k(x, y) \quad (3.5)$$

$$v(x, y, z, t) \cong \sum_{k=1}^N a_k(z, t) \phi_v^k(x, y) \quad (3.6)$$

In the Eqs.(3.5)-(3.6) N denotes the number of modes taken into account for the estimation and in general N is much smaller than the total number of modes because the first few modes are enough to represent the most of the flow. The energy content of each POD mode is represented by its eigenvalue and is shown in Figure 3.3. It can be seen, for instance, the first mode takes more than 90% of total energy. In Table 3.1 cumulative energy contributions are given for selected number of modes.

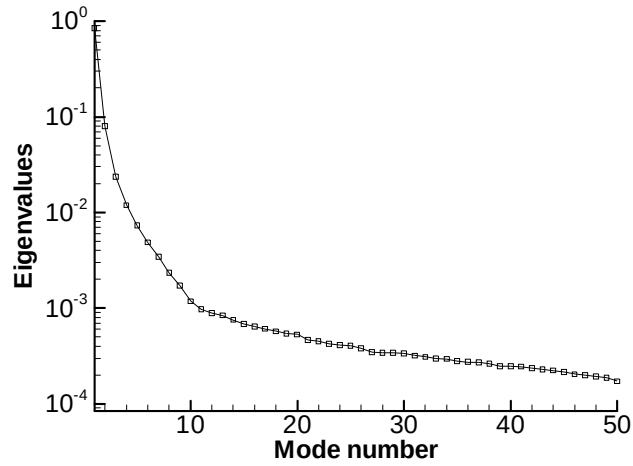


Figure 3.3: The kinetic energy content of each eigenmode.

It can be said that after the first few modes, higher modes have no significant energy content. The first two eigenmodes contain 92.68% and the first four modes contain 96.25% of total kinetic energy.

Table 3.1: Number of modes and cumulative energy contribution.

Mode number	Cumulative energy contribution (%)
M = 2	92.68
M = 4	96.25
M = 10	98.34

In Figure 3.4 the contour plots of eigenmodes $k = 2, 6, 7, 10$ and 15 are shown. As expected, as the number of modes increases, the modes become more complex and noisy. While the first few modes contain dominant features of the flow, the higher (low-energetic) modes contain detailed features and noise. Thus, with a simple idea, if we take only the first few modes in reconstruction of a data set, we can eliminate noise as well as some small features of the flow. So, POD can be considered as a natural noise filtering for experimental data. However, when we look at Figure 3.4, even the most energetic modes are somewhat noisy (though much smoother than the higher modes). Therefore, we applied Kriging to smooth all of the eigenmodes before reconstructing the flow field. Smoothing process by Kriging interpolation is given in detail in the next chapter.

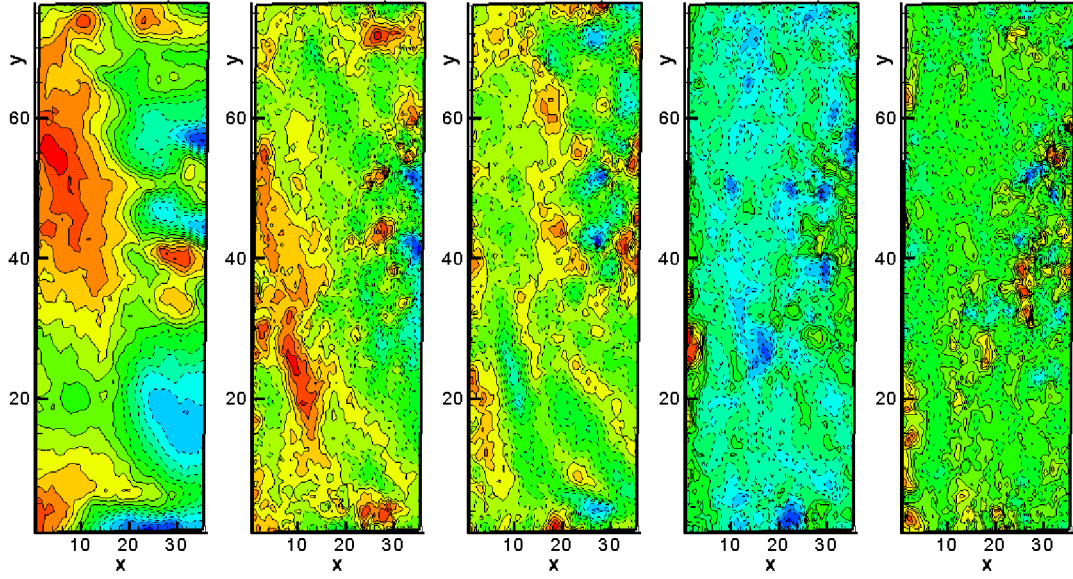


Figure 3.4: Eigenmodes (non-smoothed) of flow field, from left to right: 2nd, 6th, 7th, 10th and 15th modes, respectively.

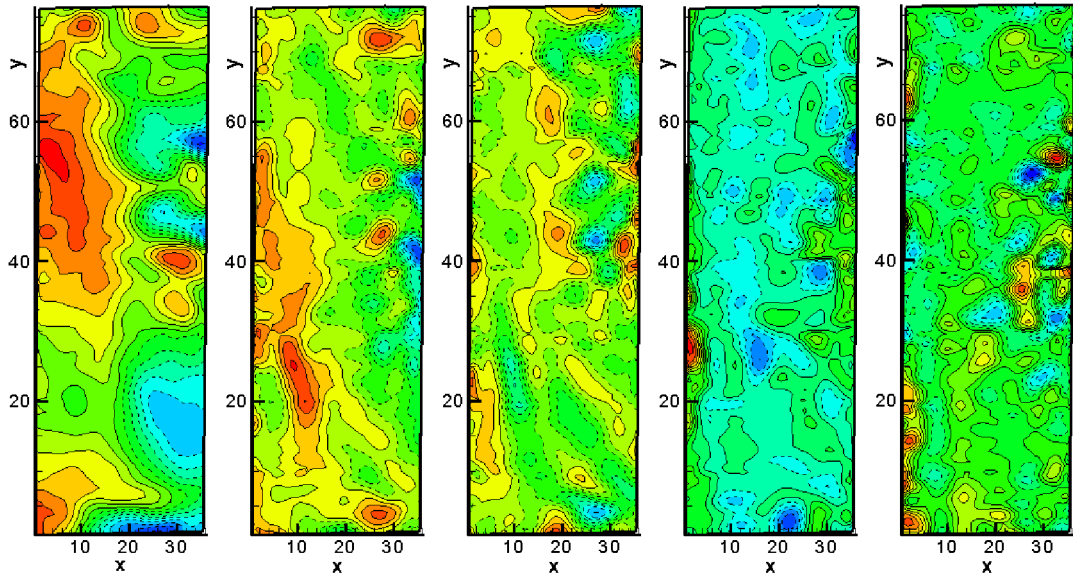


Figure 3.5: Smoothed eigenmodes of flow field, from left to right: 2nd, 6th, 7th, 10th and 15th modes, respectively.

The result is given in Figure 3.5. It is seen that the eigenmodes smoothed by Kriging lead to superior results compared to the original eigenmodes. Note also that while using all of the original POD modes recovers the original (but noisy) PIV data, all smoothed POD modes result in smoothed PIV data.

In Figure 3.6 the contour plots of the streamwise velocity ($z = 0.75$) for POD smoothing with original eigenmodes with different number of modes are given.

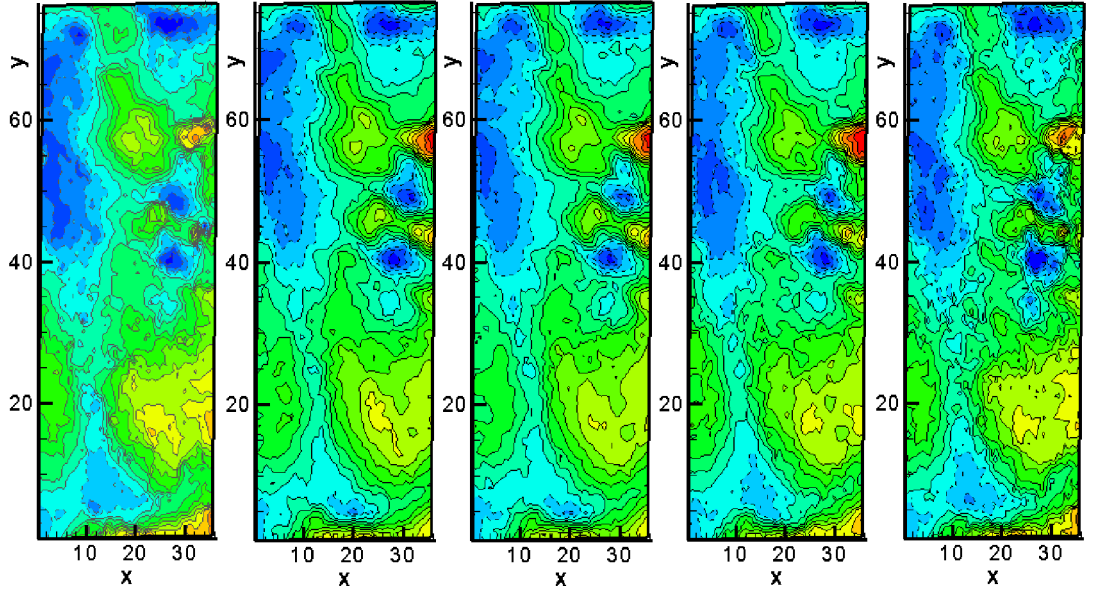


Figure 3.6: The contours of the streamwise velocity ($z = 0.75$) for POD with different numbers of modes. From left to right: PIV measurements, reconstructed with 4, 6, 10, 20 original modes.

The reconstructed data by taking the first four eigenmodes is smoother than original data, while number of modes increases reconstructed data set becomes noiser and resembles original PIV data because more kinetic energy captured. If we take all of the modes of flow field, we will reach the same PIV data.

Next, in POD reconstruction, smoothed (Kriged) eigenmodes are used instead of original eigenmodes as in Eq.(3.7). Using the kriged eigenmodes, the level of smoothing can be adjusted as desired as shown in Figure 3.7. By taking the first four kriged modes for reconstruction, we have quite smooth data, but we loose some details. On the other hand, when we take more modes, for example, even twenty kriged modes, we still have smooth data and also capture details of the flow field. Figure 3.7 can be compared with Figure 3.6 as the same PIV data are smoothed by original and kriged modes. When these two figures are compared it can be seen that kriged eigenmodes increases the level of smoothing considerably.

$$u(x, y, z, t) \cong \sum_{k=1}^N a_k(z, t) \phi_u^{k, kriged}(x, y) \quad (3.7)$$

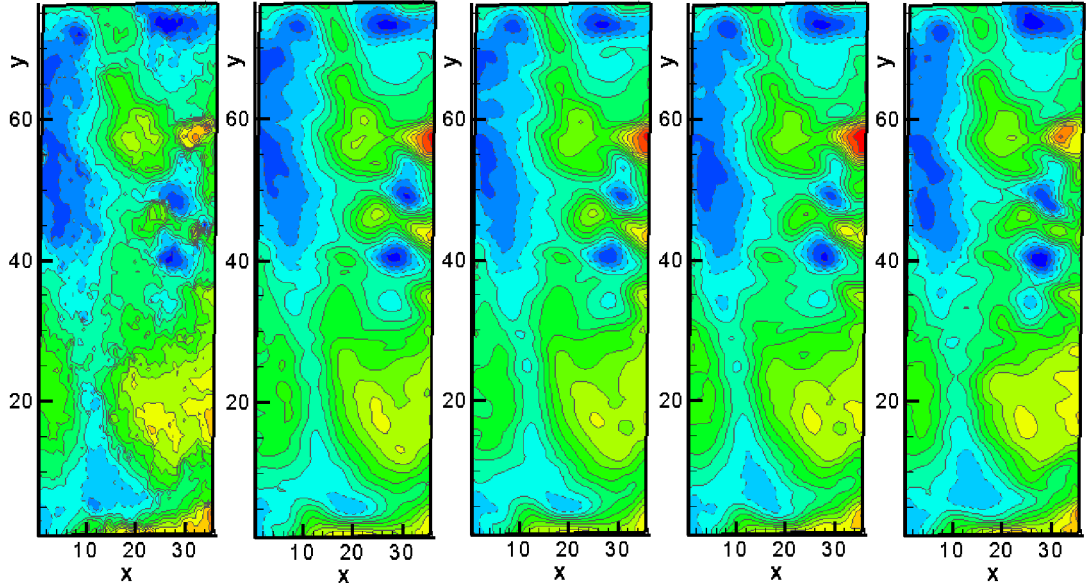


Figure 3.7: The contours of the streamwise velocity ($z = 0.75$) for POD with different numbers of modes. From left to right: PIV measurements, reconstructed with 4, 6, 10, 20 kriged modes, respectively.

3.2. Data Smoothing by Kriging Interpolation

Kriging interpolation makes a smooth data set by considering spatial dependence of all measured points. First of all, the spatial dependence or roughness of a data set has to be determined by a variogram. In this thesis, different variogram models and correlation parameters have been investigated and results are given in next chapter. The smoothing by Kriging has been achieved for the experimental data set, as well as synthetic data sets in order to have detailed and well-controlled investigations.

Different variogram models such as exponential, spherical, spline and the Gaussian models have been used to smooth the known points in the dataset. The effect of the variogram in data smoothing the function being (the streamwise velocity) is shown in Figure 3.8. It is noted that all the reported variogram models except for the Gaussian model reproduce the original PIV data (within the numerical accuracy). This means that only the Gaussian model has a capability to smooth data. In other words, the Gaussian model changes the values of the given points from experiment while the other models merely re-evaluate the same values as those that are given. We attribute this difference in slope of the variogram, as already shown and discussed in connection with Figure 2.6.

Figure 3.9 shows the streamwise velocity PIV measurements and Kriging interpolated values on a constant line $x = 33$ on $z = 0.75$ plane for various variogram models. In this figure, the red dots show PIV measurements to be smoothed and coloured lines show Kriging interpolated values by different variogram models.

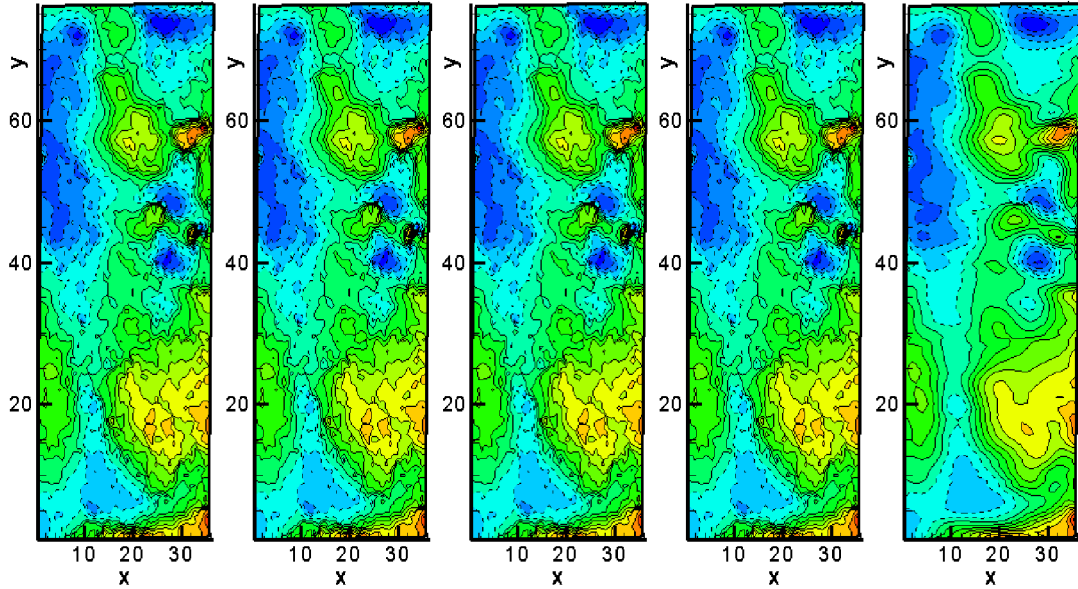


Figure 3.8: The contour plots of the streamwise velocity. From left to right: The original PIV data, smoothed PIV data by exponential model, spherical model, spline model, and the Gaussian model, respectively. (Note that all plots are for $z = 0.75$ and for the first snapshot).

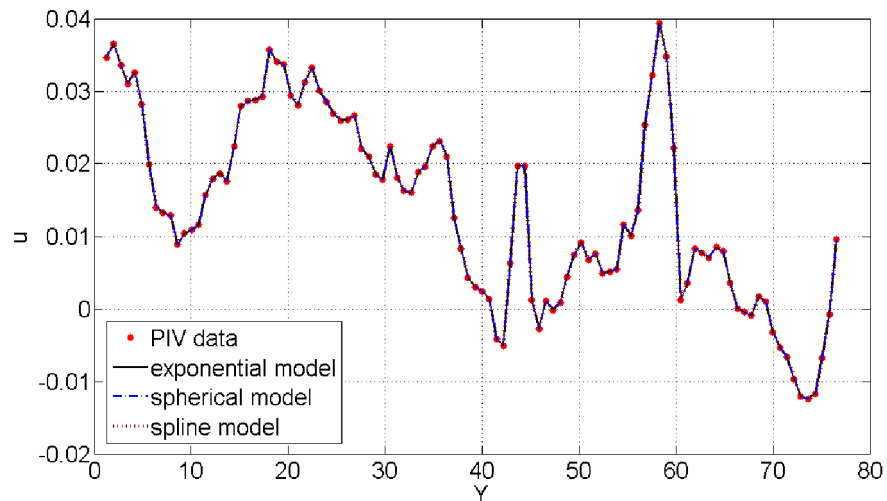


Figure 3.9: The streamwise velocity distribution on $x = 33$ ($z = 0.75$) for different variogram models.

It is apparent that the exponential, spherical and the spline models duplicates the value of measured point. That is, no change on these known points after re-evaluation. However, Figure 3.10 shows the streamwise velocity PIV measurements and smoothed values by the Gaussian model. Additionally, the smoothing results of

the Gaussian model with different correlation parameters for the first z-plane ($z = 0.75$ mm) of the first snapshot of PIV data are given in Figure 3.11 by the contour plots for the streamwise velocity.

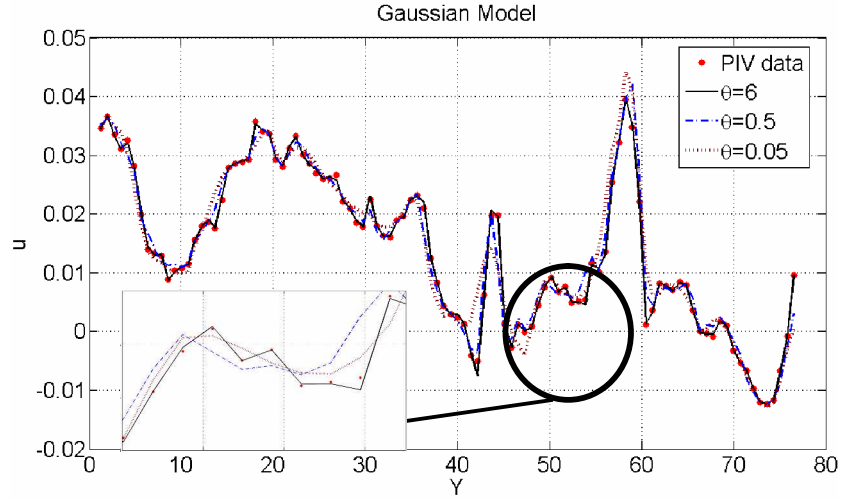


Figure 3.10: The streamwise velocity distribution on $x = 33$ ($z = 0.75$) for the the Gaussian correlation model with different correlation parameters. (See the enlarged view for selected region).

In Figure 3.10 red dots show PIV measurements while the lines show Kriging interpolation with the Gaussian model with different correlation parameters. It can be clearly seen that the correlation parameter in the Gaussian model can be used to control the level of smoothing. A high value of correlation parameter means a low smoothness and vice versa. Thus, if a high correlation parameter is used, then when estimating a point only nearby points are taken into account. In other words, for estimation process, the points will have a weight in a small range, outside of this range there is no correlation. This range can be increased by reducing the correlation parameter. The Gaussian model with correlation parameter θ , can be written as follows,

$$\gamma(h) = c_0 + c_1 \left[1 - \exp \left(-\theta \left(\frac{h}{a} \right)^2 \right) \right] \quad (3.8)$$

Similarly, other correlation parameters are also adopted for other variogram models. For other variogram models, the correlation parameter has no effect on smoothing process, however it affects the resolution enhancement process.

