UNIVERSITY OF GRANADA

FACULTY OF SCIENCES

Evolutionary computation of optimal knots allocation in smoothing methods by multivariate splines and radial function basis spaces.

by

Hasan M.H Idais.

A dissertation presented in fulfilment of the requirements for the degree of Doctor of Applied Mathematics

Doctoral Programme of Mathematics

Granada, Spain

2019

Editor: Universidad de Granada. Tesis Doctorales
Autor: Hasan M. H. Idais
ISBN: 978-84-1306-280-8
URI: http://hdl.handle.net/10481/56573

Supervised By

1. Prof. Dr. Miguel Pasadas Fernández (University of Granada, Full Professor of Applied Mathematics).
2. Prof. Dr. Pedro González Rodelas (University of Granada, Associate Professor of Applied Mathematics).

Dissertation Committee:

1. President: Victoriano Ramírez González
2. Secretary: Domingo Barrera Rosillo
3. Vocal 1: Chelo Ferreira González
4. Vocal 2: Francisco Javier Muñoz Delgado
5. Vocal 3: Mari Cruz López de Silanes
6. First Stand-in: Daniel Cárdenas Morales
7. Second Stand-in: María José Ibáñez Pérez

Acknowledgments

Undertaking this Ph. D. has been a truly life-changing experience for me and it would not have been possible to do without the support and guidance that I received from many people.
Foremost, I would like to express my sincere gratitude to my advisors Prof. Dr. Miguel Pasadas Fernández and Prof. Dr. Pedro González Rodelas for the continuous support of my Ph.D study and research, for their patience, motivation, enthusiasm, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisors and mentors for my PhD study.
Last but not the least, I would like to thank my family: my mother, brothers and I would thank my wife "Dunia" , for endless support and for her standing next to me in most hard times where I couldn't stand.

Contents

List of Figures

List of Tables

Resumen en español

Esta tesis doctoral está estructurada en cuatro capítulos, ya que este primero no es más que un amplio resumen de la misma en castellano. El resto de capítulos han sido redactados directamente en inglés, que es el idioma en el que se han confeccionado los artículos extraidos de este trabajo para ser publicados en revistas internacionales (incluidas en el listado JCR), aparte de ser el idioma vehicular habitual en el que tanto el doctorando como los directores de tesis se han podido comunicar más fluidamente, la mayor parte del tiempo.

Los artículos referidos anteriormente son:

- González, P., Idais, H., Pasadas, M., Yasin, M., Evolutionary computation for optimal knots allocation in smoothing splines of one or two variables, International Journal of Computational Intelligence Systems 11, 2018, 1294-1306. DOI: 10.2991/ijcis.11.1.96

- Idais, H., Yasin, M., Pasadas, M., González, P., Optimal knots allocation in the cubic and bicubic spline interpolation problem, Mathematics and Computers in Simulation, to appear,
  DOI: 10.1016/j.matcom.2018.11.002

Capítulo 1

El Capítulo 1 constituye una presentación en inglés de la tesis y una pequeña introducción a los conceptos fundamentales para seguir el resto del trabajo de investigación realizado. Así pues el objetivo primordial del mismo es presentar las principales definiciones y conceptos preliminares necesarios para las técnicas de interpolación o aproximación empleadas y que harán uso de B-splines (Subsección 1.5.2) o bien funciones radiales (Subsección 1.6), junto con un breve repaso de ciertas cuestiones teóricas generales requeridas en nuestra investigación.

En la Sección 1.4 se recuerda brevemente la teoría y propiedades fundamentales de los espacios de Hilbert y Sobolev, que son los espacios por antonomasia donde las técnicas generales del Análisis Funcional pueden aplicarse con mayor eficacia. Se introduce en las Secciones 1.3 y 1.5 la notación usada habitualmente para las bases de B-splines y se presentan sus propiedades fundamentales, así como una forma recursiva para su cálculo en la Definición 1.5.1 y las gráficas 1.5.2 y 1.5.3 de alguna de ellas.

También en la Tabla 1.6 se muestran algunos de los tipos de funciones radiales más extendidos y usados en la práctica, junto con los parámetros de los que dependen y a continuación se recuerdan alguno de los resultados fundamentales (consultar por ejemplo [26], [16], [48], [50], [8], [4]) relacionados con este tipo de funciones de cara a la aproximación o interpolación con este tipo de funciones, siendo también de especial importancia las de soporte compacto, entre las que se encuentran las denominadas funciones radiales de Wendland (consultar la Subsección 1.6.2).

En la siguiente Sección 1.7 son presentados los denominados *Splines Variacionales*, recordando la gran utilidad y cantidad de aplicaciones que tiene este punto de vista, tanto para la interpretación física como para la aproximación numérica, de innumerables problemas científico-técnicos.

Más concretamente, tanto los denominados *splines de suavizado* (Subsección 1.7.1) como los *splines de interpolación variacionales* (Subsección 1.7.2) son introducidos, usando fundamentalmente las referencias [36] y [39].

Finalmente, en la Sección 1.8 se presentan las definiciones básicas y se desarrollan las principales ideas relacionadas con los problemas de optimización multi-objetivo, que son de una importancia primordial en muchas áreas Cientifico-Técnicas: en Economía, Transporte, Logística, etc. También se presentan de una manera somera y muy simplificada los denominados algoritmos genéticos de ordenación no-dominada (NSGA, del inglés "Non-dominated Sorting Genetic Algorithms") que se engloban en los archiconocidos algoritmos genéticos (GA del inglés) cuyo objetivo consiste en todo caso en la adaptación evolutiva, a lo largo de sucesivas generaciones que se van sucediendo de "padres" a "hijos" en un posible conjunto de soluciones o "población", a través de ciertos genes y/o mutaciones genéticas que favorecen una mejor valoración de cara a ciertas funciones objetivo que se necesita optimizar. El problema surge cuando en muchas ocasiones estas funciones objetivo no pueden optimizarse todas a la vez sin entrar en conflicto, de manera que algunas de ellas resultan antagónicas, ya sea totalmente o en parte, de manera que hay que buscar entre un amplio conjunto de posibles "buenas" soluciones, aquella o aquellas que de alguna manera resultan más favorables en su conjunto. Surge aquí el conocido concepto del frente (o frentes en el caso más general) de Pareto, que son la frontera geométrica de cierto con-

junto de soluciones factibles, que de alguna manera optimizan alguna de las componentes de la función objetivo, aunque no todas a la vez.