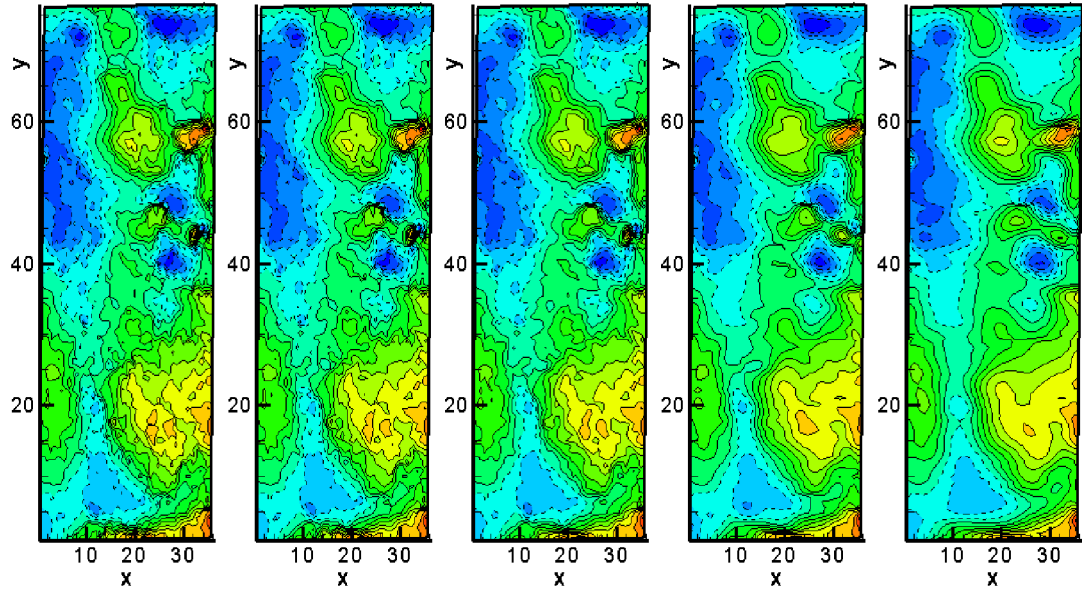


Figure 3.11: The contours of the streamwise velocity ($z = 0.75$) for the Gaussian model with different correlation parameters. From left to right: PIV measurements, the Gaussian smoothing with correlation parameters $\theta = 6, 3, 0.5, 0.1$, respectively.

It can be concluded that the Gaussian correlation model changes the values at known points when they are re-evaluated, and the correlation parameter governs the degree of this smoothing. Figure 3.12 shows the Gaussian variogram model with different correlation parameters θ .

In order to understand the specialty behind the Gaussian model, we turn back to Figure 2.6. As can be seen in Figure 2.6, the exponential and spherical models show erratic behaviour over a short distance and a linear behaviour near the origin. On the other hand, the Gaussian model describes a function that is continuous and tangential to the x-axis near the origin and shows parabolic behaviour near the origin. This distinct behaviour near the origin has a considerable effect on Kriging interpolation of the correlations to neighbouring points. In the case of actual data is not continuous and the Gaussian model is used, then negative weights are obtained and they give erratic results.

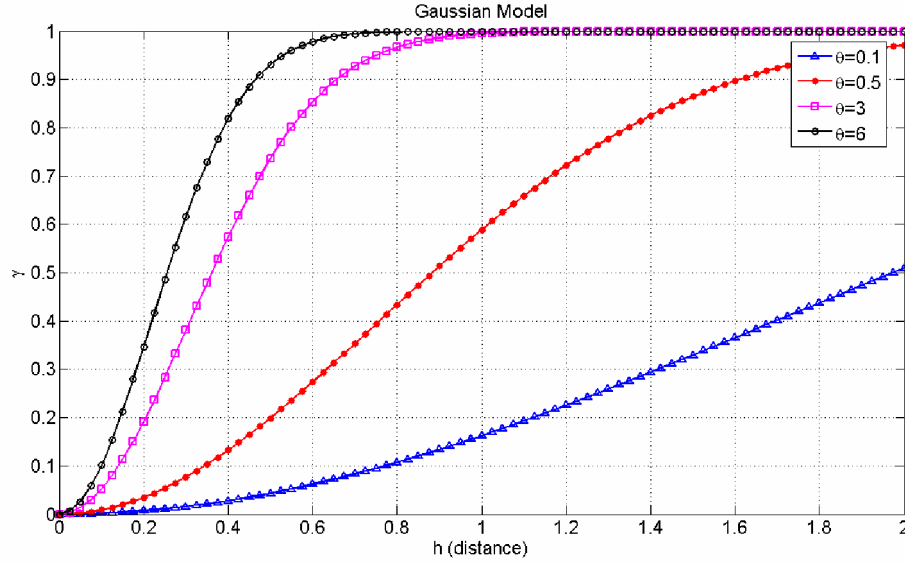


Figure 3.12: The Gaussian variogram models for different correlation parameters.

The Gaussian model creates numerical precision problems because of off-diagonal elements that have nearly zero values. This numerical problem can be removed by an addition of nugget effect. (Deutsch & Journel, 1992).

3.2.1. A comparison of Kriging and POD smoothing

Kriging and POD techniques are applied to smooth all PIV data set independently. To compare these two different methods, the contour plots of the streamwise velocity of the original PIV data and the smoothed versions of constant z -planes ($z = 0.75$ and $z = 3.75$) of the first snapshot are shown in Figure 3.13. These methods work effectively for data smoothing and give similar result to each other. The methods have their own parameters like correlation parameter, the number of modes, etc. for data reconstruction. By setting these parameters, the smoothing results can be controlled.

Although we re-evaluate all PIV data set in order to obtain a smooth data set we can not present a complete data set here. Complete, animated and compared investigations and smoothed results by Kriging interpolation and POD can be found on the following links; <http://www2.itu.edu.tr/~ceklih/research.htm>, <http://atlas.cc.itu.edu.tr/~guneshasa>, (December 2006),

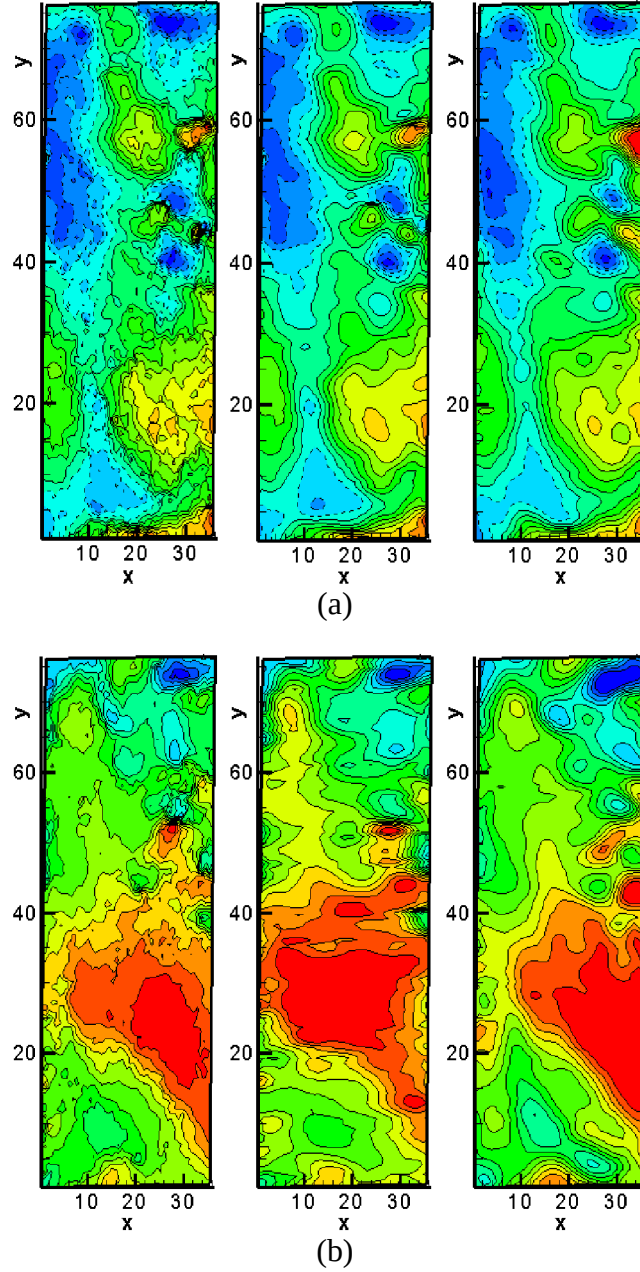


Figure 3.13: The contours of the streamwise velocity. (a) $z = 0.75$, (b) $z = 3.75$, from left to right: original PIV data, smoothed by Kriging interpolation, smoothed by POD.

3.2.2. Effect of zone number on Kriging interpolation

Performing smoothing process by Kriging interpolation requires a large computer memory when design dataset is large. In this case, we offer zonal Kriging interpolation. It means we divide the available data set into zones (parts) and smooth each zone separately by building zonal variogram and Kriging model. As the number of zones increases, the required time and memory decreases for each zone,

thus the total time and memory for smoothing process decreases noticeable. In Figure 3.14 the smoothing results of the Gaussian model with the same correlation parameter for different number of zones of the first z -plane ($z = 0.75$ mm) are given.

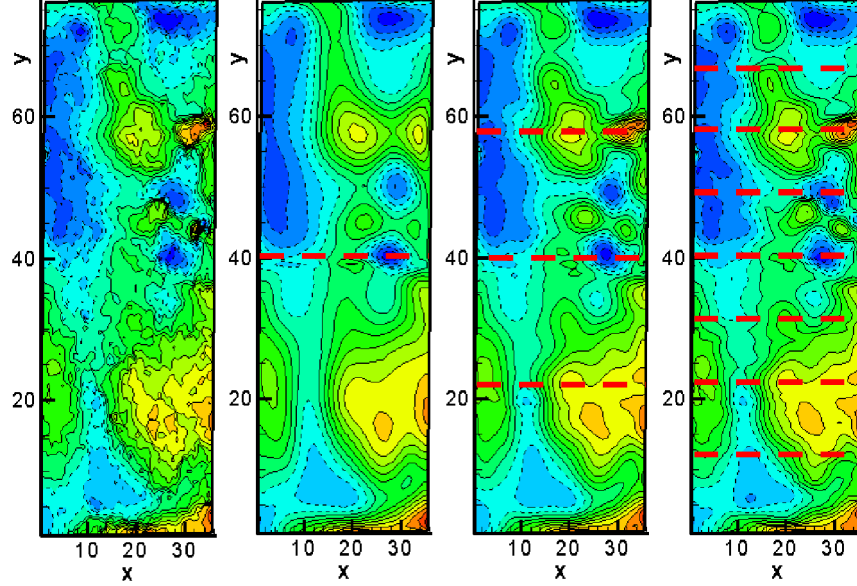


Figure 3.14: The contours of the streamwise velocity ($z = 0.75$) for the Gaussian model with different number of zones. From left to right: PIV measurements, 2 zone-smoothing, 4 zone-smoothing and 8 zone-smoothing, respectively. Note that red dashed lines show boundaries between zones.

3.2.3. 2D Line Kriging

In general, Kriging is a computationally expensive procedure and also, depending on the size of the design (known) dataset, a large memory may be required to construct and evaluate the coefficient matrix. In addition, our numerical experiments show that there is a limit for the smoothing level and after that level, data may become more noisy instead of smoother. In order to eliminate memory problems, to increase the speed of the procedure significantly, and to obtain a virtually unlimited smoothing level, we propose 2D line Kriging. This method uses a 1D spatial correlation instead of a 2D spatial correlation. 2D line Kriging re-evaluates known points on constant lines of a 2D dataset. In order to have an equivalent 2D correlation, we first take $x = \text{constant}$ grid lines (one by one) and build a Kriging model to re-evaluate known data on each x constant line separately, then we use the same procedure for the orthogonal (y) grid lines. Because this method uses correlations in both horizontal and vertical directions (consecutively) it is also

effectively a 2D method. The results of 2D line Kriging for smoothing by the Gaussian model for different correlation parameters for $z = 0.75$ are shown in Figure 3.15. When this figure is compared with Figure 3.11, it is seen that 2D line Kriging leads to considerably smoother data.

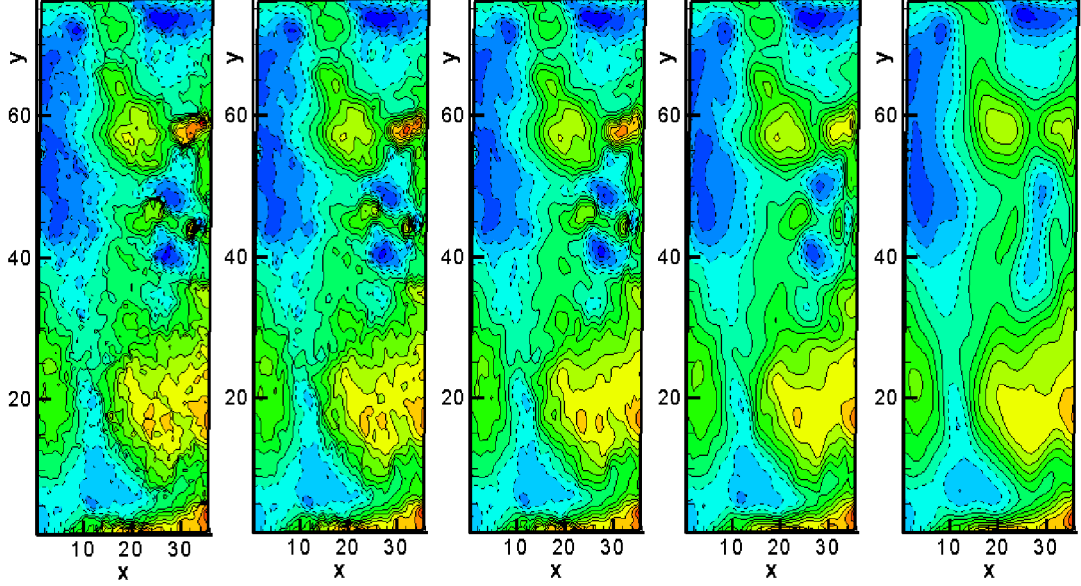


Figure 3.15: The contours of the streamwise velocity ($z = 0.75$) for 2D line Kriging by the Gaussian model with different correlation parameters. From left to right: PIV measurements, 2D line Kriging smoothing with $\theta = 10, 5, 2$ and 0.5 , respectively.

It is apparent that 2D line Kriging is suitable and can be implemented for datasets with Cartesian grids, i.e., known points should be on coordinate lines. 2D line Kriging smoothing is similar in application to digital filters, which we investigate and compare next. Figure 3.16 shows the streamwise velocity and revaluated values on a constant line $x = 30$ on $z = 0.75$ plane. In this figure red dots show PIV measurements to be smoothed and the lines show re-evaluated values by 2D line Kriging by the Gaussian model with different correlation parameters.

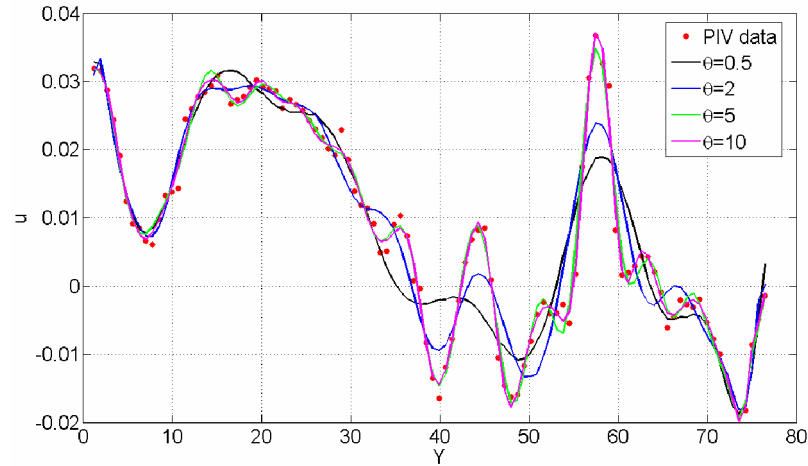


Figure 3.16: The streamwise velocity distribution on $x = 30$ line ($z = 0.75$) for 2D line Kriging by the Gaussian correlation model with different correlation parameters.

3.3. Data Smoothing via FIR/IIR Digital Filters

A digital filter passes certain frequency components in a signal through the system without any distortion and blocks other frequency components. Digital filters widely used in signal processing applications. In this thesis Matlab routines are used to design and analyze digital filters. Different types of digital filters like Butterworth, FFT-based IIR / FIR low pass filters are used for smoothing applications. Double-sided filters used to obtain zero phase application i.e., no time shifting. The properties of digital filters are discussed in Chapter 2.

2D digital filters are also applicable on coordinate lines as 2D line Kriging. In order to obtain a 2D digital filtering/smoothing, we first filter the dataset on constant lines (one line by one) in horizontal direction and next perform the same filtering method for the vertical direction. In Figure 3.17, 2D second-order Butterworth type double-sided low-pass filtering results for $z = 0.75$ plane are given for different cut-off frequencies. As shown in Figure 3.17, while the cut-off frequency decreases, the digital filtering leads to smoother data.

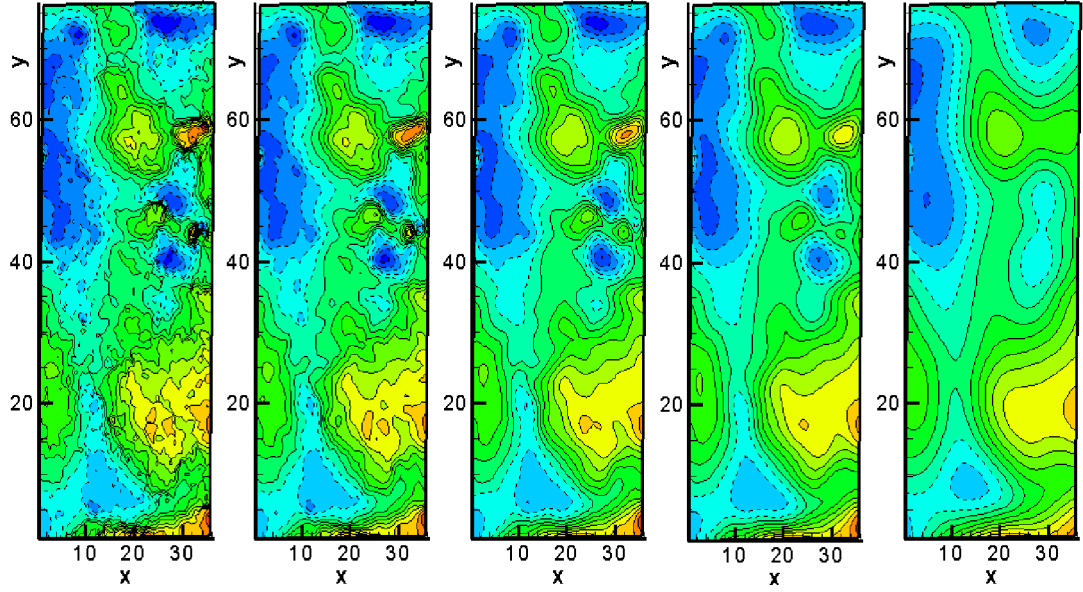


Figure 3.17: The contours of the streamwise velocity ($z = 0.75$) for 2D IIR Butterworth filter with different cut-off frequencies. From left to right: PIV measurements, smoothed data using $f_c = 0.5, 0.3, 0.2$, and 0.1 , respectively.

Figure 3.18 shows the streamwise velocity of the PIV measurements and filtered values on a constant line $x = 30$. Red dots depict PIV measurements and the lines show values filtered by the 2D low-pass IIR Butterworth filter with different cut-off frequencies.

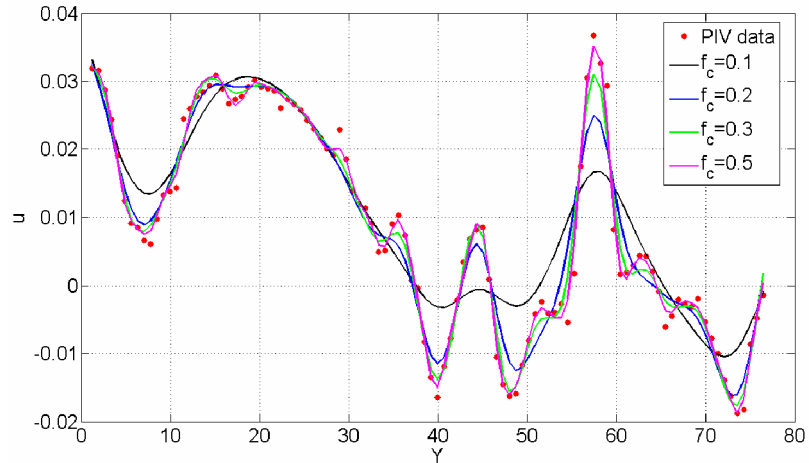


Figure 3.18: The streamwise velocity distribution on $x = 30$ ($z = 0.75$) for low-pass IIR Butterworth filter with different cut-off frequencies.

In Figure 3.19, the amplitude response of the system is given for different values of cut-off frequency. As shown in Figure 3.19, the characteristics of the Butterworth filter are that they are maximally flat in the pass band and monotonic overall.

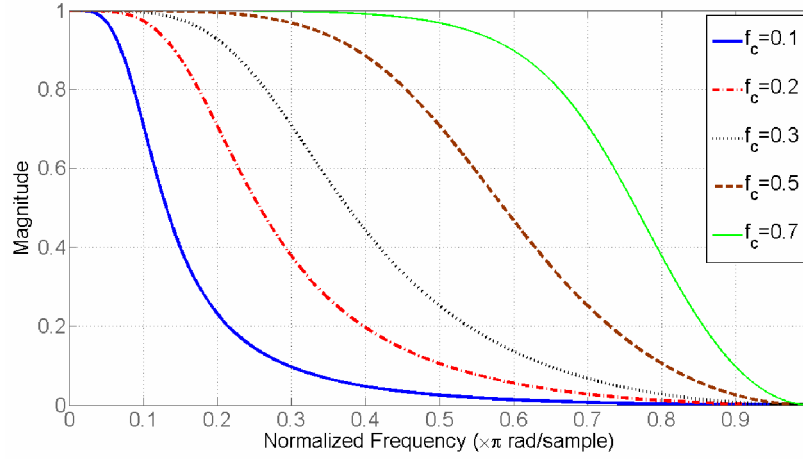


Figure 3.19: Amplitude response of the second order low-pass IIR Butterworth filter for different values of cut-off frequency.

In Figure 3.20, the amplitude response of a low-pass IIR Butterworth filter having a normalized cut-off frequency of 0.3 is given for different values of filter order. It can be said that while the order of filter increases behaviour of filter gets closer to ideal filter.

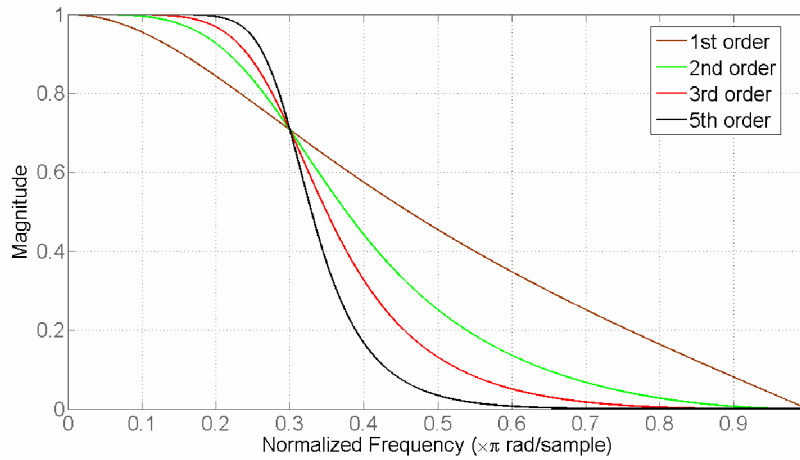


Figure 3.20: Amplitude response of low-pass IIR Butterworth filter ($f_c = 0.3$) for different values of filter order.

3.4. Smoothing of test data

When the results of ordinary Kriging interpolation, 2D line Kriging and digital filters are compared in Figures 3.11, 3.15, 3.17 respectively, it can be said that all of these methods are useful and have certain advantages/disadvantages but detailed analyses of the methods are hard to establish because the actual PIV data has no

‘known’ solution to make a comparison. Therefore, we create a synthetic (analytical) data set and disturb it by adding random noise. Then, the root-mean-square error (*rms*) of the disturbed data can be determined by taking the difference of actual data and disturbed data. We disturbed the actual solution and then we try to smooth it using aforementioned procedures. The following equation defines the *rms* error,

$$rms = \sqrt{\frac{1}{N} \sum_{i=1}^N [z_A(x, y) - z_p(x, y)]^2} \quad (3.9)$$

where, z_A is the actual value and z_p is the estimated value. The performance of the different methods can be evaluated by the *rms* values as defined above.

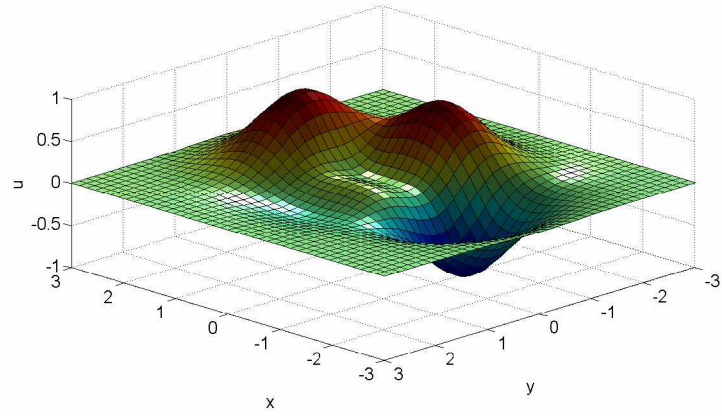
3.4.1. Creating, disturbing and smoothing test data

A 2D synthetic complex data set with several local minima and maxima is defined by Eq.(3.10) (41x41 grid points). This data set is made noisy by adding random (negative or positive) numbers to actual values at each grid point.

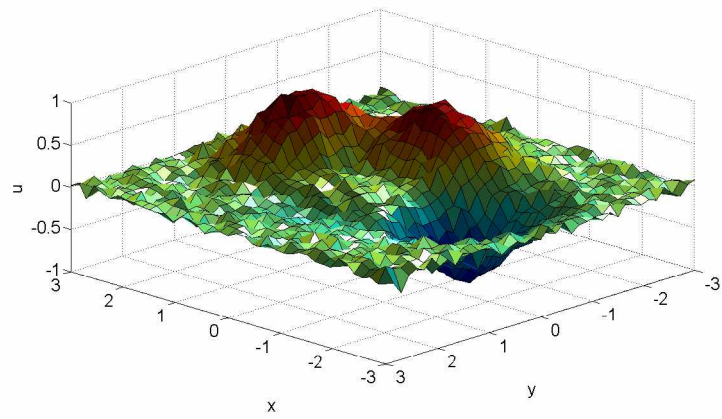
$$u(x, y) = (x^5 - y^3 + y^2) e^{-(x^2 + y^2)} \quad (3.10)$$

In Figure 3.21 the actual solution of the test data, its disturbed case and smoothing result by the Gaussian model with correlation parameter 0.5 are given. It can be seen that Kriging interpolation amazingly eliminates the noise. At some regions in the test data, especially near the boundaries, the error after smoothing is relatively higher, as the noise at this region is also relatively higher.

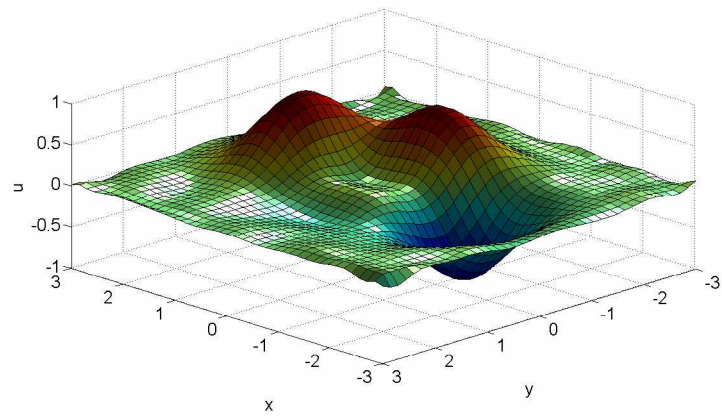
In Figure 3.22, the contour plot of $u(x, y)$ and the smoothing results are compared for the Gaussian model with different correlation parameters. The *rms* value for noisy data is 0.05 and when it is smoothed by the Gaussian model with correlation parameter $\theta = 0.1$ it reduces $rms = 0.035$, and for $\theta = 0.5$ and $\theta = 1$, $rms = 0.016$ and $rms = 0.02$, respectively.



(a)



(b)



(c)

Figure 3.21: (a) Actual data, (b) disturbed data, (c) smoothed data by the Gaussian model ($\theta = 0.5$).

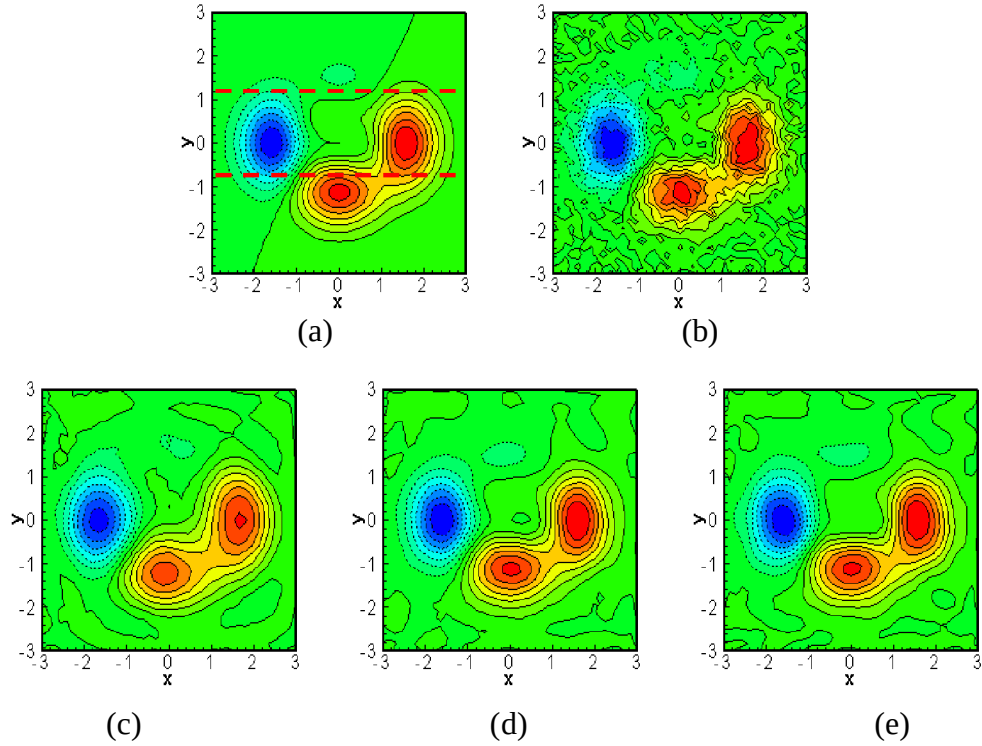


Figure 3.22: (a) Actual data, (b) disturbed data ($rms = 0.05$), (c) smoothed data by the Gaussian model ($\theta = 0.1$) ($rms = 0.035$), (d) smoothed data by the Gaussian model ($\theta = 0.5$) ($rms = 0.016$), (e) smoothed data by the Gaussian model ($\theta = 1$) ($rms = 0.02$).

As can be seen in Figure 3.22 by dashed lines, we also take constant y lines and plot actual, disturbed and smoothed data for a detailed comparison in Figure 3.23. The red dots show actual values and black diamonds show disturbed data to be smoothed. The blue line is the result obtained by Kriging interpolation with the Gaussian model ($\theta = 0.5$). When these plots are compared it can be said that the level of noise affects the smoothing result. For relatively less noisy data, the smoothing process is more successful. On the same constant lines, the results of digital filters and 2D line Kriging are given in next.

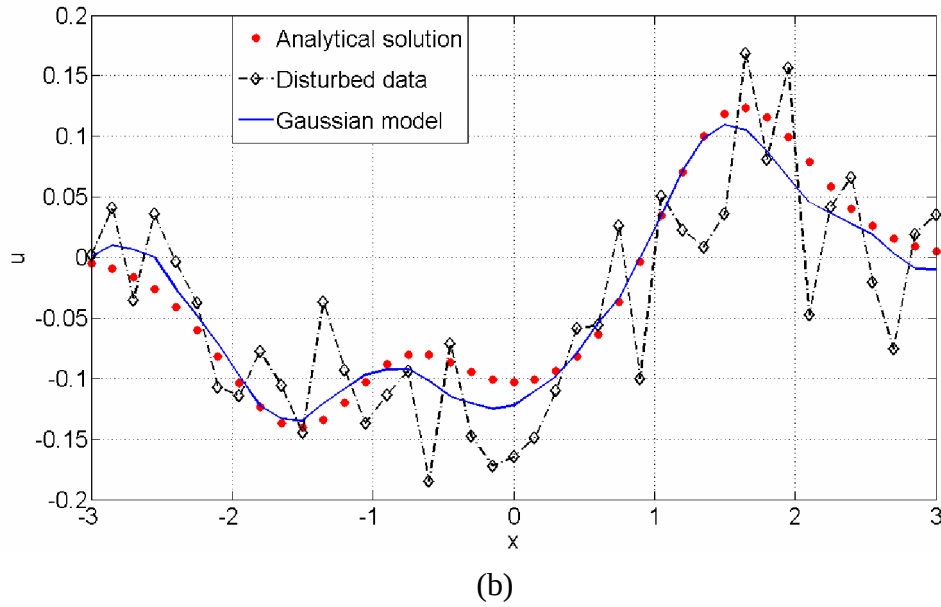
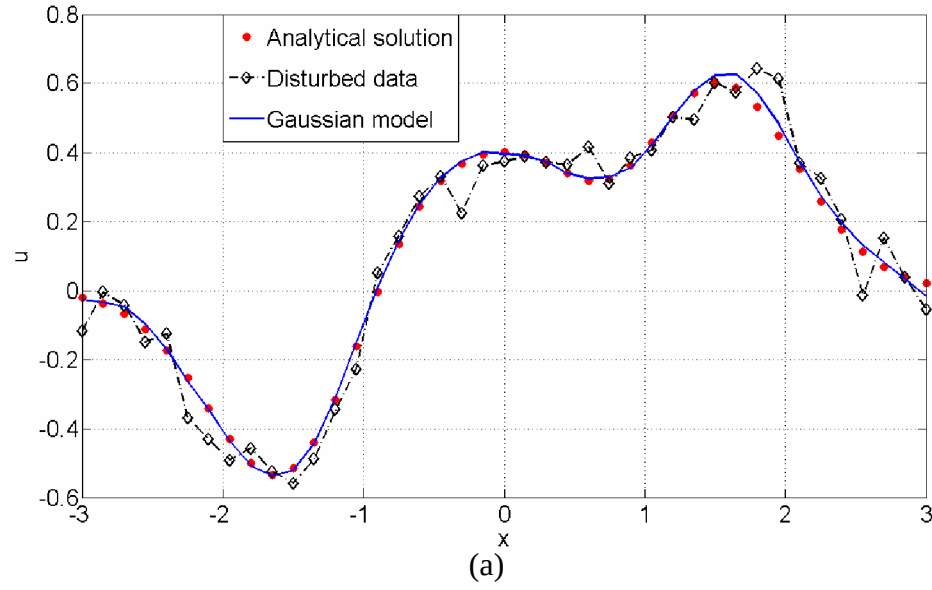


Figure 3.23: Data smoothing (a) on $y = -0.6$, $\theta = 0.5$, (b) on $y = 1.35$, $\theta = 0.5$.

In Figure 3.24, the results of 2D line Kriging with different correlation parameters are given for the test data. 2D line Kriging works reasonable for the aim of smoothing and the correlation parameter has an importance on the result. The advantage of this method is that it works much fast and easy to perform for large data sets. In Figure 3.25 a-b the plots are given for values of variable on constants lines $y = -0.6$ and $y = 1.35$ respectively. In this figure red dots show actual values of the variable and black diamonds show disturbed data to be smoothed. The continuous lines show smoothed data distribution by 2D line Kriging with the Gaussian model of different correlation parameters.

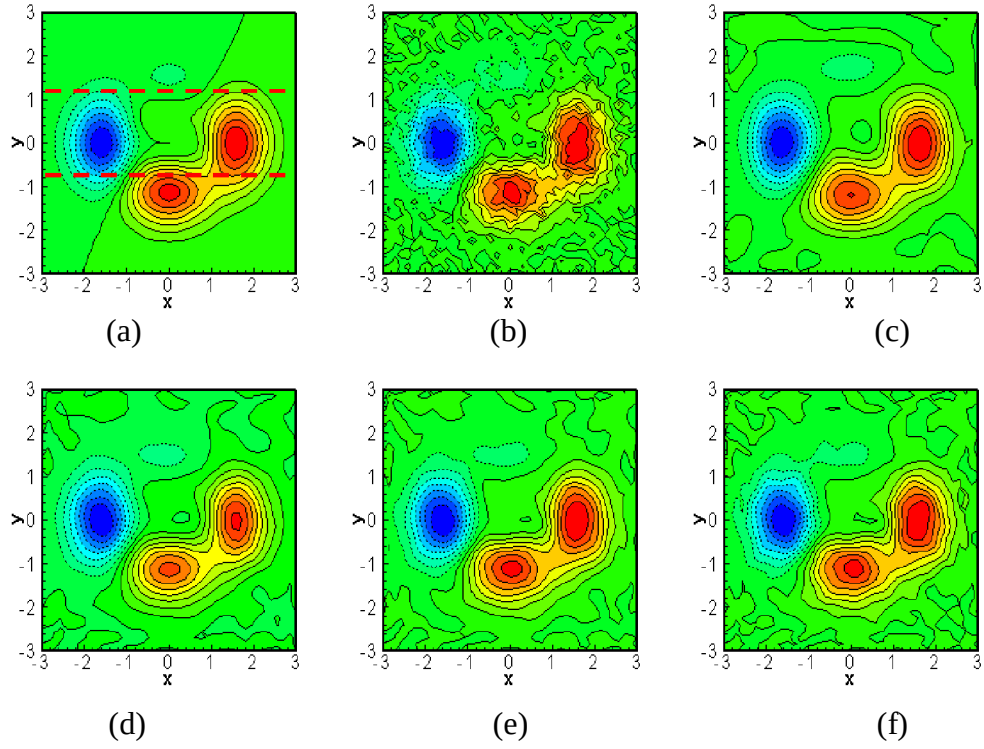
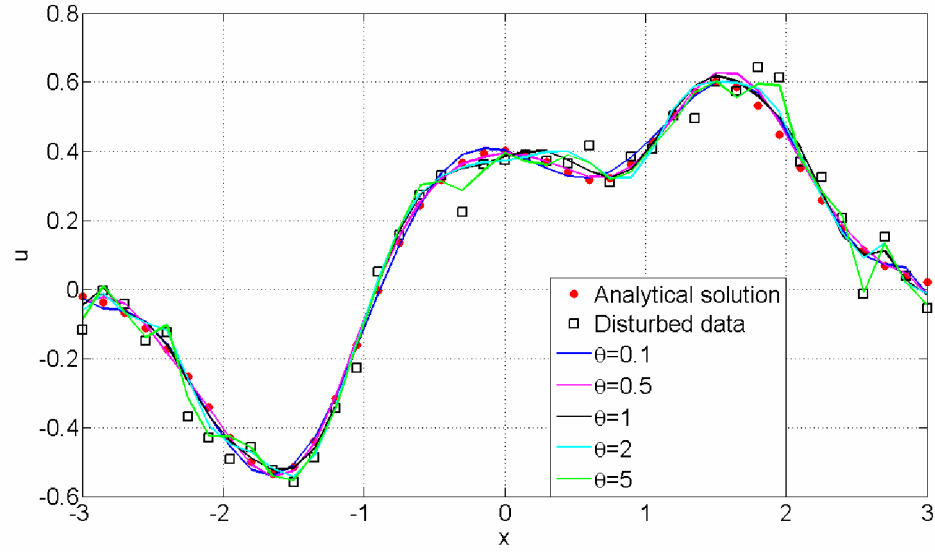
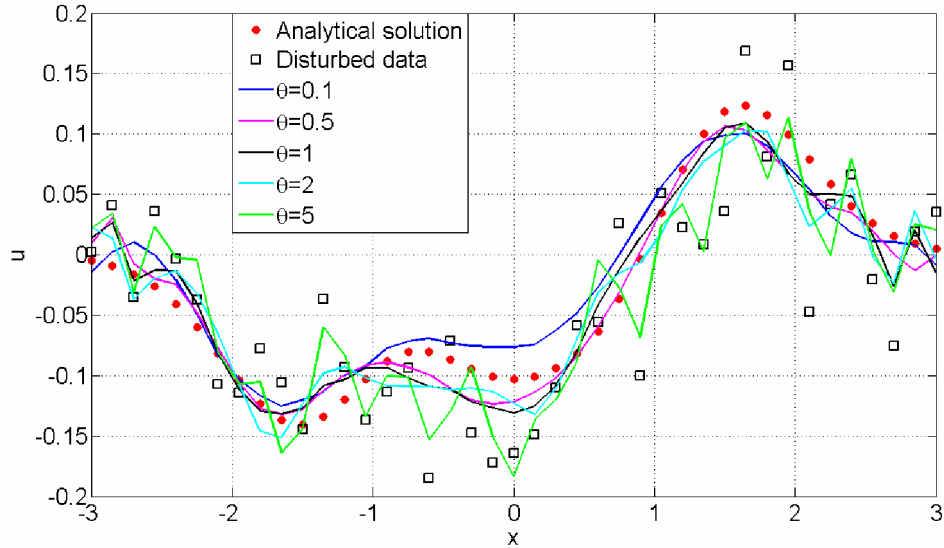


Figure 3.24: (a) Actual data, (b) disturbed data ($rms = 0.05$), (c) smoothed data by 2D line Kriging with the Gaussian model ($\theta = 0.1$) ($rms = 0.023$), (d) smoothed data by 2D line Kriging with the Gaussian model ($\theta = 0.5$) ($rms = 0.019$), (e) smoothed data by 2D line Kriging with the Gaussian model ($\theta = 1$) ($rms = 0.023$), (f) smoothed data by 2D line Kriging with the Gaussian model ($\theta = 2$) ($rms = 0.029$).