De hecho surge aquí de manera natural también el concepto de solución no-dominada, en el sentido de que no existe ninguna otra posible solución que mejore globalmente todas las componentes de la función objetivo, y justo en esta idea básica se basan los denominados algoritmos genéticos con ordenamiento no-dominado (NSGA del inglés Non-dominated Sorting Genetic Algorithm). Consultar la Subsección 1.8.2 para una descripción más detallada de la idea básica de este tipo de algoritmos, así como de las variables y parámetros que son necesarios incluir y definir de cara a su aplicación práctica. Cabe destacar por ejemplo la inicialización de la población, con un tamaño adecuado, así como la adecuada elección de otros muchos parámetros como la denominada "distancia de acumulación o apiñamiento" ("crowding distance" en inglés) que dependerá en gran medida de las características del problema a resolver.

Por otro lado, como en todo algoritmo genético, también habrá que tener en cuenta ciertos operadores de selección, de mutación y de recombinación que se usarán para ir generando las sucesivas generaciones de individuos que conformarán las poblaciones de posibles soluciones a estimar en cada momento y que si todo va bien irán llevándonos a alguno de los valores óptimos de nuestro problema. Precisamente, con el objetivo de evitar caer en alguno de los posibles óptimos locales de ciertos problemas de este tipo, la denominada "fracción de Pareto" permitirá preservar en nuestra población ciertos individuos que aunque en un principio pudieran parecer dar lugar a una evaluación peor de la función objetivo, sí que promueven cierta diversidad en esta población, que a la larga redundará en una mayor capacidad para salir de estos óptimos locales, con vistas a conseguir aproximar mejor el buscado óptimo (ya sea máximo o mínimo) global (consultar la Sección 2.3 para ver cómo se ha llevado a cabo todo esta implementación en el caso de la optimización de la localización de los nodos para los splines de suavizado de tipo cúbico o bicúbico o bien la Sección 4.1 para el problema de interpolación con splines naturales en una o dos variables).

Capítulo 2

En el Capítulo 2 empezamos ya planteando y estudiando el interesante problema de la posible distribución óptima a la hora de colocar los nodos usados en la construcción de B-splines de suavizado para la aproximación de ciertos datos, tanto de una (para datos 2D) como de dos variables (para datos 3D),

mediante la aplicación de un algorítmo de tipo NSGA-II (Non Dominated Sorting Genetic Algorithm, 2nd. version) adaptado específicamente para este propósito (consultar la Subsección 1.8.2 y la Sección 2.3).

De hecho la aproximación de funciones, así como el ajuste de datos mediante curvas o superficies es uno de los problemas fundamentales de la Matemática Aplicada y el Cálculo Científico en general, con aplicaciones en numerosísimas ramas de la Ciencia y la Técnica: como el diseño asistido por ordenador (CAD, del inglés "Computed Aided Design"), la realidad virtual y la animación por computadora, la visión artificial y la visualización automatizada de todo tipo de imágnes médicas, geológicas, etc. (consultar por ejemplo [54], [76]).

Por otra parte, existen diferentes estrategias y metodologías que se han usado para la adecuada selección de los nodos cuando se busca la mejor aproximación o el ajuste de cierto conjunto de datos 2D o 3D mediante curvas o superficies usando B-splines. Por ejemplo, la methodología empleada en [42] usa un método de ajuste por mínimos cuadrados con superficies de tipo B-spline, que dependen de una parametrización sensible a dicho ajuste, a partir de una distribución uniforme de los nodos. En las últimas décadas, un número creciente de trabajos y algoritmos han sido desarrollados para este tipo de problemas. Por ejemplo, los autores en [72] desarrollaron un algoritmo genético para optimizar tanto el número como la localización de los correspondientes nodos en la aproximación mediante splines de ciertos datos de tipo bio-médico. Posteriormente, un nuevo enfoque fue introducido para mejorar el ajuste de dichos nodos para curvas de tipo B-spline, usando un algoritmo apropiado de selección clonal elitista [24] que mejoraba sensiblemente los resultados obtenidos. La metodología en [82] describe más bien un algoritmo iterativo, que también puede usar ciertos criterios de tipo heurístico para la distribución óptima de los nodos en el ajuste de superficies usando B-splines de dos variables. Kozera et al. en [40] también proporcionan un nuevo método para elegir los nodos de manera que se minimice la función coste usando cierta escala especial.

En todo caso, el ajuste de datos 3D o de superficies puede involucrar diferentes tipos de splines; por ejemplo, algunas técnicas sin malla para la grabación de imágenes médicas mediante la modificación de ciertos nodos seleccionados en los denominados P-splines [56]. Otro método en [77] describe la técnica de *promediado unificado* relacionada con el emplazamiento de los nodos en curvas de tipo B-spline para problemas tanto de aproximación como de interpolación.

En [25] se presenta un método para resolver el problema de ajuste de curvas, usando un algoritmo genético de tipo jerárquico (HGA, del inglés Hierarchical Genetic Algorithm). Otras metodologías computacionales basadas

en una especie de "Sistema Inmune Artificial" (AIS, del inglés Artificial Immune Systems) se han usado también para resolver este problema de ajuste de nodos en curvas de tipo B-spline. El objetivo de la aplicación de este algoritmo es determinar la localización de dichos nodos de manera automática y eficiente.

En [73] los autores describen una poderosa metodología, basada en una técnica probabilística para aproximar el óptimo global de una función mediante una especie de "enfriamiento simulado" (SA, del inglés Simulated Annealing) para el emplazamiento óptimo de los nodos. De hecho, en este tipo de técnicas metaheurísticas, el empezar aceptando de partida ciertas soluciones no tan buenas de antemano resulta a la larga fundamental, ya que permite al final ampliar mucho más el espacio de búsqueda de las posibles soluciones de cara a poder alcanzar con mayor eficacia y seguridad el óptimo global, en caso de que este exista.

Otros autores en [35] usan ciertas técnicas de optimización dispersa para este propósito. Sin embargo, como también se pone de manifiesto en [71] y en muchos otros trabajos relacionados, una metodología final y definitiva para el problema considerado del emplazamiento óptimo o de selección de los nodos, para la aproximación o el ajuste de curvas o superficies usando splines de suavizado no está todavía completamente investigado, ni tan siquiera en el caso unidimensional.

En este sentido, lo que nosotros proponemos es una nueva metodología para el emplazamiento óptimo de nodos que se pueden mover aleatoriamente en un cierto rango o intervalo de cada una de las variables involucradas, y que permiten aproximar o ajustar curvas o superficies a los datos 2D/3D, usando splines de suavizado de una o dos variables independientes. Esta nueva técnica, basada en el desarrollo y adaptación a este tipo de problemas de un algoritmo genético multi-objetivo con ordenamiento no-dominado, de tipo NSGA-II (consultar la Subsección 2.3). Revisando los ejemplos y experimentos realizados en la Sección 2.4 podemos comprobar de hecho que este planteamiento proporciona resultados bastante precisos, incluso para curvas o superficies con discontinuidades o picos pronunciados, determinando también de manera automática tanto el número de nodos necesarios como su emplazamiento óptimo para una adecuado ajuste en el caso de B-splines cúbicos (en el caso de ajuste de curvas en la Subsección 2.4.1) o bicúbicos (en el caso de superficies en la Subsección 2.4.2).

Las conclusiones finales son presentadas en la última Sección 2.5, concluyendo en todo caso el buen comportamiento de la estrategia MOGA aplicada a este problema de emplazamiento óptimo de nodos para la aproximación de datos 2D/3D mediante splines cúbicos/bicúbicos, ayudando en todo caso el análisis del frente de Pareto para una elección adecuada también

del número de nodos a emplear, aunque como regla general el incremento de este número de nodos suele incrementar la precisión de la aproximación obtenida, pero sólo hasta cierto punto, si tuviéramos también en cuenta el esfuerzo computacional añadido. Lo que sí que se concluye de manera contundente es que nuestra estrategia proporciona resultados mucho mejores que otros más clásicos existentes en la literatura científica relacionada, y basados en la mera selección de nodos a partir de una distribuión uniforme inicial.

Capítulo 3

En el Capítulo 3, se trata ahora de la posible obtención de la distribución óptima de nodos, tanto de una (para datos 2D) como de dos variables (para datos 3D), pero esta vez para un problema de tipo interpolatorio, mediante splines cúbicos o bi-cúbicos naturales (con condiciones de derivadas segundas nulas en la frontera del dominio). Este caso concreto de interpolación puede resultar de especial interés o incluso ser la clave de muchas tecnologías o procedimientos de ingeniería inversa, tratamiento de señales, el análisis sistemático de datos o funciones, o incluso la simple representación, almacenaje o compresión de los mismos.

Y lo mismo que ocurría en el problema de la aproximación, aquí también varios autores han planteado diferentes métodos y estrategias para la interpolación de ciertos datos bidimensionales o tridimensionales, como por ejemplo [42, 45, 79] y todos ellos manifiestan la importancia y el efecto determinante del emplazamiento de los nodos en la precisión y efectividad del procedimiento empleado para la obtención de los resultados finales, tanto en el caso del problema de aproximación como el de interpolación.

De hecho bastantes metodologías, al menos para funciones de una variable, han sido presentadas para la selección y optimización de parámetros para el cálculo de los correspondientes B-splines, usando técnicas basadas en la selección de ciertos nodos especiales, denominados puntos dominantes [51, 52], o mediante un proceso de selección de nodos, como en [53]. También en [24] se presenta una tecnica para la modificación automática de nodos usando un algoritmo de selección clonal elitista. Otros procedimientos usan ciertas técnicas de mínimos cuadrados a partir de distribuciones uniformes de nodos, en conexión con una parametrización sensible a posibles desviaciones de un resultado óptimo [42]. En este sentido merece la pena tener en cuenta los siguientes trabajos [45] y [79]. En [40] los autores presentan un método para seleccionar estos nodos desconocidos mediante la minimización de cierta función de coste (o función objetivo), que a la postre resulta ser

uno de los ingredientes fundamentales en la implementación de los denominados algoritmos genéticos (GA, del inglés Genetic Algorithms). Otras estrategias, como por ejemplo la presentada en [25], describe técnicas que emplean métodos computacionales que simulan cierto sistema inmune artificial (AIS, del inglés Artificial Immune System), basados en cierta estructura jerárquica para determinar tanto el número de nodos como la localización de los mismos automáticamente. Estos procedimientos computacionales adaptativos de tipo AIS están inspirados en teorías y modelos de sistemas inmunológicos biológicos reales, pero adaptados para su resolución de problemas matemáticos o de ingenieria.