(a)



(b)

Figure 3.25: Data smoothing (a) on $y = -0.6$, $\theta = 0.5$, (b) on $y = 1.35$, $\theta = 0.5$.

Next, the noisy data is smoothed by digital filters with different values of cut-off frequencies and orders. In Figure 3.26, the contour plots of smoothing results by a 2D IIR Butterworth type low-pass, double-sided (zero-phase) filter with different cut-off frequencies are given in comparison with results of Kriging interpolation with $\theta = 0.5$ which gives the lowest *rms* error. Figure 3.26 depicts the importance of cut-off frequency for a digital filter. For a small cut-off frequency, a very smooth result can be obtained but it is significantly different from the actual data, i.e., it has a higher value of *rms* error than the noisy data! On the other hand, a large cut-off

frequency gives a relatively smooth but still noisy data and has a large value of *rms* error. When the cut-off frequency is selected appropriately, the digital filter gives a result that is quite smooth and close to the actual data with a low value of *rms* error.

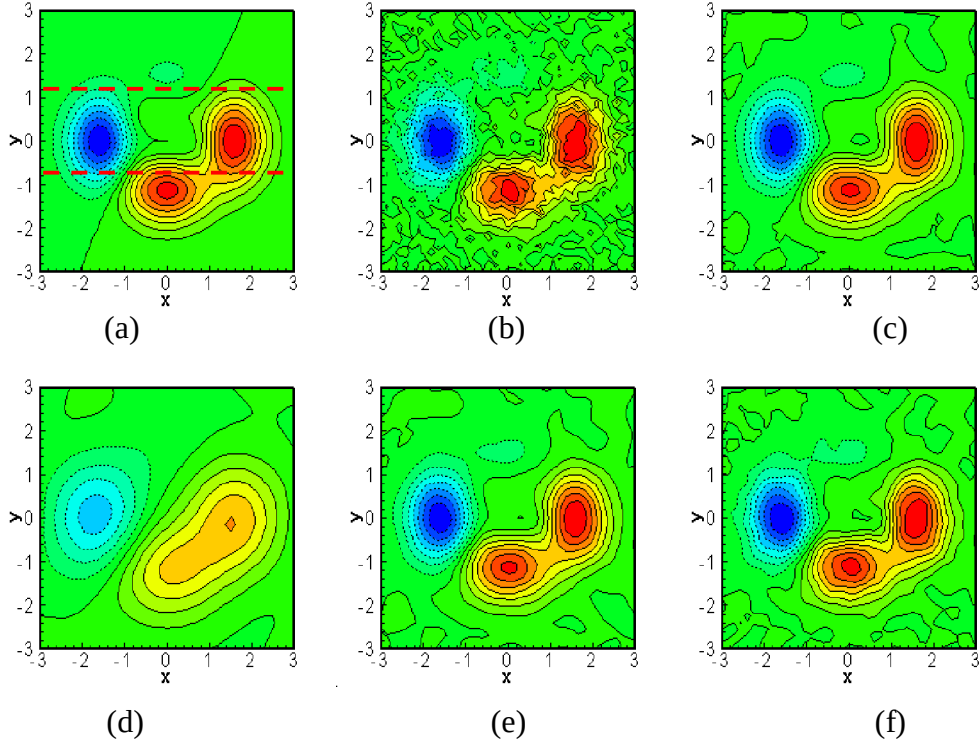
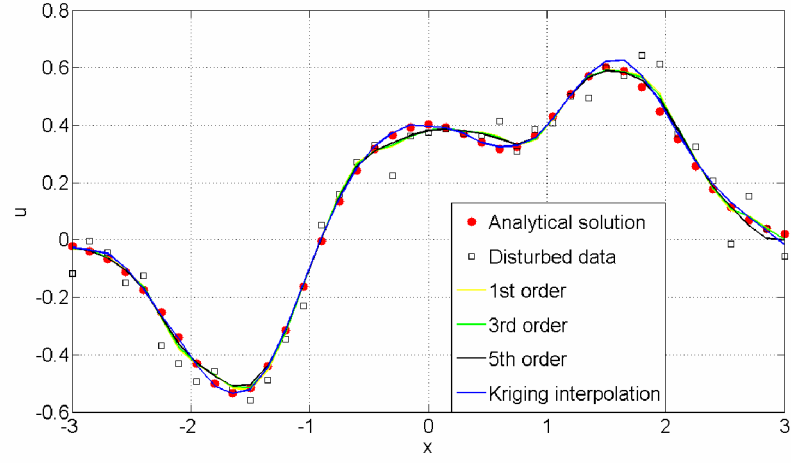
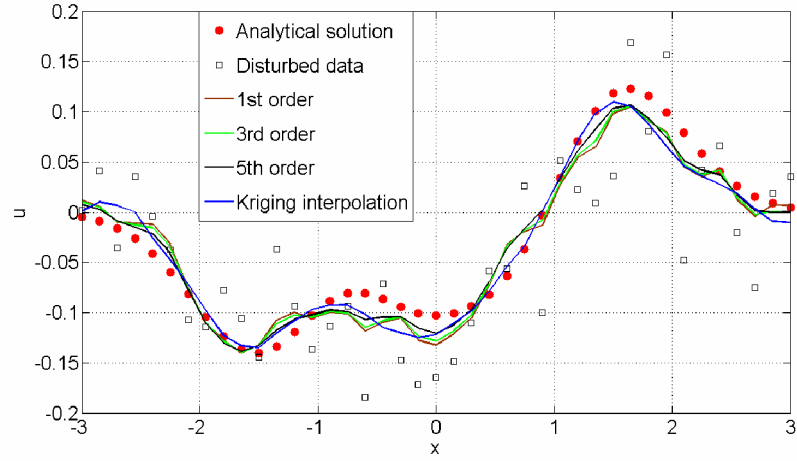


Figure 3.26: (a) Actual data, (b) disturbed data ($rms = 0.05$), (c) smoothed data by Kriging with the Gaussian model ($\theta = 0.5$) ($rms = 0.035$), (d) smoothed data by IIR Butterworth filter $f_c = 0.1$ ($rms = 0.085$), (e) smoothed data by IIR Butterworth filter $f_c = 0.3$ ($rms = 0.0154$), (f) smoothed data by IIR Butterworth filter $f_c = 0.5$ ($rms = 0.0227$).

In Figure 3.27-a-b the data are given for a comparison with Kriging interpolation on grid lines $y = -0.6$ and $y = 1.35$, respectively. Red dots show actual data and black diamonds show disturbed data to be smoothed. The blue line shows distribution of smoothed data by Kriging interpolation with the Gaussian model (correlation parameter 0.5). All of the other solid lines denote smoothed data by the low-pass, double-sided (zero-phase) 2D IIR Butterworth filter with a cut-off frequency of 0.3 for different filter orders.



(a)



(b)

Figure 3.27: Comparison of 2D Butterworth filter and Kriging interpolation. (a) $y = -0.6$, (b) $y = 1.35$.

In this thesis, we performed smoothing methods for PIV data set on constant z -planes, i.e., our methods are two-dimensional. We also investigate a three-dimensional version of Kriging interpolation for smoothing. Instead of smoothing on constant planes we built a 3D model and compared with a 2D model, and conclude that there is no significant effect of a 3D model.

4. ESTIMATION OF MISSING PLANES

The next task in this thesis was to increase the resolution of the smoothed PIV data. The PIV measurements are carried out for 5 different wall normal planes (z-planes), and it was necessary to enhance the resolution in wall normal direction for further computations (e.g., to find the cross-flow velocity, and to employ vortex identification methods such as the λ_2 -method). For this reason, we estimate the missing z-planes between existing (measured) planes. As in the previous section, we use POD and Kriging procedures to estimate these planes. In order to achieve resolution enhancement in z-direction, we take constant y-planes, so we have 2D planes consisting of x and z-directions. On these planes, there are 49 grid points in x-direction and only 5 grid points in z-direction. Using this data set, we increase the grid points in the z-direction from 5 to 31 or more. After working from plane to plane, a finer resolved 3D data set is obtained.

4.1. Estimation via POD

Once POD modes are obtained from the available PIV data, the missing z-planes can be estimated by reconstruction equations. In these equations, the spatio-temporal expansion coefficients $a_k(z,t)$ for a desired z value can be evaluated. In the reconstruction, a desired number of modes can be taken for estimation of missing planes. As it has been mentioned in the previous chapter, the first few most energetic modes are enough for the estimation. In this thesis, we typically used the first six modes.

4.2. Estimation via Kriging interpolation

Kriging interpolation can also be used for the estimation of the missing z-planes. Similar to the POD estimation, for each constant y-plane, a Kriging interpolation is built from existing data on the current plane and the resolution is enhanced in z-

direction. In this thesis, we used the exponential Kriging model for resolution enhancement, as existing z -planes are already smoothed by the Gaussian Kriging model.

Figure 4.1 shows the contour plots of the streamwise velocity for new planes ($z = 1.125$, $z = 3.375$) predicted by linear interpolation, Kriging interpolation and POD ‘interpolation’.

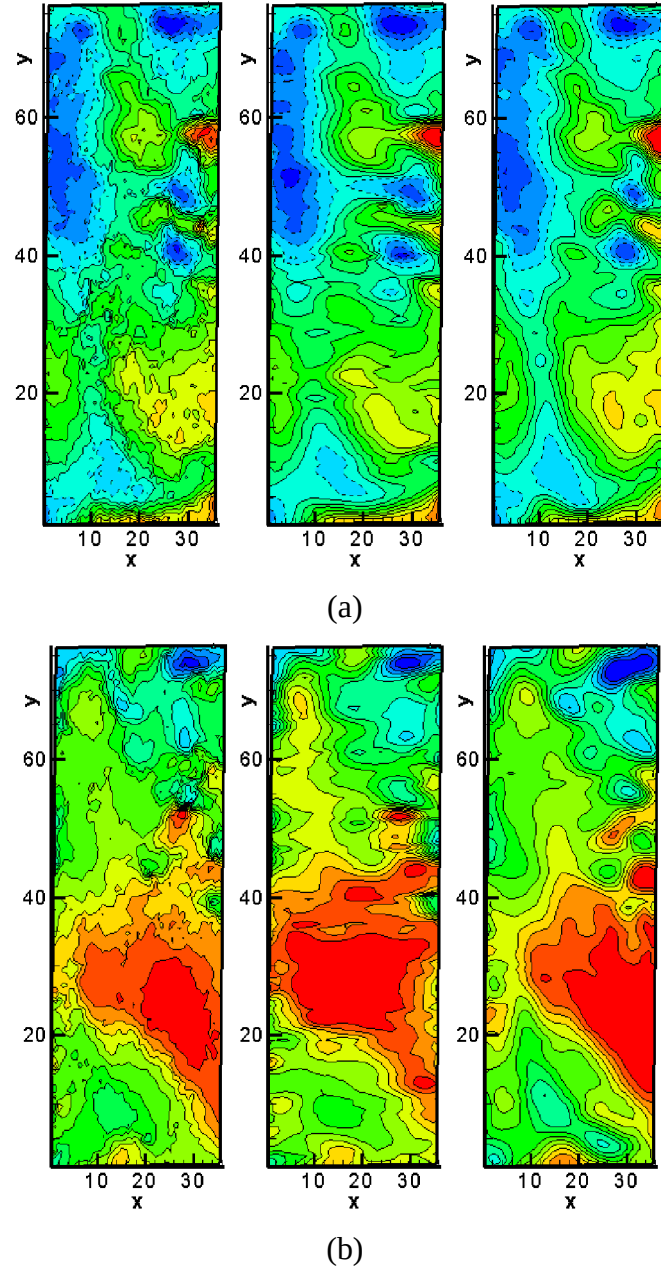


Figure 4.1: The contours of the streamwise velocity (a) $z = 1.125$ plane, (b) $z = 3.375$ plane. From left to right: linear interpolation, Kriging interpolation, POD ‘interpolation’.

It is noted in Figure 4.1 that all the interpolations are done separately for each constant y plane and then all interpolated y -planes are combined into a 3D data set to extract constant z -planes. If there would be an irregularity on any y -plane, this should appear in the $z = \text{constant}$ plane.

In Figure 4.2 the contour plots of the streamwise velocity of original PIV data and its enhancement by Kriging interpolation and POD “interpolation” are given for a constant y -plane, $y = 5$.

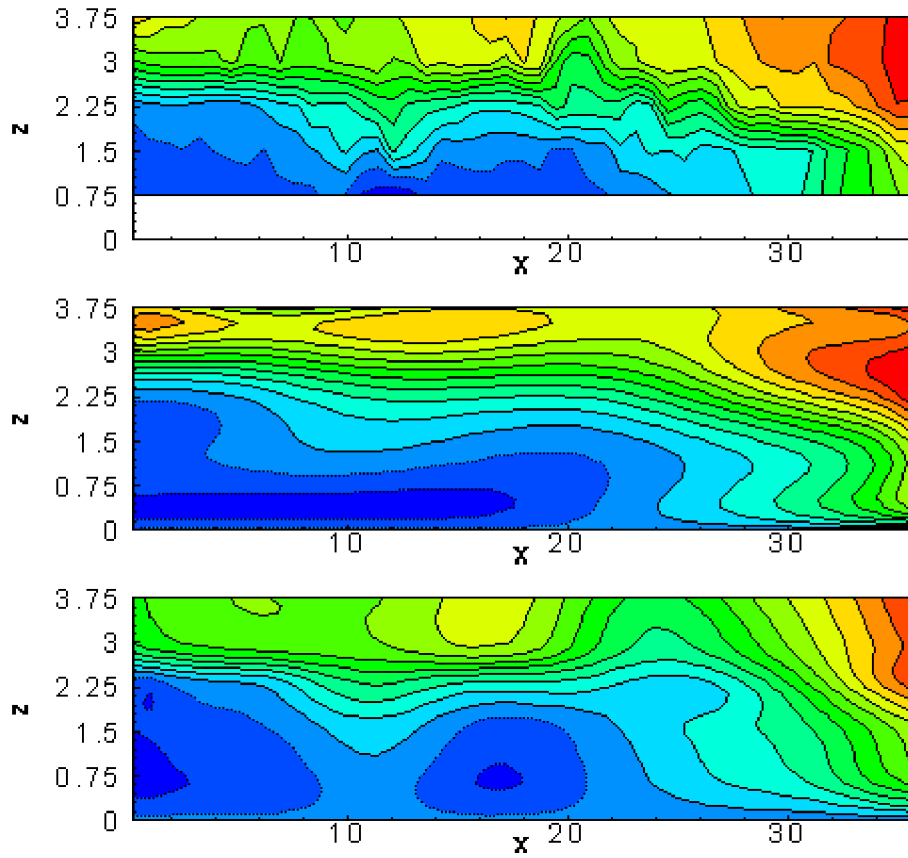


Figure 4.2: The contours of the streamwise velocity for $y = 5$ for the first snapshot, from top to bottom: original PIV data, enhancement of PIV data by Kriging interpolation, enhancement of PIV data by POD “interpolation”, respectively.

In Figure 4.3 the contour plots of the streamwise velocity of original PIV data and its enhancement by Kriging interpolation and POD “interpolation” are given for a constant x -plane, $x = 10$. In this figure we noticed that the circled region in the original PIV data is not physically possible, because of a measurement error. Kriging interpolation and POD “interpolation” cure this region and give more realistic flow field.

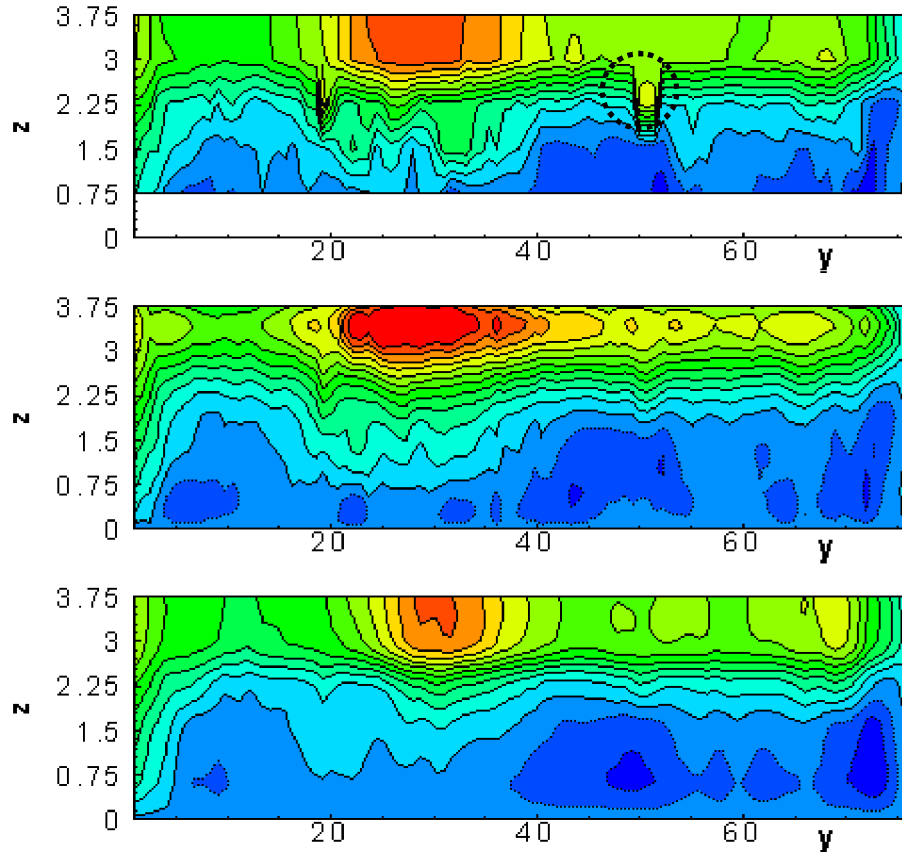


Figure 4.3: The contours of the streamwise velocity for $x = 10$. From top to bottom: original PIV data, enhanced data by Kriging interpolation, enhanced data by POD “interpolation”.

4.3. Estimation via Inverse-distance Weighting Interpolation

In Figure 4.4, the estimation of missing planes is given for IdW interpolation in a comparison with Kriging and linear interpolation. It is noted that IdW interpolation can not be an alternative to Kriging interpolation even though it is computationally efficient.

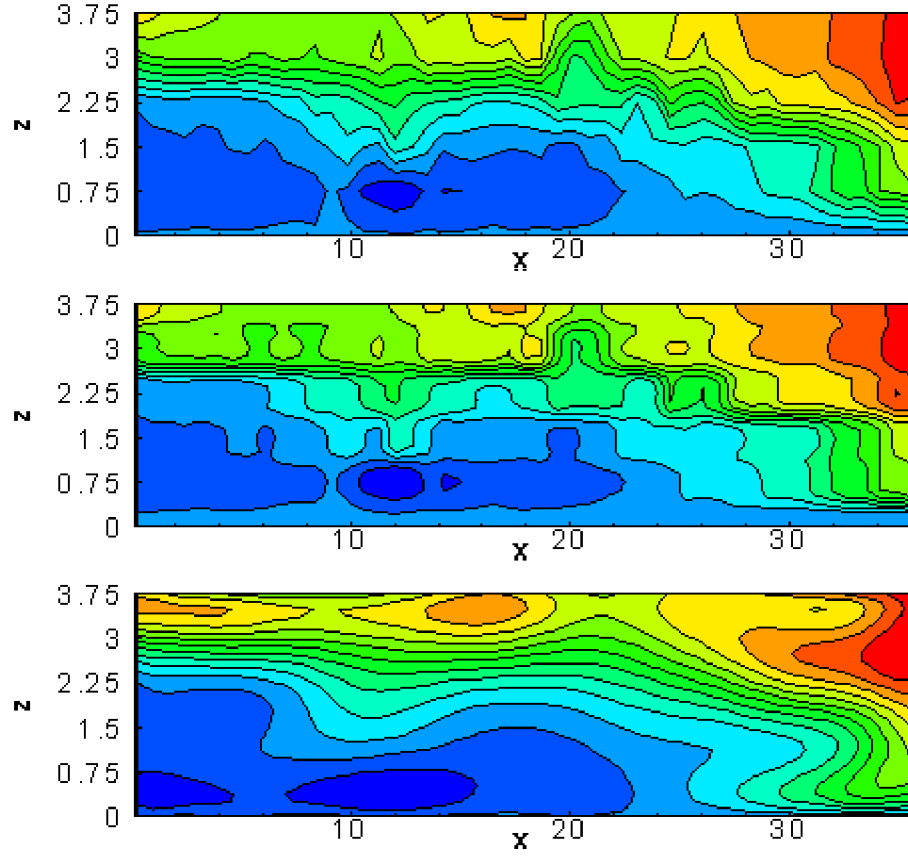


Figure 4.4: The contours of the streamwise velocity for $y = 10$. From top to bottom: linear interpolation; Inverse-distance Weighting interpolation ($p = 3$) and Kriging interpolation.

4.4. Calculation of the cross-flow velocity component

The streamwise and spanwise velocity components have been measured using PIV. In the flow field the cross-flow (w) velocity is missing. Once resolution of the experimental data has been increased, the cross-flow velocity can be calculated via divergence-free continuity equation Eq.(4.1), employing highly resolved interpolated the streamwise and the spanwise velocity (u,v) components.

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \quad (4.1)$$

Two terms of Eq.(4.1), have to be calculated and then numerical integration is needed to calculate w . We performed third order numerical derivation and

integration schemes for these calculations and the accuracy of these numerical schemes are investigated on a test data (see in Appendix).

The cross-flow velocity is calculated via divergence-free continuity equation from the original PIV data and enhanced data set by Kriging interpolation and POD “interpolation”, and the calculated cross-flow velocity contours on $z = 3.0$ for the first snapshot are shown in Figure 4.5. If the cross-flow velocity is calculated from original PIV data its contour plot is much noisy and does not show a realistic flow field. However, if it is calculated from a smoothed and enhanced data set its contour plots are much smoother and show a realistic flow field.

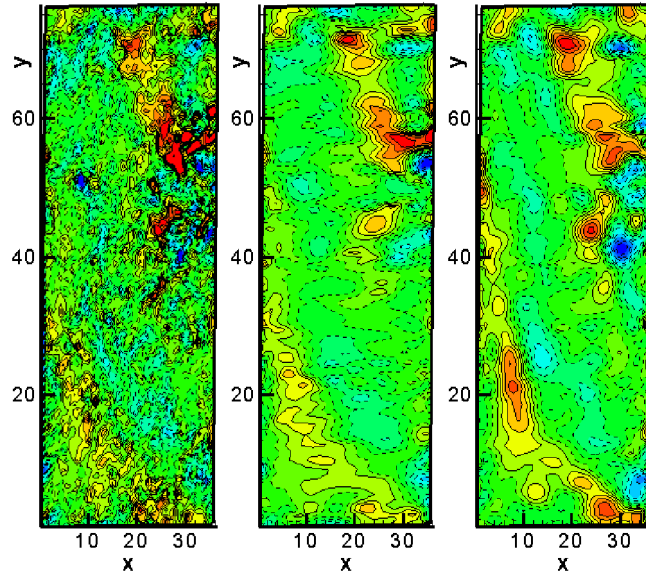


Figure 4.5: The contours of the cross-flow velocity for $y = 5$. From left to right: calculated cross-flow velocity from the original PIV data, enhanced cross-flow velocity by Kriging interpolation, enhanced cross-flow velocity by POD “interpolation”.

In Figure 4.6 in plane velocity vectors of original PIV data, enhanced by Kriging interpolation and POD “interpolation” are given on the plane $y = 5$. In this figure vectors denote the cross-flow velocity (w) in z -direction and the streamwise velocity (u) in x -direction, and colours denote the spanwise velocity (v). In this figure it can be seen that the original PIV data has only 5 grid points in z -direction and gappy z grids are estimated by Kriging and POD methods.

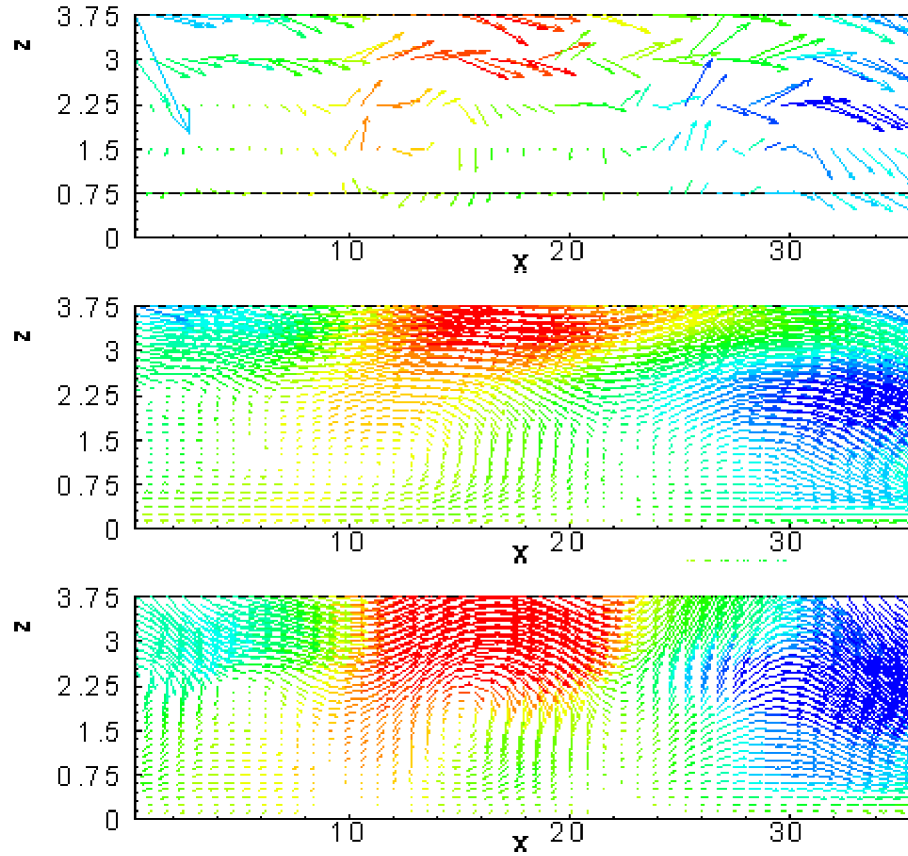


Figure 4.6: In plane velocity vectors (u, w) on plane $y = 5$. Colours denote the spanwise velocity component (v) . From top to bottom: original PIV data, Kriging interpolation and POD “interpolation”.

Although we enhanced the resolution of the whole PIV data set for all snapshots we can not present a complete data set here. The complete, animated and compared PIV data set and enhanced results by Kriging interpolation and POD “interpolation” can be found on <http://www2.itu.edu.tr/~ceklih/research.htm>, <http://atlas.cc.itu.edu.tr/~guneshasa>, (December 2006)

5. RECONSTRUCTION OF “BLACK ZONES”

In this section, the capability of Kriging interpolation for the reconstruction of large zones of missing data (so called “black-zones”) was investigated. We shall demonstrate that if correct variogram and correlation parameters are selected Kriging interpolation can reconstruct such continuous missing regions in an accurate way.

5.1. Black zone in test data

For a detailed investigation, first we create a synthetic test data set using an analytical function (41 grid points for both x and y -directions, respectively), and in the data set we create an artificial black zone in a rectangular region by removing existing grid points in the test data. Using Kriging, this black zone is reconstructed. A Gaussian model is constructed using the available data outside of the black zone. In Figure 5.1a, the contour plot of the original data is shown and the large missing region (black zone) is outlined by a dashed line. In Figure 5.1b, the contour plot of the reconstructed data is given. When we compare these two contour plots we see that the black-zone region is recovered reasonably well by Kriging interpolation.

Because we know the actual values of interpolated data, we can calculate an *rms* error value for the estimation using Eq.3.9. For this size of black zone the *rms* error of the reconstruction is $1.48 \cdot 10^{-3}$. A detailed comparison is shown in Figure 5.2, where the original data versus the reconstructed data on a constant line ($x = -1,05$) is given for the variable.

In Figure 5.2 red dots show actual function values and continuous line shows the reconstructed values on the constant line $x = -1,05$. The black zone is marked between two dashed lines. In this figure it can be seen that the reconstructed data is closely follows to the actual data.

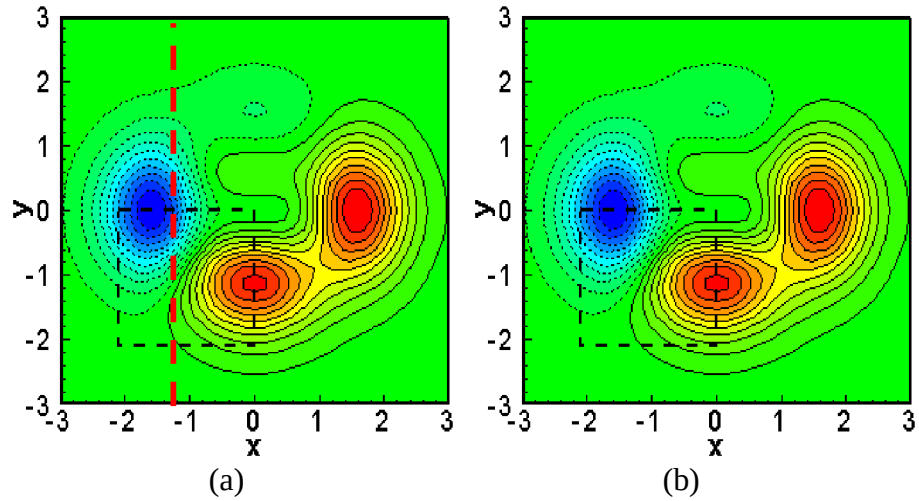


Figure 5.1: Kriging interpolation for a black zone. The Gaussian variogram model is used. (a) original data, (b) Kriging interpolated data (compare the dashed black-zone), ($rms = 1.48 \cdot 10^{-3}$).

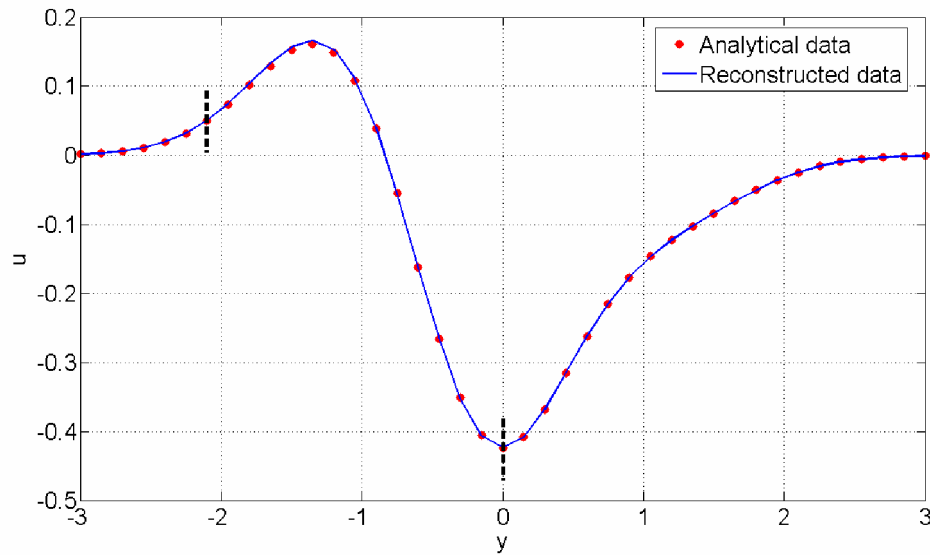


Figure 5.2: The original and the reconstructed function values on a selected constant line ($x = -1,05$).

For this size of black zone Kriging interpolation works really amazingly, so we increase the size of black zone and then try to reconstruct it. In Figures 5.3-4, the contour plot of original data set is shown and the black zone is marked by dashed line and the contour plot of the reconstructed data is given for different sizes of black zones.

When we compare black zones of varying sizes, it can be easily seen that as the size of the black zone increases, deviations of the reconstructed data from actual data

also increases. However, even for very large black zones, the reconstruction quality is impressive. In Table 5.1 the *rms* values of the reconstruction for different sizes of black zones are given. We believe that the success of Kriging interpolation in recovering black zones is due to its ability to follow the gradients of data accurately.

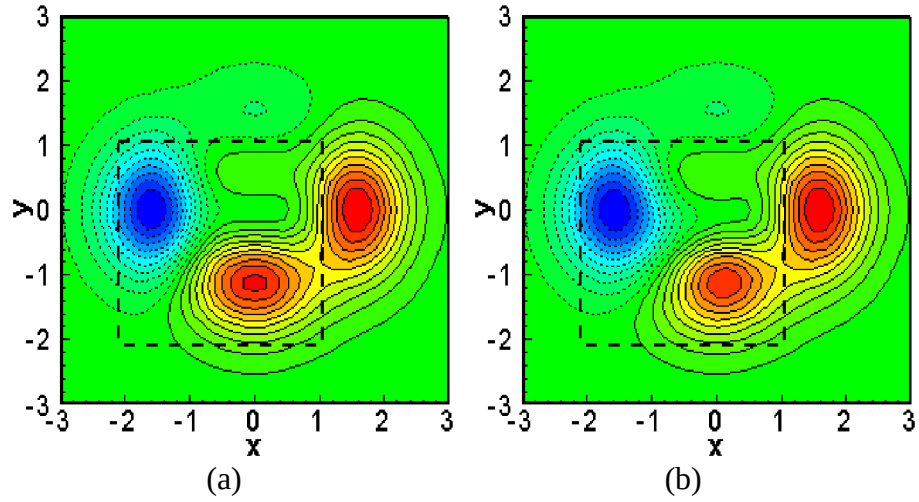


Figure 5.3: Kriging interpolation results employing the Gaussian variogram model for black-zone, (a) original data, (b) Kriging interpolated data,(compare the dashed black zone), ($rms = 6.177 \cdot 10^{-2}$).

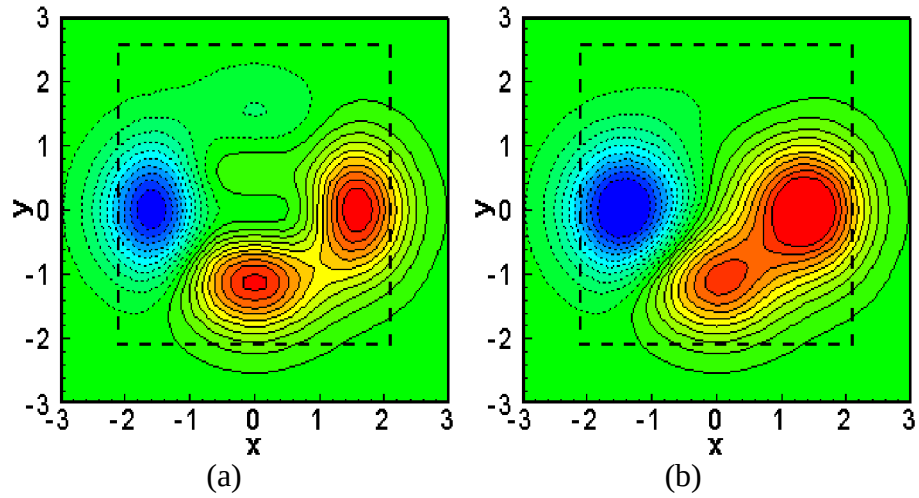


Figure 5.4: Kriging interpolation results employing the Gaussian variogram model for black-zone, (a) original data, (b) Kriging interpolated data, (compare the dashed black-zone), ($rms = 0.134$).

Table 5.1: The *rms* error values for reconstruction of different size of black zones.

Area of black zone	<i>rms</i> error of reconstruction
12.25%	$1.48 \cdot 10^{-3}$
18.7%	$1.408 \cdot 10^{-2}$
26.7%	$6.177 \cdot 10^{-2}$
49%	0.133
53.6%	0.134
59.5%	0.184

Our numerical investigations show that the black zones have been reconstructed for the test data remarkably well by Kriging interpolation. However, our test data is quite smooth and continuous. On the other hand, instantaneous experimental data fields are generally somewhat noisy. In this context, we investigate the capability of Kriging to reconstruct a black zone with noisy data. With this aim, we first disturb our test data and then create an artificial black zone by removing existing grid points. Then, we try to reconstruct the black zone from the noisy data. The result is given in Figure 5.5. Although the available data set is noisy, after reconstruction by the Gaussian model, we obtain a smooth region in the black zone. In this case the *rms* error value is 0.1, however for smooth data it was $1.48 \cdot 10^{-3}$ for the same size of black zone.

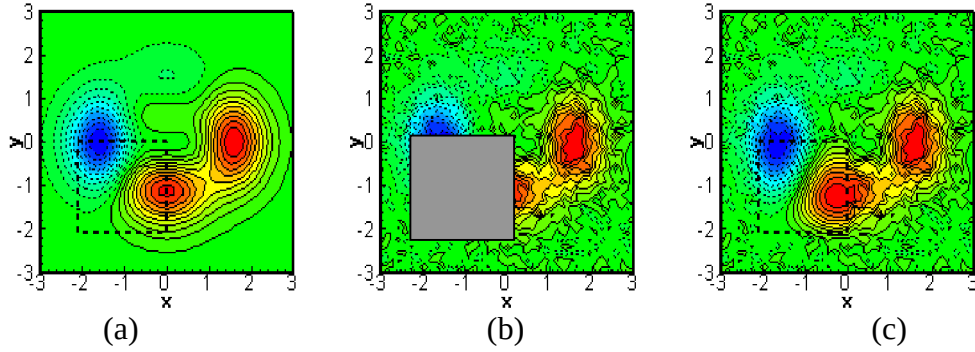


Figure 5.5: Kriging interpolation results for a black zone with noisy data. The Gaussian model is used. a) actual data, b) disturbed data with black zone, c) reconstructed data.

5.2. Black zone in PIV data

When we reconstruct black zone successfully in the test data by Kriging interpolation then we investigate it for a black zone in the experimental data set. A black zone was created in a rectangular region by removing all information the

region on the z -plane, $z = 0.75$ as shown by dashed rectangular in Figure 5.6. In order to employ Kriging interpolation method faster and systematically we eliminated the upper side of the plane. Then Kriging interpolation with the Gaussian model and different correlation parameters was employed to reconstruct to black zone in the experimental data set. Results have shown that correlation parameter is important to obtain a reasonable result. In Figure 5.6 reconstruction results are given by the contours of the streamwise velocity for different values of correlation parameter.