Siguiendo las ideas de Man et al. en [49] la representación computacional elegida para almacenar y manipular los correspondientes "cromosomas" en cualquier GA también podría gestionarse utilizando una estructura jerárquica más compleja, dando lugar a los denominados algoritmos genéticos jerárquicos (HGAs, del inglés Hierarchical Genetic Algorithms), que emularía la formación del DNA en la Biología de los seres vivos. Nosotros hemos descartado esta opción, ya que hemos considerado que dicha complicación no aportaba ventajas evidentes para el tipo de problemas tratados en esta memoria de tesis.

Otros autores en [31], [35] plantearon enfoques diferentes para la optimización del número de nodos y su localización definitiva en el proceso de ajuste de curvas mediante splines usando un modelo de optimización dispersa; a su vez en [74], un procedimiento bastante rápido y eficiente computacionalmente para el emplazamiento óptimo de nodos usando una técnica de "enfriamiento simulado" (SA, de Simulation Annealing en inglés) también es presentado. En este procedimiento estocástico, una combinación de "paso de montaña" (hill-climbing en inglés) y "avance aleatorio" (random walks en inglés) es llevado a cabo para simular un proceso físico bastante conocido en el templado metalúrgico. Así pues SA es un método de descenso modificado por ciertos movimientos ascendentes de tipo aleatorio, con el objeto de intentar escapar de cualquier mínimo local que no sea global.

En [17] y las múltiples referencias allí indicadas se pueden consultar muchos más detalles sobre las ideas que sostienen los procedimientos para resolver problemas de optimización multi-objetivo, así como los aspectos fundamentales de los GAs en general y del algoritmo NSGA-II genérico en particular.

De hecho, los primeros GAs fueron empleados fundamentalmente optimizar una sola función de coste (eran problemas de tipo mono-objetivo); pero pronto surgió la necesidad imperiosa de resolver también problemas con varias o muchas funciones coste (los denominados problemas de optimización multi-objetivo), dada la gran diversidad de problemas de la vida

real en esta situación. De hecho, el primer GA multi-objetivo fue propuesto por Schaffer [67] y lo denominó VEGA (del inglés "Vector Evaluated Genetic Algorithm"). A continuación, varios Algoritmos Genéticos Multi-Objetivo (MOGA, del inglés Multi-Objective Genetic Algorithm) fueron desarrollados (ver por ejemplo [22], Niched Evolutionary Algorithms [41], Random Weighted Genetic Algorithms (RWGA) [78] y muchos otros, que pueden ser consultados en [15].

Pero una de las estrategias más fiables y exitosas en este sentido ha sido la aplicación de la noción que el autor Goldberg denominó *ordenamiento no-dominado* en el GA junto con cierto anidamiento para poder localizar puntos que pertenezcan a frentes de Pareto múltiples simultáneamente [66]. Pero las mayores extensiones y mejoras de toda esta estrategia se ha conseguido con las más recientes versiones de tipo NSGA-II y NSGA-III (consultar [33] para una comparativa entre estas dos versiones), siendo el punto clave de estas dos nuevas versiones respecto al procedimento NSGA original, las siguientes:

- Un procedimiento rápido de *ordenamiento no dominante* en el que la población es ordenada siguiendo una jerarquía de subpoblaciones basadas en cierta dominancia según el frente de Pareto correspondiente. Cualquier similitud entre los correspondientes miembros es pues evaluada para cada uno de estos subgrupos y ciertas medidas de similitud serán empleadas para resaltar en un frente preferente aquellos individuos que son soluciones no dominadas (según el pseudo-orden impuesto por las funciones coste del problema multi-objetivo) por el resto.

- La apropiada consideración de un *enfoque elitista* con el objetivo de mejorar las propiedades de convergencia del algoritmo.

- El empleo de un *operador de nicho*, que mantenga cierto nivel de soluciones multiples o suficientemente diversas.

Más recientemente, un nuevo algorimo de optimización múltiple a través de un enfoque unificado y válido tanto para problemas con una sola o varias funciones objetivo ha sido propuesto y analizado en [64]. En esta versión unificada de estos algoritmos los autores también muestran cómo funciona esta estrategia con los habituales parámetros de Algoritmos Evolutivos (EAs, del inglés Evolutionary Algorithms), de manera que no se necesitaría ningún parámetro ajustable adicional. También se hace hincapié en que ciertos procedimientos MOGA, como los NSGA-II, van muy bien para problemas bi-objetivo pero no son totalmente escalables para resolver eficientemente problemas multi-objetivo, cuando el número de funciones a optimizar a la vez es

elevado. aunque sin embargo funcionan relativamente bien resolviendo problemas mono-objetivo. Por otra parte, para problemas con tres o más funciones objetivo, la versión más reciente NSGA-III del algoritmo es preferida, aunque también se propone un procedimiento unificado que es capaz de manejar eficientemente también el caso multi-objetivo sin necesidad de reimplementar o cambiar el algoritmo de optimización cuando tengamos que tratar con problemas con un número de funciones objetivo diferente del problema original, dotándolo de una gran versatilidad.