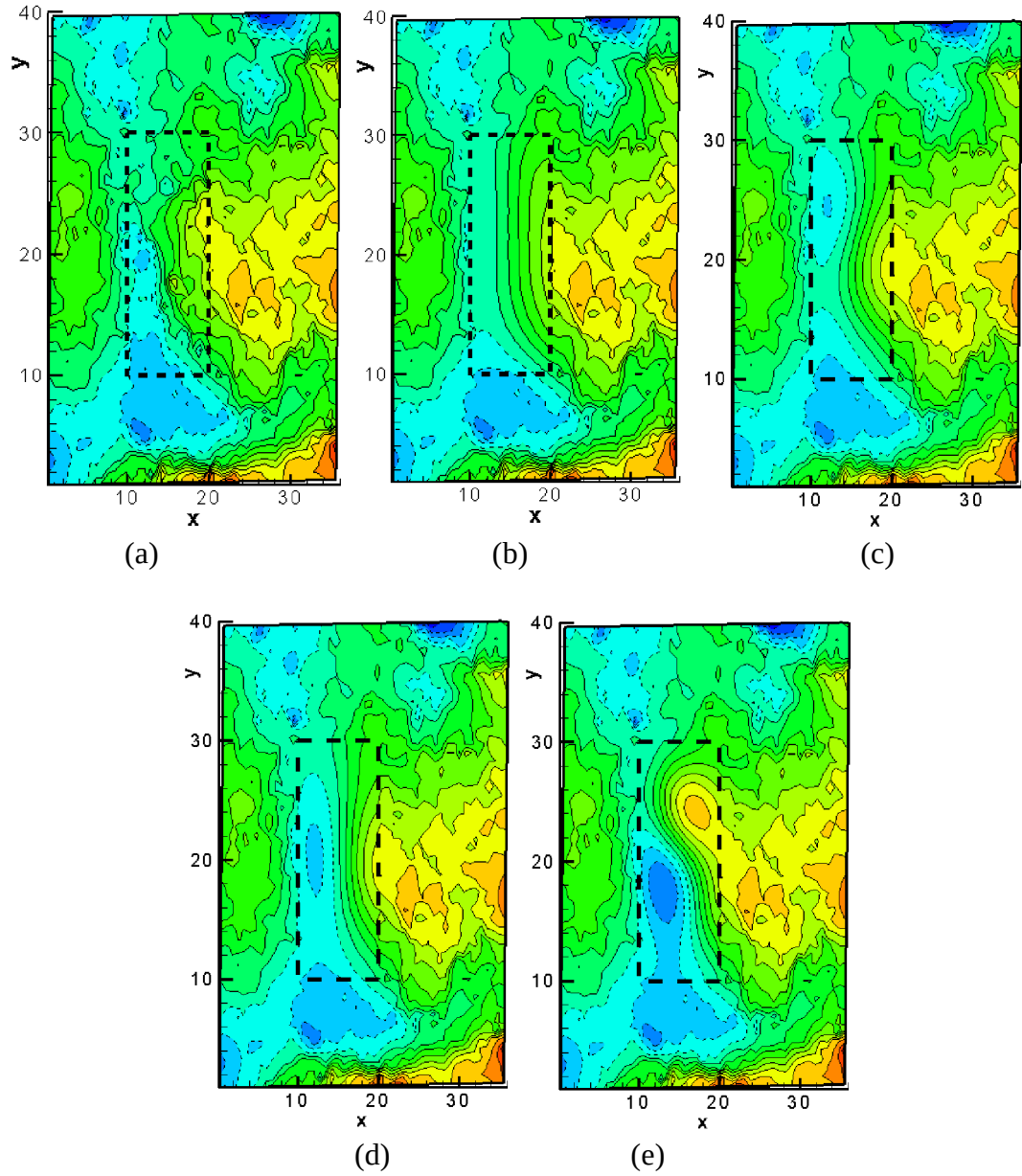


Figure 5.6: Kriging interpolation results for the contours of the streamwise velocity component, employing the Gaussian variogram with different correlation parameter for black-zone in experimental data, a) original PIV data, b) $\theta = 0.1$ c) $\theta = 0.2$, d) $\theta = 0.3$, e) $\theta = 0.5$.

In Figure 5.6 it can be seen that reconstruction region is quite smooth although the outside of black zone is much noisy. In this case the idea is that smooth outside of the black zone before reconstruction then reconstruct the black zone from smoothed data. With this aim the experimental data in the Figure 5.6 with black zone was smoothed by the Gaussian model with $\theta = 0.5$ and original experimental data and its smoothed version with black zone are given in Figure 5.7.

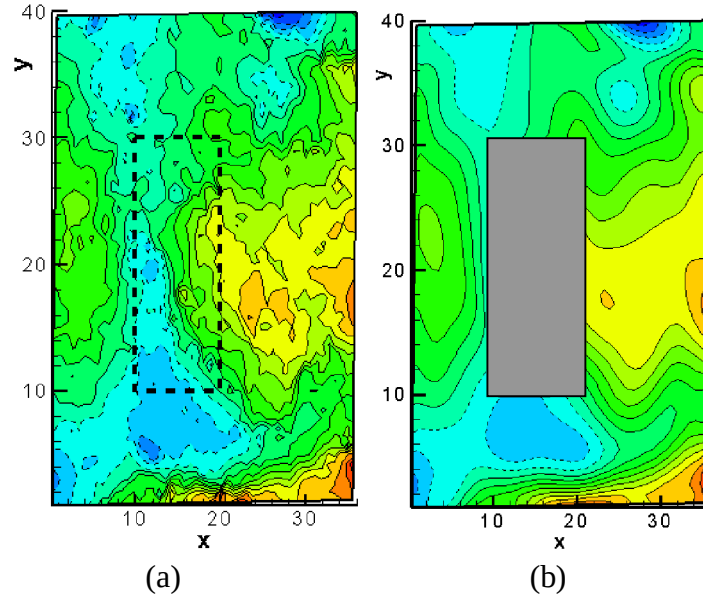


Figure 5.7: Smoothing the outside of black zone. (a) Original PIV data, b) smoothed by the Gaussian model $\theta = 0.5$.

Once outside of black zone is smoothed black zone is reconstructed by Kriging interpolation by different values of correlation parameter, and results are given in the Figure 5.8. When Figure 5.6 and 5.8 are compared a considerable difference is not observed, only at the boundary of black zone and outside of it the contour lines are more continuous in Figure 5.8.

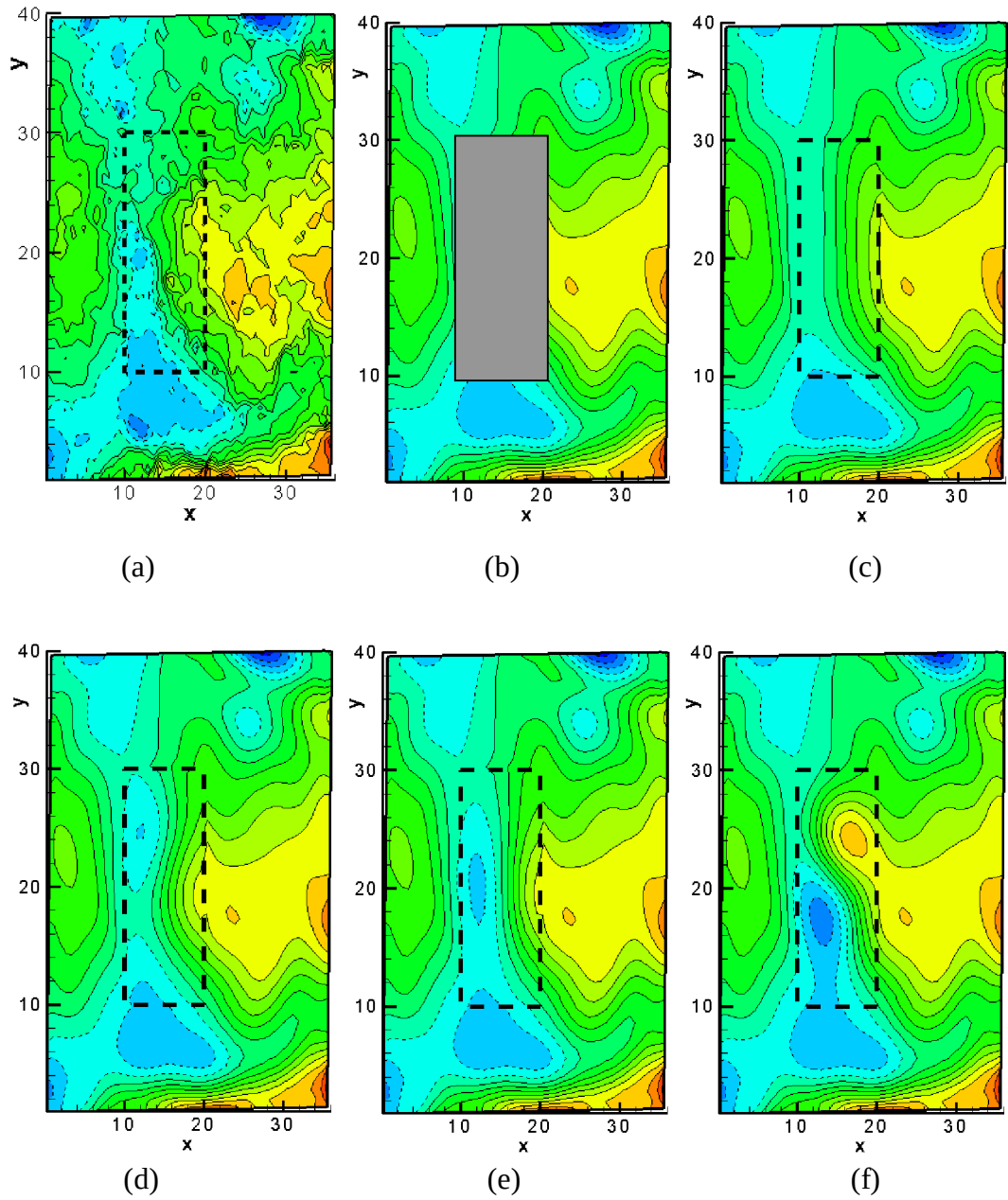


Figure 5.8: Kriging interpolation results for the contours of the streamwise velocity component, employing the Gaussian variogram with different correlation parameter for black-zone on the smoothed experimental data, a) original PIV data, b) smoothed PIV data c) $\theta = 0.1$ d) $\theta = 0.2$, e) $\theta = 0.3$, f) $\theta = 0.5$.

6. APPLICATION OF KRIGING INTERPOLATION FOR PROBLEMS

When we conclude Kriging interpolation estimates any desired point in a data set accurately in a robust way we investigate a possibility to use Kriging interpolation for computer experiments and design for example optimization problems. In the field of combinatorial optimization the aim is developing efficient techniques for finding global minimum or maximum values of a function of many independent variables. With this context we construct a combinatorial optimization algorithm by using Kriging interpolation in a combination with Simulated Annealing (SA) algorithm. In our optimization algorithm there are two independent algorithms, so it is a meta-algorithm. The most common example of the combinatorial optimization is the travelling salesman. In this problem, a salesman must visit a large number of cities and a means of calculating the cost of travelling cost between and two cities. The problem is finding the route of salesman that total cost will be the lowest and the salesman will pass any city just one time and finally has to turn back the starting point. If the salesman starts with a random itinerary, he can then pair-wise trade the order of visits to cities, hoping to reduce the cost with each exchange. The difficulty with this approach is that while it rapidly finds a local minimum, it cannot get from there to the global minimum. A local minimum or relative minimum, is a minimum value within some neighbourhood that need not be (but may be) a global minimum that the smallest value over its entire range.

6.1. Simulated Annealing

Simulated Annealing (SA) is a generic probabilistic meta-algorithm for the global optimization problem, was invented by **Kirkpatrick et al., (1983)**. SA's major advantage over other methods is an ability to avoid becoming trapped at local minima.

SA is based on an analogy between techniques involving heating and controlled cooling of a material to increase the size of its crystals and reduce their defects (the annealing process in metallurgy) and the search for a minimum in a more general system. The heat causes the atoms to become unstuck from their initial positions and wander randomly through states of higher energy, the slow cooling gives them more chances of finding configurations with lower internal energy than the initial one. By analogy with this physical process in statistical process, each step of the SA algorithm replaces the current solution by a random "nearby" solution, chosen with a probability that depends on the difference between the corresponding function values and on a global parameter T (called the temperature), that is gradually decreased during the process. The dependency is such that the current solution changes almost randomly when T is large, but increasingly "downhill" as T goes to zero. The allowance for "uphill" moves saves the method from becoming stuck at local minima.

The algorithm is based upon that of **Metropolis et al., (1953)**, which was originally proposed as a means of finding the equilibrium configuration of a collection of atoms at a given temperature. The connection between this algorithm and mathematical minimization was first noted by **Kirkpatrick et al., (1983)** who proposed that it form the basis of an optimization technique for combinatorial problems.

The subject of statistical physics is to observe the behaviour of the system of atoms in thermal equilibrium at a certain temperature. This behaviour is defined by the atomic positions r_i and system is weighted by its Boltzmann probability factor as in Eq.6.1,

$$\exp(-E\{r_i\}/k_B T) \quad (6.1)$$

Where, $E\{r_i\}$ is the energy of the configuration, k_B is Boltzmann's constant, T is temperature.

It is important to understand the state of system at the low temperature limit, i.e., if atoms are solid or fluid. At low temperature properties of a body are getting clear because at low T the Boltzmann distribution collapses into the lowest energy state.

However, to find ground state the application steps of annealing also are important, for example the first substance should have melted and then temperature should have being lowered and a long time should spent at the freezing temperature. If it is not done substance can not reach to the optimal structures and they will have defects. It is an optimization problem to find low temperature state of a system.

Metropolis et al., (1953) offered an algorithm to determine behaves of atoms at a certain temperature. The algorithm is as in following. An atom is given a small random displacement and energy change ΔE is determined for the new point of atom, when $\Delta E \leq 0$ the new point is accepted and this new configuration is used for the following step. If $\Delta E > 0$ the new point is accepted probabilistically $P(\Delta E) = \exp(-\Delta E / k_B T)$. Uniformly distributed random numbers between interval (0,1) are produced and compared with the $P(\Delta E)$, the new configuration is accepted only when $P(\Delta E)$ is greater then the random number, otherwise the original configuration is remained to start for the next step. These steps are repeated many times, i.e., long enough then a steady state is reached. An analogy is used to construct an optimization algorithm (**Kirkpatrick et al., 1983**).

A Kriging model is built and validated from available data. This available data may come from experiments or computer computations such as CFD analyzes and the task is determine the optimum variables for this data set. For the experiments or computations it is important to determine the design sites, i.e., for which values of variables the experiments or computations should have been done in order to the most efficiently control and reduce the uncertainties. We used Latin Hypercube Sampling (LHS) algorithm to determine design sites. LHS is based on random numbers and ensures that all portions of the vector space are represented (**McKay et al, 1979**).

When the design sites are determined by LHS through the vector field we run computer code and use as input for the optimization algorithm and build Kriging model based on this input. Once Kriging model is build we start with initial values of variables in the vector filed of variables and predict a value for this point by Kriging interpolation. The initial values and their predicted solution are taken as the best configuration of the problem. Then we change the values of the variables

nearby the current values and estimate a solution for this configuration. If the last configuration is better than the best configuration we take it as the best configuration, if not then it is treated probabilistically. The probability function is in Eq.6.2. Uniformly distributed random numbers are produced between interval (0,1) and compared with probability value and when probability value is greater than random value the current configuration is taken as the best configuration. These steps are repeated for a long time and finally a best solution is obtained.

$$P = \exp[(e - e_n)/T] \quad (6.2)$$

where, e is the best value and e_n is the last estimated value, T is the temperature of the problem and decreasing during the optimization process, it can be calculated by Eq.6.3.

$$T = T_0(1 - k/K)^c \quad (6.3)$$

where, T_0 is the initial temperature and should be given carefully, c is a parameter to control cooling rate (if $c = 1$ cooling is linear), k is the number of evaluated steps and K is the maximum evaluation that given at the beginning of the optimization. Figure 6.1 shows the change of the temperature (cooling) with the iteration number.

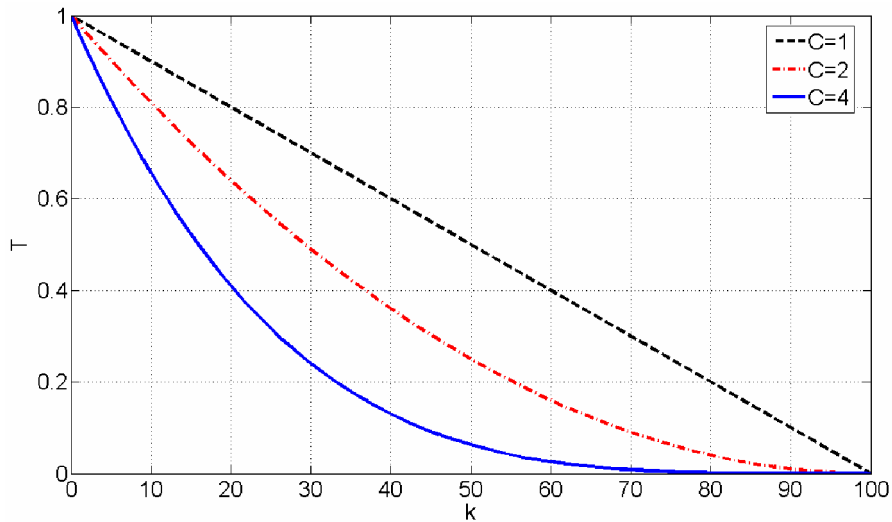


Figure 6.1: Temperature of system.

As it can be seen in this figure when c equals to one the cooling is linear and if we use larger values of c then we obtain faster cooling. We use the probability function

when estimated value is higher then best solution, in this case $(e - e_n)$ is always negative. Therefore as the power of exponential function we always have negative value in Eq.6.2 so as T is decreasing the probability value also will be decreasing. When P value is low, in general random generated number will be higher than P so in this case current configuration will not be accepted when it is not better then the best one. Therefore the algorithm may stuck in that configuration. In this manner our task is determine the most convenient parameters in the optimization algorithm. In Figure 6.2 a-c P values are given with evaluation numbers for different parameters ($T_0 = 1 - c = 1$, $T_0 = 1 - c = 2$ and $T_0 = 10 - c = 1$ respectively).

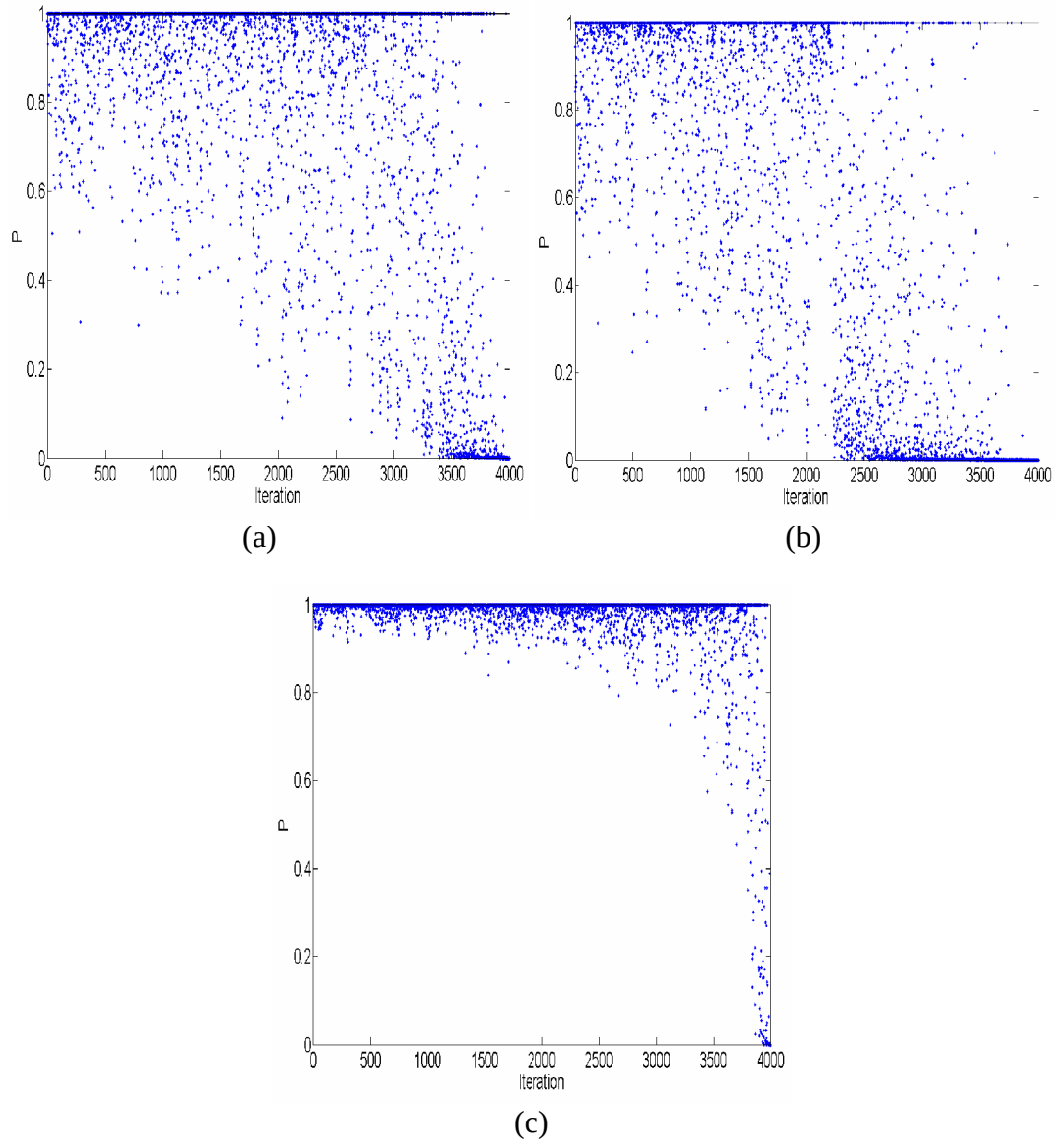


Figure 6.2: Probability values, a) $T_0 = 1, c = 1$, b) $T_0 = 1, c = 2$, c) $T_0 = 10, c = 1$.