En esta tesis, nosotros lo que hemos desarrollado y adaptado para nuestro problema de emplazamiento óptimo de nodos para la interpolación de un conjunto de datos usando splines cúbicos o bicúbicos ha sido una metodología NSGA-II bi-objetivo, empleando como funciones objetivo varias versiones discretas de ciertas normas habituales en este tipo de cuestiones. Pero tenemos que resaltar que, aunque cietos procedimientos evolutivos usando algoritmos genéticos habían sido ya propuestos en la literatura científica sobre emplazamiento óptimo de nodos para el ajuste y aproximación de problemas de una variable independiente, no hemos encontrado referencias equivalentes para el caso interpolatorio, ni tampoco para el caso de funciones de dos variables. De hecho, muchos de los algoritmos genéticos (GAs) encontrados en la literatura especializada se aplicaban fundamentalmente a ciertos planteamientos combinatorios para la selección de nodos entre una cierta partición uniforme del intervalo de partida, mientras que nosotros estamos empleando un algoritmo MOGA de tipo NSGA-II mucho más sofisticado, con una representación mediante números reales de los correspondientes cromosomas, de manera que estos nodos o puntos de interpolación se pueden mover con total libertad a lo largo del intervalo o intervalos considerados en cada una de las variables, o incluso en algunos casos pueden llegar a colisionar y desaparecer, una vez que varios de esos nodos se acerquen entre sí por debajo de cierta distancia umbral. Así pues, esta sería la una de las peculiaridades y versatilidad de nuestro planteamiento, junto con la habilidad de tratar tanto con splines de una o dos variables, para interpolar (o aproximar según sea el caso) datos 2D o 3D.

Capítulo 4

En un capítulo final, se estudia también de forma preliminar el interesante uso de funciones de base radial (abreviadamente RBFs, de su denominacón en inglés: *Radial Basis Functions*), empezando por una de las familias de funciones de este tipo más conocidas y simples de implementar, como son

las RBFs de tipo Gaussiano. En este caso se presenta un amplio y versátil abanico de nuevas posibilidades para la colocación óptima de los centros que se usarán para la construcción de la base de funciones radiales correspondiente, construida a partir de una cierta función generadora concreta. Así pues, según sea dicha función generadora empleada, que además puede depender de uno o varios parámetros adicionales, también podríamos ampliar mucho más el espectro de posibilidades para la obtención óptima de los nodos o centros de las funciones de base a elegir de cara al problema de interpolación o aproximación a resolver, pudiéndose además tratar en este caso de una manera unificada el caso multi-dimensional, sea cual sea la dimensión. No obstante, tratándose esta última parte de un trabajo meramente preliminar, todavía no se ha explorado exhaustivamente todas estas nuevas posibilidades de esta formulación, y quedaría pendiente para una posterior continuación del trabajo de investigación realizado, usando también funciones radiales de soporte compacto (CS-RBFs) e incluyendo también alguno de los parámetros de los que dependen las correspondientes funciones generadoras en el correspondiente proceso de optimización multi-objetivo que tratamos con los algoritmos genéticos de tipo NSGA-II y -III empleados.

Chapter 1

Introduction and Preliminaries.

1.1 Thesis Presentation

This doctoral thesis is structured in four main Chapters, with an extense Spanish summary in a previous one.

In this introductory chapter 1, we just make an introduction to the subjects studied in this thesis, together with remembering some general theoretical questions, the main definitions and preliminaries for interpolation and approximation techniques using B-spline and radial functions, and fixing the required notation used along it.

Chapter 2 studies the optimal knots allocation in smoothing splines of one and two variables by special computations using the NSGA-II algorithm, and make some numerical comparisons with the last few known studies.

In Chapter 3, a similar problem but by interpolation, of obtaining the optimal knots allocation in the bicubic spline case, by an optimization paradigm for placement of knots in cubic and bicubic splines interpolation is accomplished.

In a final Chapter 4, we study the Radial Basis Functions (RBFs) approach, where the Gaussian one is a famous and very well known family. This part also presents new methodologies for optimal placement of random centers, for approximating or fitting curve to data using Gaussian radial basis functions of one and two variables, using a technique that optimize the number of centers and its optimal placement by a multi-objective genetic algorithm. Also compactly supported RBFs are taken into consideration for this type of problems.

1.2 Introduction

This chapter presents many important theoretical topics related to this work, which include some necessary notation, Hilbert and Sobolev spaces theory and definitions, the concept of interpolating and approximating B-splines, etc. Also, we introduce the problem of radial basis function approximation, and we present basic definitions and considerations for variational splines and multi-objective evolutionary algorithms.

Function approximation and interpolation are relevant problems in many applied domains; they are also key technologies in the field of geometric modeling, inverse engineering, computer vision, and so on. In the last few decades, a huge amount of research has been done both developing a theoretical framework and building algorithms with different splines and radial basis functions, especially the compactly supported radial basis functions.

In other hand, function approximation and interpolation with B-splines or radial basis functions play an important role in applied sciences for the industry standard shape modeling as it is found in the construction of care bodies, airplane fuselage and many others.

In addition, the multi-objective optimization problems contribute significantly to many mathematical and physical applications, as they have the possibility to optimize more than one parameter or function at the same time to obtain the best, or at least one of the best possible solutions.

Now, it is necessary to describe the general idea of genetic algorithms (GAs) whose use of randomized operators, operating over a population of candidate solutions to generate new points in the search space.

To obtain an optimal solution in a multi-objective optimization problem, a Non-dominated Sorting Genetic Algorithm is developed in a way to avoid the large number of local minima existing in the problem of knots placement; this algorithm work through different stages: population initialization, non-dominated sorting, crowding distance, selection, genetic operator, and finally recombination and selection.

1.3 Notation

As usual, the symbols $\mathbb{R}$, $\mathbb{Z}$ and $\mathbb{N}$ represent, respectively, the sets of real, integer and natural numbers. In later chapters, the subset ${\mathbb{R}^*}_+ = \{x \in \mathbb{R} : /; x > 0\}$ also appears. Let $n \in \mathbb{N}^* \equiv \{n \in \mathbb{N} : n > 0\}$; it will be considered that $\mathbb{R}^n$ is endowed with this Euclidean scalar product $\langle ., . \rangle_n$ and the associated usual

norm $\langle . \rangle_n$, where

$$\forall\, \mathbf{x} = (x_1, \ldots, x_n),\ \mathbf{y} = (y_1, \ldots, y_n) \in \mathbb{R}^n, \langle \mathbf{x}, \mathbf{y} \rangle_n = \sum_{i=1}^{n} x_i y_i$$

$$\forall \mathbf{x} \in \mathbb{R}^n,\ \langle \mathbf{x} \rangle_n = \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle}_n$$

Remember that the set $\mathbb{R}^n$ has Hilbert space structure with this Euclidean scalar product.

For all $\mathbf{x} \in \mathbb{R}^n$, and for all $\lambda > 0$, we define the sets

$$\mathbf{B}(\mathbf{x}, \lambda) = \{\mathbf{y} \in \mathbb{R}^n\ : \langle \mathbf{x} - \mathbf{y} \rangle_n < \lambda\},$$

$$\overline{\mathbf{B}}(\mathbf{x}, \lambda) = \{\mathbf{y} \in \mathbb{R}^n\ : \langle \mathbf{x} - \mathbf{y} \rangle_n \leq \lambda\},$$

called, respectively, open and closed balls of centers $\mathbf{x}$ and radius λ.

The canonical basis of $\mathbb{R}^n$ will be indicated through $\{\mathbf{e}_1, \ldots, \mathbf{e}_n\}$, and

$$p_i : \quad \mathbb{R}^n \mapsto \mathbb{R}$$

$$(x_1, \ldots, x_n) \longmapsto x_i,\ \ i = 1, \ldots, n,$$

denote the canonical projections of $\mathbb{R}^n$ into $\mathbb{R}$

Given $E \subseteq \mathbb{R}^n$, we will indicate by $\bar{E}, \partial E, int(E)$ and $card(E)$, respectively, the closure, the boundary, the interior and the cardinal of E. We will call the diameter of E to be the positive real number

$$diam(E) = \sup_{\mathbf{x}, \mathbf{y} \in E} \langle \mathbf{x} - \mathbf{y} \rangle_n$$

For all natural $p \in \mathbb{N}^\star$, $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_p) \in \mathbb{N}^p$ it is defined

$$|\boldsymbol{\alpha}| = \sum_{i=1}^{p} |\alpha_i| \quad and \quad |\boldsymbol{\alpha}|_\infty = \max_{1 \leq i \leq p} |\alpha_i|$$

where $|\boldsymbol{\alpha}|$ is called the order of $\boldsymbol{\alpha}$.

Given two normed spaces $(X, \|.\|_X), (Y, \|, \|_Y)$, lets denote by $\mathcal{L}(X, Y)$ the space of linear and continuous applications $\Phi : X \to Y$ and it will be designated by X', the topological dual space of the normed space X; that is,

the space $\mathcal{L}(X, \mathbb{R})$. Also, given $\Phi \in \mathcal{L}(X, Y)$, it will be indicated by $ker(\Phi)$, as usual, the kernel of Φ; that is, the subspace

$$ker(\Phi) = \{\mathbf{x} \in X \ : \ \Phi(\mathbf{x}) = 0\}$$

For all $m, n \in \mathbb{N}^*$, it is designated by $\mathcal{M}_{m,n}$ the space of matrices of real numbers that have m rows and n columns.

We will use the notations $\mathbf{A} = (a_{ij}),\ a_{ij} \in \mathbb{R},\ 1 \leq i \leq m,\ 1 \leq j \leq n$, or $A = (\mathbf{a}_i), \mathbf{a}_i \in \mathbb{R}^m,\ 1 \leq i \leq n$ indistinctly, to designate $\mathbf{A} \in \mathcal{M}_{m,n}$.

We provide this space with the following Euclidean scalar product

$$\forall \mathbf{A} = (a_{ij}),\ \mathbf{B} = (b_{ij}) \in \mathcal{M}_{m,n},\ \langle \mathbf{A}, \mathbf{B} \rangle_{m,n} = \sum_{i=1}^{m} \sum_{j=1}^{n} a_{ij}\, b_{ij},$$

and the corresponding matrix norm

$$\forall \mathbf{A} \in \mathcal{M}_{m,n},\ \langle \mathbf{A} \rangle_{m,n} = \langle \mathbf{A}, \mathbf{A} \rangle_{m,n}^{\frac{1}{2}}.$$

Let f and g be two real functions of real variable, we denote by

$$f(d) = \mathcal{O}(g(d)), \quad d \to d_0,$$

if there are real constants $C > 0$ and $\eta > 0$ such that

$$\forall d \in \mathbb{R},\ |d - d_0| < \eta,\ |f(d)| \ \leq C|g(d)|.$$

In the same way, we will write

$$f(d) = o(g(d)), \ \ d \to d_0,$$

when

$$\lim_{d \to d_0} \frac{f(d)}{g(d)} = 0.$$

1.4 Hilbert and Sobolev spaces

In this section we introduce some definitions and results of functional analysis that will be used in the sequel. They can be understood and expanded through the references [3], [13], [81] and [6].

Definition 1.4.1. *Suppose H be a linear space over $\mathbb{V} = \mathbb{R}$ or $\mathbb{C}$. An inner product is a map :* $\langle .,. \rangle : H \times H \longrightarrow \mathbb{V}$
that satisfy the following properties:

(i) Conjugate symmetry : for any $x, y \in H$, $(x.y) = \overline{(y.x)}$.

(ii) Linearity: for any $x, y, z \in H$, and $\alpha, \beta \in \mathbb{V}$,

$$(\alpha x + \beta y, z) = \alpha(x, z) + \beta(y, z).$$

(ii) Positive-definite: for any $x \in H$, $(x, x) \geq 0$, and $(x, x) = 0$ if and only if $x = 0$.

The space H, together with such an inner product, is called an inner product space.