In Figure 6.3 a-c an example for optimization process is given for different parameters that are given for probability values as in Figure 6.2 a-c. We create a test data for 1681 grid points by Eq.3.10 to evaluate our optimization algorithm. The actual solution contours of test data are plotted in Figure 6.3 a-c. Then we build test data again only for 50 grid points. Location of these grid points are determined by LHS algorithm and these grid points are shown by black dots in Figure 6.3 a-c.

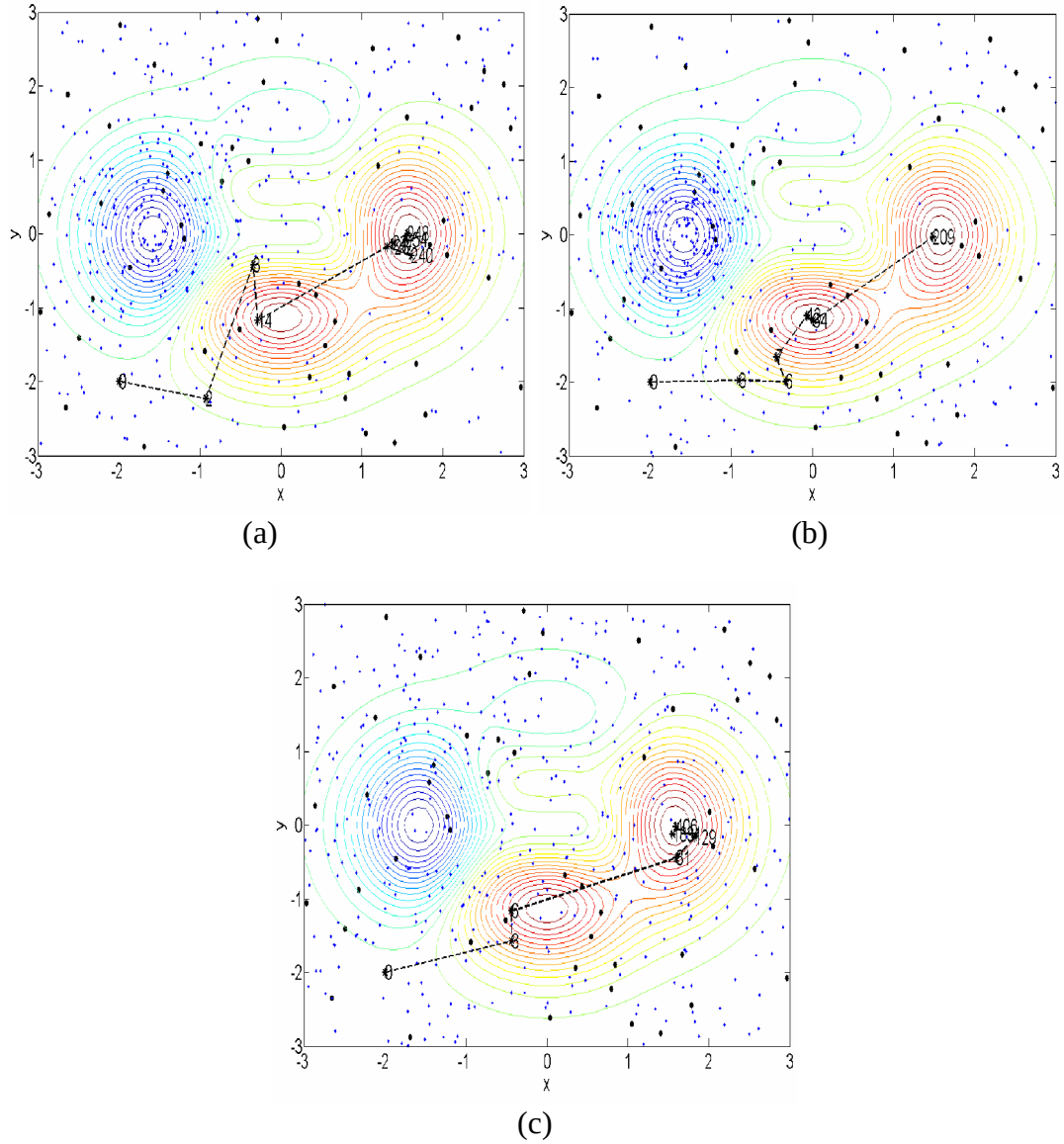


Figure 6.3: Optimization process, a) $T_0 = 1, c = 1$, b) $T_0 = 1, c = 2$, c) $T_0 = 10, c = 1$.

A Kriging model is built by using the available data i.e., values at 50 grid points. And this model is validated by re-calculation and comparison the existing data. By using this Kriging model in our optimization process our task is determine the highest value of the variable in the data set. The highest point of the actual solution

is at $(x, y) = (1.8, 0)$. In Figure 6.3 a-c blue dots show tried sites by optimization algorithm and the trajectory of the best points during the optimization process is given by dashed line and numbers of evaluation when a best point is found are given. When we compare Figure 6.2 a-b it can be seen that for same initial temperature for larger c value P values reach to zero more rapidly and it effects the location of tried sites that can be seen in Figure 6.3 a-b. In Figure 6.3 a there is a better distribution of tried sites in Figure 6.3 b there are even any un-tried regions. In Figure 6.3 a it can be seen that for a larger initial temperature we obtain relatively larger P values and it reaches zero at the end of optimization process. This behaviour results the well distributed tried sites though the domain that it can be seen Figure 6.3 c.

7. CONCLUSION

In this thesis, POD, Kriging interpolation and digital filters are investigated in order to smooth and enhance the resolution of an experimental data for a laminar separation bubble of an airfoil. POD modes were extracted from experimental data. Different from original POD modes in this thesis POD modes were extracted not only for snapshots but also for constant spatial planes, therefore a new instantaneous frame called “pseudo-snapshot” was introduced in POD. The modes were smoothed by Kriging interpolation prior to the reconstruction of POD. It was found out that these smoothed modes have significantly increased the accuracy of results.

Kriging interpolation is also investigated in detail. The investigations have shown that the selection of model variogram and its correlation parameter are important for Kriging predictions. The model selection needs experience and well understanding on physical manner of the data. The model variogram and its correlation parameter should be selected based on the aim of the computations. The level of smoothing can be controlled by the correlation parameter. In order to construct a faster and alternative procedure we propose a new line Kriging that uses one-dimensional correlation instead of two-dimensional or three-dimensional correlation. This version of Kriging works computationally much faster and the level of smoothing can be increased. The idea behind the line Kriging was the way of the implementation of digital filters. Digital filters such as Butterworth type filter were also applied for data smoothing purposes.

After smoothed and enhanced the resolution of the PIV data by Kriging interpolation or POD “interpolation”, third component of the velocity which was missing in the PIV data set was calculated from divergence free continuity equation. The results have shown that these data enhancement/smoothing procedures are very important for such a calculation.

The capability of Kriging interpolation to reconstruct the black zones was investigated and it has shown that Kriging can be used effectively for black zones and accuracy of reconstruction depends on if data noisy or smooth.

We also investigated to use Kriging in optimization algorithms. For this aim, a meta-algorithm was built with a combination of Kriging and Simulated Annealing algorithms. It is concluded that this meta-algorithm can be used to find global minimum/maximum of a data set.

For future work, the following issues might be interesting to investigate: currently there is no guarantee that the reconstructed flow in the missing region satisfies the conservation equations of the flow. However, in a future work, it may be possible to implement a Kriging interpolation in an iterative scheme such that, for example, the recovered missing region satisfies the continuity equation. Fast and accurate optimization algorithms might be developed based on the Kriging method.

REFERENCES

- Berkooz, G., Holmes, P., Lumley, J.L., (1993),** “The proper orthogonal decomposition in the analysis of turbulent flows” *Annual Review of Fluid Mechanics*, 25, 539–575.
- Burgmann S., Schöreder W., and Brücker C., (2006),** “Volumetric measurement of vortical structures in the reattachment region of a laminar separation bubble using stereo scanning PIV” 13th Int Symp on Applications of Laser Techniques to Fluid Mechanics Lisbon, Portugal, 26-29 June.
- Cekli H.E., and Gunes H., (2006),** “Spatial resolution Enhancement and reconstruction of mixed convection data using Kriging method”, ASME International Mechanical Engineering Congress and Exposition, November 5-10, Chicago, Illinois, USA.
- Davis, J.C., (2002),** “Statistics and data analysis in Geology”, New York, J. Wiley.
- Deutsch C.V., and Journel A.G., (1992),** “GSLIB: Geostatistical Software Library and User's Guide”, New York, Oxford University Press.
- Everson, R. and Sirovich, L., (1995),** “The Karhunen-Loeve transform of incomplete data” *J. Optical Soc. Of America*, A vol. 12 (8).
- Golden Software, Inc.,** “Variogram Tutorial”.
- Gunes, H., Sirisup, S., and Karniadakis, G.E., (2006),** “Gappy data: To Krig or not to Krig”, *Journal of Computational Physics*, Vol. 212, Issue 1, pp. 358-382.
- Isaaks E.H., and Srivastava R.M., (1989),** “Introduction to Applied Geostatistics”, New York, Oxford University Press.
- Kahler, C.J., (2003),** “Experimental investigation of flows at low Reynolds number using optical techniques”, Braunschweig Technical University, RTO / AVT-104 Vki Lecture Series.
- Kirkpatrick, S., Gelatt, C.D., and Vecchi, M.P., (1983),** “Optimization by Simulated Annealing” *Science*, May 1983, Volume 220, Number 4598.
- Krige, D.G., (1951),** "A statistical approach to some basic mine valuation problems on the Witwatersrand". *J. of the Chem., Metal. and Mining Soc. of South Africa* 52 (6): 119-139.

- Linnick, M.N, and Rist, U. (2004)**, “Detection and visualization of structures in transitional fluid flows”, Internal report, IAG, Stuttgart University.
- Lumley, J. L., (1967)**, “The structure of inhomogeneous turbulence. In Atmospheric Turbulence and Wave Propagation” A. M. Yaglom, V. I. Tatarski, 166-178. Moscow.
- Matheron, G., (1962)**, “Traité de Géostatistique appliquée”, Tome 1, Editions Technip, Paris, 334 p.
- McKay, M.D., Conover, W.J., and Beckman, R.J., (1979)**, “A comparison of three methods for selecting values of input variables in the analysis of output from a computer code”, Technometrics, vol. 21, no. 2.
- Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., and Teller, E., (1953)**, “Equation of state calculations by fast computing machines”, J. of Chemical Physics, Volume 21, Number 6, June.
- Mitra, S.K., (2001)**, “Digital Signal Processing”, 2nd ed., McGraw-Hill.
- Oppenheim, A. V. and Schafer, R.W., (1989)**, “Discrete-Time Signal Processing”, Englewood Cliffs, NJ: Prentice-Hall.
- Piskin S., and Gunes H., (2005)**, “Kriging reconstruction of gappy data for twin-jet flow”, International Conference on Computational Science and Engineering, Istanbul.
- Raffel, M., Willert, C.E., and Kompenhans, J., (1998)**, “Particle Image Velocimetry – A practical guide”, Springer Verlag.
- Sirisup, S. (2004)**, “Issues in Low-Dimensional Modeling of Unsteady Flows: Convergence, Asymptotic Stability and Reconstruction Procedures”, Ph.D. Thesis. Brown University.
- Sirovich, L., (1987)**, “Turbulence and the dynamics of coherent structures” I-III. Q. Appl. Math. 45, 561.
- Venturi, D., and Karniadakis G.E., (2004)**, “Gappy data and reconstruction procedures for flow past a cylinder”, J. Fluid Mech. 519 315–336.
- <http://www2.itu.edu.tr/~ceklil/research.htm> , (December 2006),

APPENDIX

Numerical schemes:

3rd order numerical derivation and integration schemes are used to calculate the cross-flow velocity.

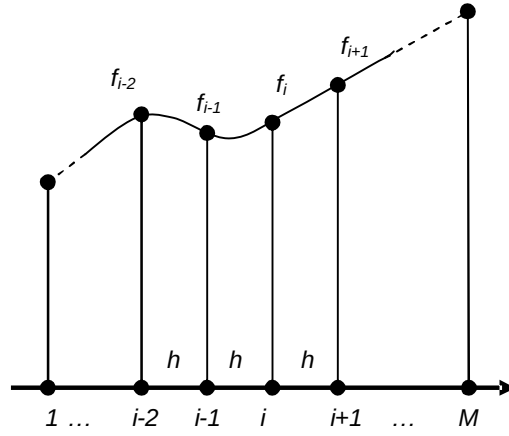


Figure A.1: Stencil for numerical derivation and integration.

Numerical derivation:

$$\left. \frac{\partial f}{\partial x} \right|_i = \frac{1}{6h} (f_{i-2} - 6f_{i-1} + 3f_i + 2f_{i+1}) \quad (\text{A.1})$$

$i = 3, \dots, M-1$

$$\left. \frac{\partial f}{\partial x} \right|_1 = \frac{1}{6h} (-11f_1 + 18f_2 - 9f_3 + 2f_4) \quad (\text{A.2})$$

$$\left. \frac{\partial f}{\partial x} \right|_n = \frac{1}{6h} (11f_n - 18f_{n-1} + 9f_{n-2} - 2f_{n-3}) \quad (\text{A.3})$$

Numerical integration:

$$y_i = y_{i-1} - \frac{h}{24}\{f_{i-2} + f_{i+1}\} + \frac{13h}{24}\{f_{i-1} + f_i\}$$

$$i = 3, \dots, M-1 \quad (\text{A.4})$$

$$y_1 = a$$

$$y_2 = y_1 + \frac{h}{24}\{9f_1 + 19f_2 - 5f_3 + f_4\} \quad (\text{A.5})$$

$$y_M = f_{M-2} + \frac{h}{3}\{f_{M-2} + 4f_{M-1} + f_M\} \quad (\text{A.6})$$

Validation of numerical schemes:

The accuracy of numerical schemes was investigated on a test data. The test was created by Eq.A.7 for 21x21 (low-resolved) and 41x41 (high-resolved) grid numbers in x and y -directions, respectively. Actual data was derived numerically by the above scheme, and the derived data was integrated by the aforementioned numerical integration. If there would not be any numerical calculation errors we would have obtain identically same actual data. To check accuracy of our numerical schemes we plot the variable contours of the final integrated data on the top of the variable the contours of the actual data for low resolved and high resolved data sets in Figure A.2 a – b, respectively. When these figures are compared to each other it can be said that the numerical derivation and integration schemes have a good accuracy and resolution of the data set is very important for such computations.

$$u(x, y) = (x^5 - y^3 + y^2)e^{-(x^2+y^2)} \quad (\text{A.7})$$

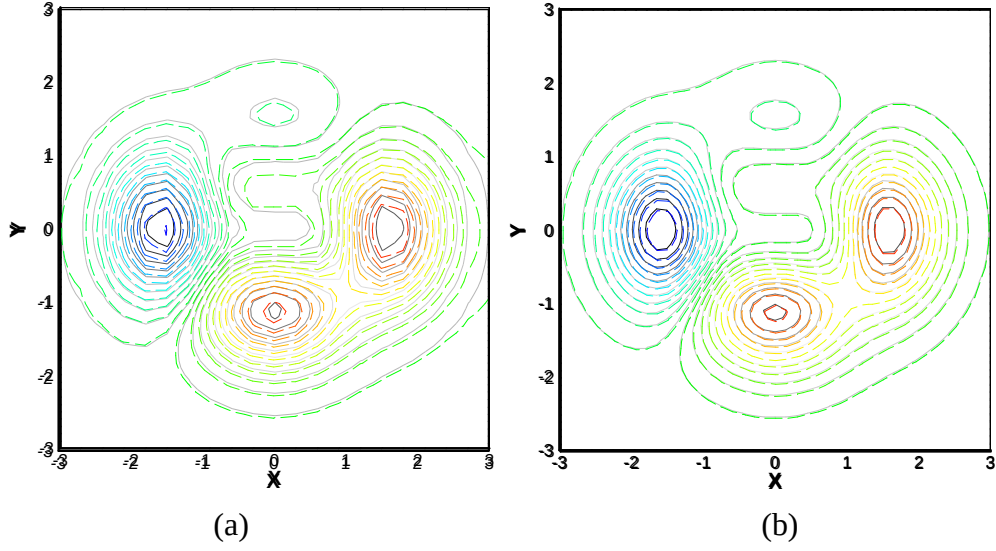


Figure A.2: The contours of the actual and the final integrated data, continuous line denotes the actual data and dashed line denotes the final integrated data, (a) low-resolution, (21x21 grid points), (b) high-resolution, (41x41 grid points).

2D x-y graphs are given for x versus variable and its derivative on a constant line $y = -0.6$ for low- and high-resolved data in Figure A.3a-b, respectively.

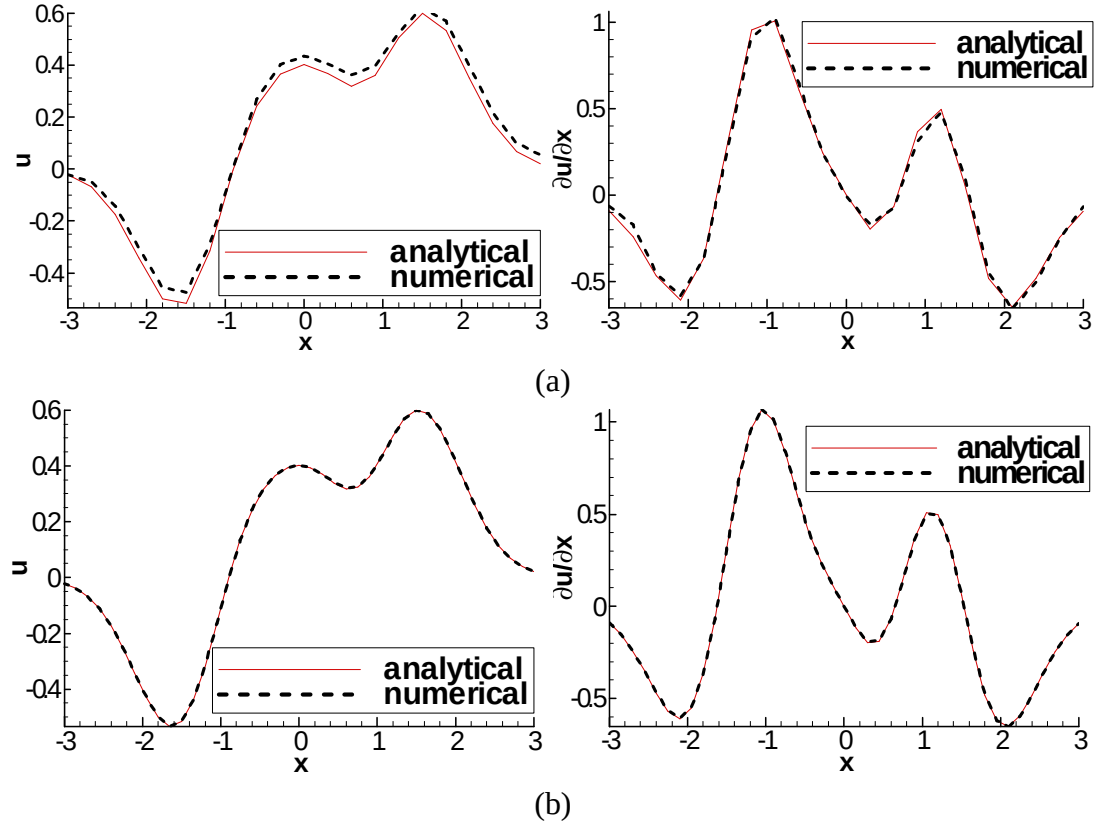


Figure A.3: Analytical and numerical calculated values of variable and its derivative on constant line $y = -0.6$, (a) low-resolution, (21x21 grid points), (b) high-resolution, (41x41 grid points). left: variable, right: derivative.