Definition 1.4.2. *A Hilbert space is a vector space H equipped with an inner product, such that H is complete for the norm $\| \, . \, \|$.*

In what follows, H will always denote a Hilbert space.

Definition 1.4.3. *Let Y be a normed space with norm $\| \, . \, \|_Y$; a sequence $\{y_n\}_{n \in N} \subseteq Y$ is strongly convergent or simply converges to $y \in Y$ if*

$$\lim_{n \to +\infty} \| \, y_n - y \, \|_Y = 0.$$

Definition 1.4.4. *Also if Y is a Hilbert space, with the scalar product $(.,.)_Y$, then the sequence $\{y_n\}_{n \in N}$ is said that weakly converges to y in Y if:*

$$\forall u \in Y, \ \lim_{n \to +\infty} (y_n, u)_Y = (y, u)_Y$$

If the sequence $\{y_n\}_{n \in \mathbb{N}}$ has strong convergence to y, it will be denoted by

$$y_n \to y \text{ or } \lim_{n \to +\infty} y_n = y$$

whereas the weak convergence will be denoted by

$$y_n \rightharpoonup y \ \ or \lim_{n \to +\infty}{}^{*} y_n = y.$$

Definition 1.4.5. *Let X be a normed space. It is said that a set $E \subset X$ is relatively compact in X, if the closure of E is a compact subset of X.*

Lemma 1.4.6. *Let Y be a Hilbert space and $\{y_n\}_{n\in N}$ a sequence in Y. It is verified that:*

(1) if $y_n \to y$, then $y_n \rightharpoonup y$.

(2) if $y_n \rightharpoonup y$, then $\{y_n\}_{n\in\mathbb{N}}$ is bounded in Y and $\| y \| \leq \liminf_{n\to\infty} \| y_n \|$.

(3) if $\{y_n\}_{n\in\mathbb{N}}$ is bounded in Y and all subsequences weakly convergent extracted from it have the same weak limit y, then $y_n \rightharpoonup y$;

(4) if $y_n \rightharpoonup y$, then for all $f \in X'$, $f(y_n) \rightharpoonup f(y)$;

(5) if $y_n \rightharpoonup y$ and $\| y_n \| \to \| y \|$, then $y_n \to y$.

Lemma 1.4.7. *Let Y be a Hilbert space, X a Banach space, and $T \in \mathcal{L}(X, Y)$. Then, T is compact if and only if, for each succession $\{x_n\}$ of X weakly convergent to x in X, the sequence $\{Tx_n\}_{n\in\mathbb{N}}$ is strongly convergent to Tx in Y*

Theorem 1.4.8. *(Hilbert Space Projection Theorem): Let M be a closed subspace of a Hilbert space H and let v in H be given. Then*

(a) $\exists$ a unique v_M in M such that $\| v - v_M \|_H \leq \| v - z \|_H$, for all z in M, (v_M is the best approximation in M to v because it is the unique point in M closest to v).

(b) $\forall z \in M, \langle v - v_M, z\rangle_H = 0$

(c) all v in H may be uniquely epressed as $v = v_M + x_M$, where $Pv = v_M \in M$, $Qv = x_M \in M^\perp$ and $\| v \|_H^2 = \| Pv \|_H^2 + \| Qv \|_H^2$.

In many problems in the area of Mathematical Physics and Variational Calculus it is not enough to deal with the classical solutions of differential equations; it is also important to define the notion of weak derivatives and to work in the so called Sobolev spaces.

The most important part is that the solutions of partial differential equations are naturally found in Sobolev spaces, rather than in spaces of continuous functions and with the derivatives understood in the classical sense.

Before talking about Sobolev spaces, it is important to define the $\mathcal{L}^p$ spaces: let D be an open set of $\mathbb{R}^d$, define the support of a real function on D, $v : D \longrightarrow \mathbb{R}$, as

$$sup(v) = \overline{\{\mathbf{x} \in D \,/\, v(\mathbf{x}) \neq 0\}}$$

We say that a function v is of class k in this open domain D, and we will denote it $v \in \mathcal{C}^k(D)$, if v is k times differentiable on D with continuous derivatives; of Schwartz' class of real functions on D, which we will denote $\mathcal{D}(D)$, as the set of function

$$v : D \longrightarrow \mathbb{R}$$

that are indefinitely derivable over D and with compact support in D (which is usually indicated $supp(D) \subset\subset D$); that is to say,

$$\mathcal{D}(D) \equiv \mathcal{C}_0^\infty(D) = \{v : D \longrightarrow \mathbb{R} \,/\, v \in \mathcal{C}^\infty(D),\ supp(D) \subset\subset D\}\,.$$

A set $E \subset R^d$ is called a set of measure zero if it can be covered with a numerable system of open cubes (n-cubes) in which the sum of volumes is as small as one wishes

$$\forall \varepsilon > 0,\ \ E \subset \bigcup_{i=1}^{\infty} K_i,\ \ \Sigma_{i=1}^{\infty} |K_i| < \varepsilon$$

and certain property is verified for all $\mathbf{x} \in D \subset \mathbb{R}^d$, except at most in a set of measure zero, it is said to be verified in almost every point of D.

Definition 1.4.9. *We will say that $v \in \mathcal{L}(D)$ is integrable Lebesgue or absolutely integrable ($\int_D |v|\, d\mathbf{x}) < +\infty$) . We will say that $v \in \mathcal{L}^p(D)$ (with $1 \leq p < +\infty$) if it exists and is finite the integral $\int_D |v|^p\, d\mathbf{x} < +\infty$.*

Proposition 1.4.10. *It can be seen that $\mathcal{D}(D)$ is dense in $\mathcal{L}^p(D)$ for $1 \leq p < +\infty$, in the sense that any function of this last space could be obtained*

as the limit of a sequence, of infinitely derivable functions. In addition, in $\mathcal{L}^p(D)$ *you can define the norm*

$$\| v \|_{\mathcal{L}^p(D)} = \left(\int_D (|v|^p)\, d\mathbf{x} \right)^{\frac{1}{p}}$$

that gives this space a Banach space structure, for every $1 \leq p < \infty$.