Smoothing data via different variogram models: Application to Poisson Equation

The solution of Poisson equation Eq.A.8 was used as a test data that actual solution is known. The data was created by solving the poisson equation with following boundary conditions in intervals (0,2) and (0,1) in x and y -directions, respectively. The equation was disturbed by adding random numbers to values at grid points. The task is smooth the disturbed data by Kriging with different variogram models.

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = (x^2 + y^2)e^{xy} \quad (\text{A.8})$$

$$\text{Boundary conditions: } u(0,y) = 1, u(2,y) = e^{2y}, u(x,0) = 1, u(x,1) = e^x$$

Because of actual data is known root mean square (*rms*) error values can be calculated for disturbed and smoothed data. First, a disturbed data was created for 11.09% *rms* value. The actual, disturbed and smoothed data with different variogram models and correlation parameters are given in Figures 7-10, by the contours of variables. Mathematical functions of these variogram models, the Gaussian, sinus1 and sinus2, are given by Eq.A.9-11, respectively.

$$\gamma(h) = 1 - \exp(-\theta h^2) \quad (\text{A.9})$$

$$\gamma(h) = \theta h \sin(h) \quad (\text{A.10})$$

$$\gamma(h) = \theta \left(1 - \frac{\sin(h)}{h} \right) \quad (\text{A.11})$$

Plots of these variogram models are given in Figure A.4-6.

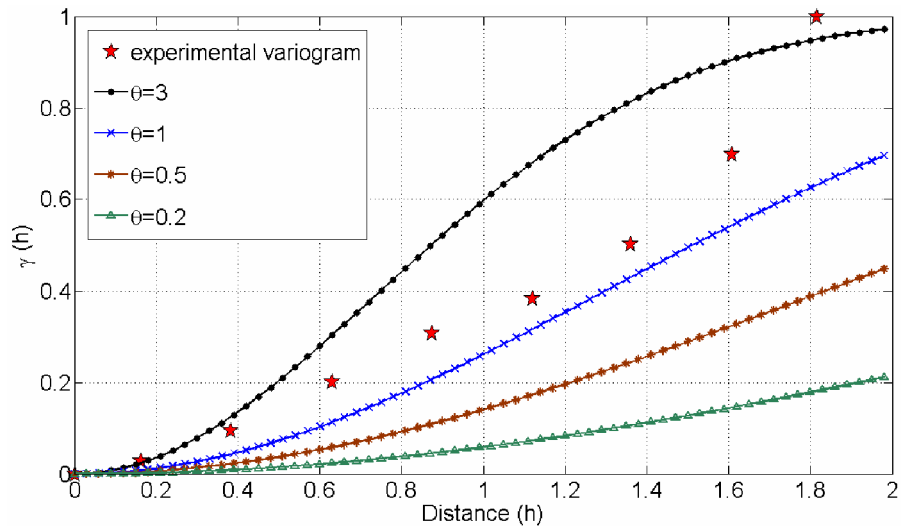


Figure A.4: The Gaussian variogram model for different correlation parameters.

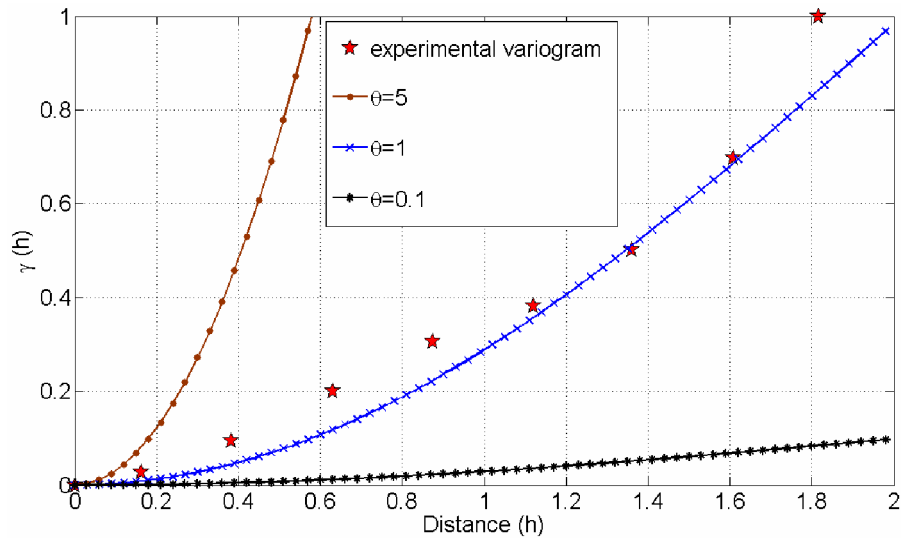


Figure A.5: Sinus1 variogram model for different correlation parameters.

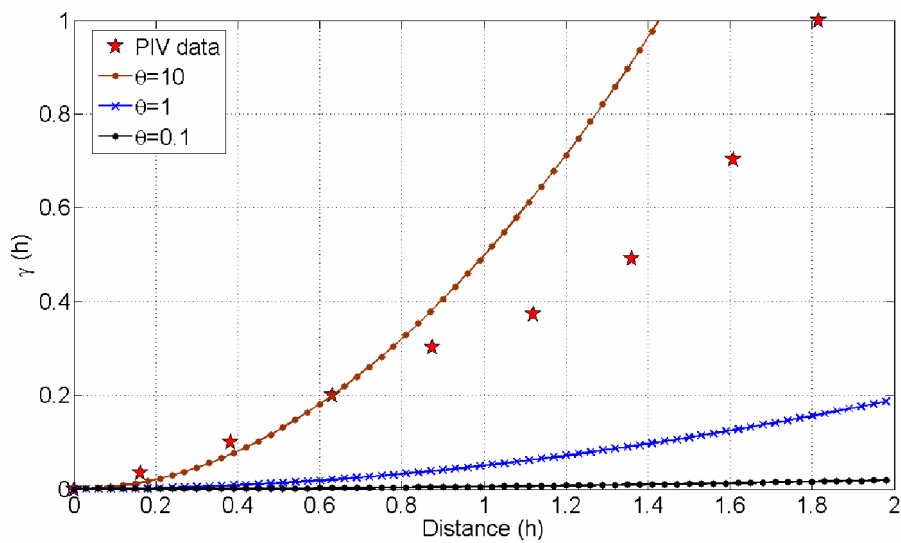


Figure A.6: Sinus2 variogram model for different correlation parameters.

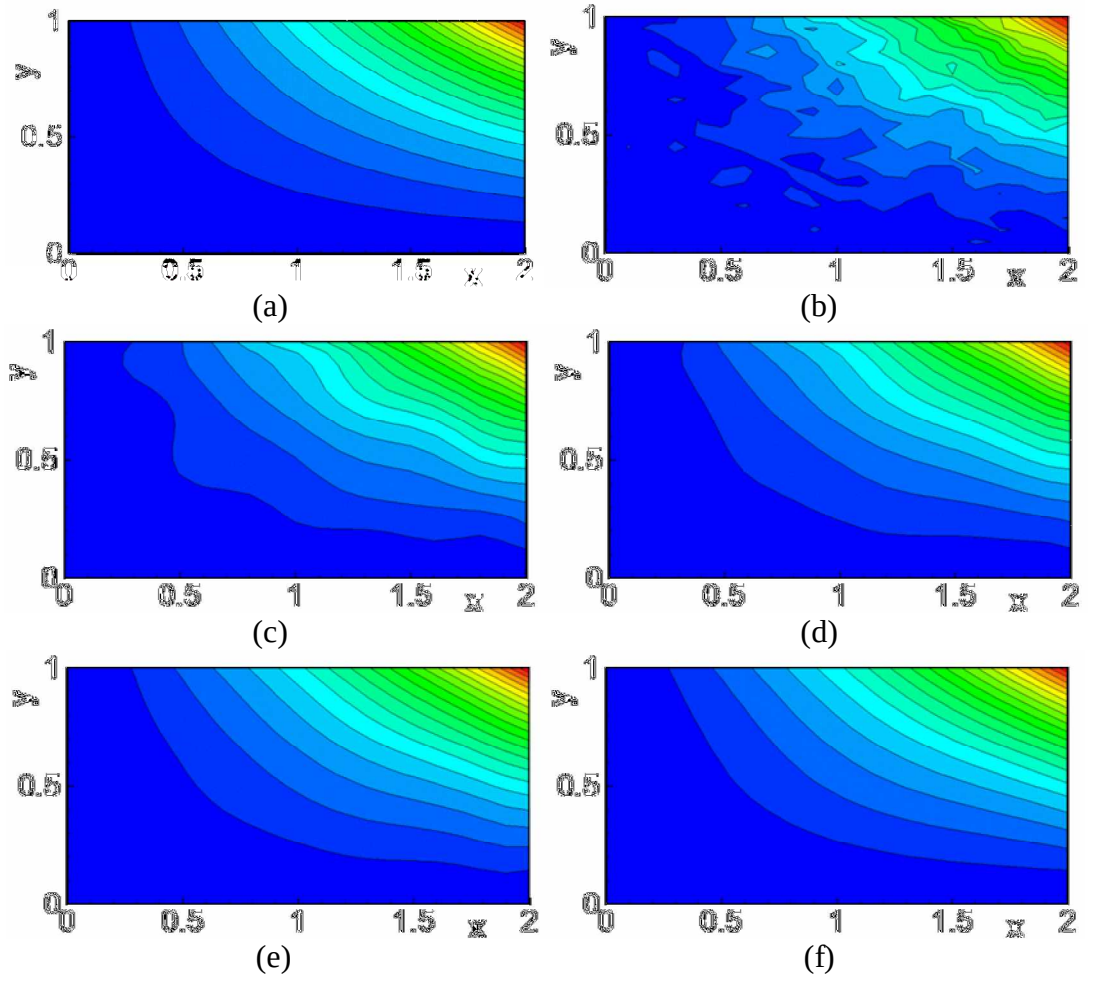


Figure A.7: The contours of the variable of the actual, disturbed and smoothed data with the Gaussian model for different correlation parameter, (a) actual data, (b) disturbed data, ($rms = 11.09\%$), (c) $\theta = 3$, ($rms = 4.24\%$), (d) $\theta = 1$, ($rms = 3.46\%$), (e) $\theta = 0.5$, ($rms = 2.87\%$), (f) $\theta = 0.2$, ($rms = 2.49\%$).

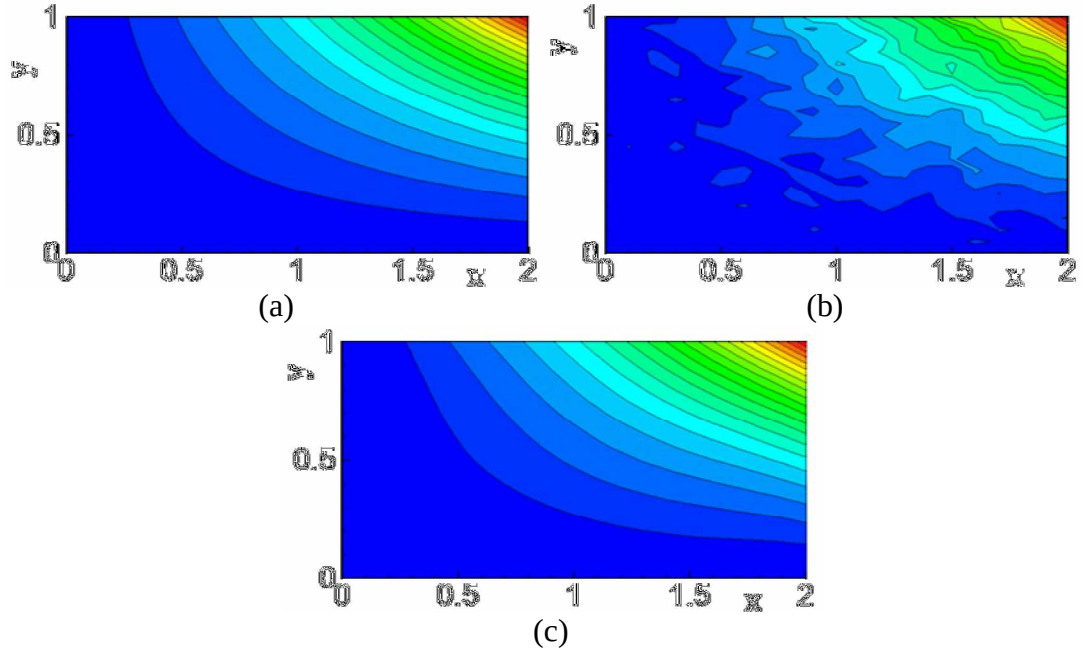


Figure A.8: The contours of the variable of the actual, disturbed and smoothed data with the sinus1 model for different correlation parameter, (a) actual data, (b) disturbed data, ($rms = 11.09\%$), (c) $\theta = 0.1$ ($rms = 2.44\%$).

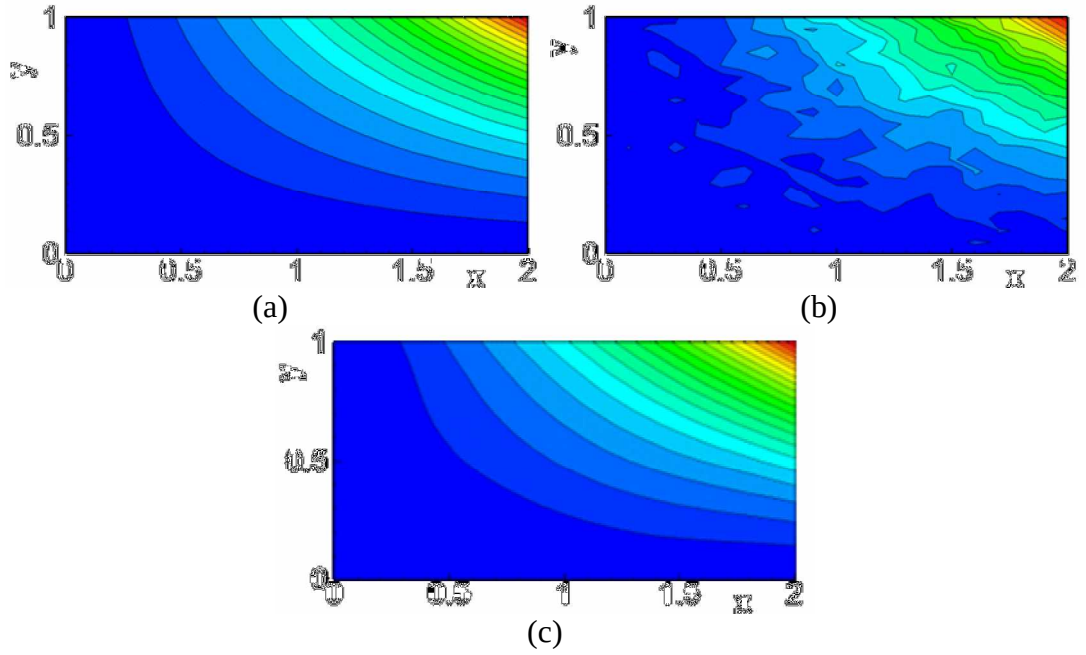


Figure A.9: The contours of the variable of the actual, disturbed and smoothed data with the sinus2 model for different correlation parameter, (a) actual data, (b) disturbed data, ($rms = 11.09\%$), (c) $\theta = 0.1$, ($rms = 2.2\%$).

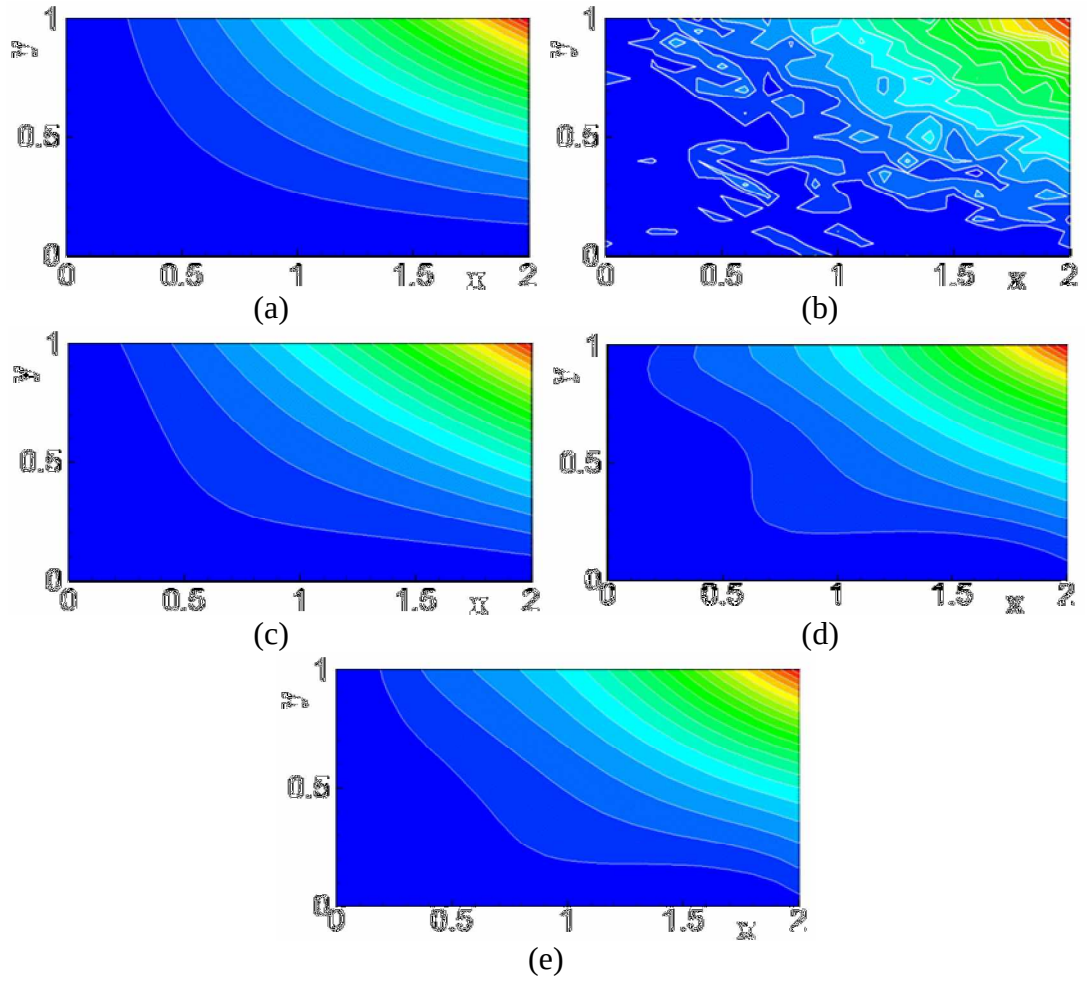


Figure A.10: The contours of the variable of the actual, disturbed and smoothed data with different variogram models for higher *rms* value, (a) actual data, (b) disturbed data (*rms* = 18.4%), (c) The Gaussian model, $\theta = 0.01$, (*rms* = 4.48%), (d) sinus1 model, $\theta = 1$, (*rms* = 6.03%), (e) sinus2 model, $\theta = 1$ (*rms* = 5.297%).

BIOGRAPHY

Hakki Ergun Cekli was born in Bremen, Germany, in 1980. He completed his high school education at Kemal Hasoglu High School, Istanbul, in 1997, and received his B.Sc. degree in mechanical engineering from Yildiz Technical University in 2003. Currently, he is a graduate student at Thermo – Fluids M.Sc program at the Institute of Science of Istanbul Technical University.

Hakki Ergun Cekli has been working as a research assistant at Istanbul Technical University, Faculty of Mechanical Engineering, Hydromechanics and Hydraulic Machinery Division.