Given a multi-index $\boldsymbol{\alpha} = (\alpha_1, \alpha_2 ..., \alpha_d) \in \mathbb{N}^d$, it is denoted

$$|\boldsymbol{\alpha}| := \sum_{i=1}^{d} \alpha_i,$$

$$\partial^{\boldsymbol{\alpha}} \equiv \frac{\partial^{|\boldsymbol{\alpha}|}}{\partial x_1^{\alpha_1} \dots \partial x_d^{\alpha_d}}.$$

It is said that a function $v : D \longrightarrow \mathbb{R}$ admits weak partial derivatives of order $\boldsymbol{\alpha} \in \mathbb{N}^d$, $\partial^{\boldsymbol{\alpha}} v$ (also called partial derivative in the sense of distributions), if there is a function $\partial^{\boldsymbol{\alpha}} v : D \longrightarrow \mathbb{R}$ such that:

$$\int_D \partial^{\boldsymbol{\alpha}} v \, \varphi \, dx = (-1)^{|\boldsymbol{\alpha}|} \int_D v \, \partial^{\boldsymbol{\alpha}} \varphi \, dx, \quad \forall \varphi \in \mathcal{D}(D).$$

Sobolev's space $H^k(D)$ can be also deduced as the space of functions $v \in \mathcal{L}^2(D)$ for which there exists every partial weak derivative $\partial^{\boldsymbol{\alpha}} v$, with $|\boldsymbol{\alpha}| \leq k$, also belonging to $\mathcal{L}^2(D)$; that is,

$$H^k(D) = \left\{ v : D \longrightarrow \mathbb{R} \ / \ \partial^{\boldsymbol{\alpha}} v \in \mathcal{L}^2(D), \forall |\boldsymbol{\alpha}| \leq k \right\}.$$

Equipped with the norm

$$\|v\|_{k,D} := \left(\sum_{|\boldsymbol{\alpha}| \leq k} \int_D (\partial^{\boldsymbol{\alpha}} v)^2 \, d\mathbf{x} \right)^{\frac{1}{2}},$$

the $H^k(D)$ space is a Hilbert space.

Proposition 1.4.11. *If we assume that* E *is a set of points* $D \subset \mathbb{R}^d$ *(usually* $d = 1, 2$ *in our applications) that contains at least one other subset of points* $\mathbb{P}_r$*-unisolvent, we can assure that the expression*

$$[[v]]_{E,r} = \left(\sum_{p \in E} v(p)^2 + |v|_{r+1}{}^2 \right)^{\frac{1}{2}}$$

is a norm on $H^{r+1}(D)$ equivalent to the usual norm $\|.\|$, defined in this Sobolev space.

The Sobolev space $H^k(D)$ is a Hilbert space equipped with the inner semi-products given by

$$(u,v)_l = \sum_{|\boldsymbol{\alpha}|=l} \int_D \partial^{\boldsymbol{\alpha}} u(x)\, \partial^{\boldsymbol{\alpha}} v(\mathbf{x})\, d\mathbf{x}, \;\; 0 \leq l \leq k$$

the semi-norms given by $|u|_l = (u,u)_l^{\frac{1}{2}}$, for all $l = 0, \cdots, k$, and the norm $\|u\|_k = \left(\sum_{l \leq k} |u|_l^2\right)^{\frac{1}{2}}$.

1.5 B-splines

It is more than 70 years since I. J. Schoenberg [61] introduced spline functions into the mathematical literature. Since then, splines have proved to be enormously important in several branches of Mathematics, such as approximation theory, numerical analysis, numerical treatment of differential, integral and partial differential equations, statistics and so on.

B-Spline functions are special functions in approximation theory, because they are constructed using low order polynomial functions, joined with smoothness conditions at breakpoints. We present a definition of the B-Spline functions, and some important properties needed when using uniform B-Splines in a Finite Elements method. Finally, also the interpolating and approximating splines, will be introduced.
The theory of B-splines functions has been developed among many references and is shown for example in [27], [57], [5], [14], [63] and [62].

1.5.1 Definition and properties

B-splines functions are capable of changing in order to be appropriate for numerical theories of interpolation, data adjustment, and so on. These functions enable the creation and management of complex shapes and surfaces using a limited number of points.

Given a knot sequence

$$\ldots < t_{-2} < t_{-1} < t_0 < t_1 < t_2 \ldots$$

verifying

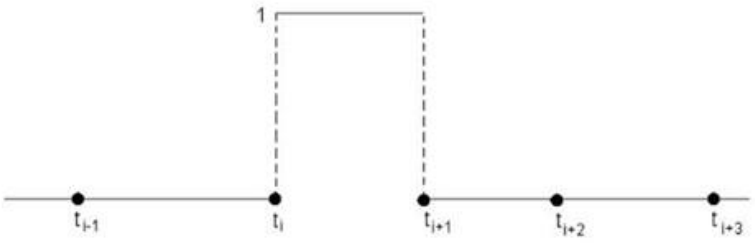

Figure 1.5.1: Examples of B-splines of degree 0

$$\lim_{i\to+\infty} t_{-i} = -\infty, \qquad \lim_{i\to+\infty} t_i = +\infty.$$

Now, we can provide different families of B-splines; first, it is necessary to define the B-splines of zero degree over the interval $[t_i, t_{i+1}]$.

The classical definition of the B-spline of order 0, described as:

$$\forall i \in \mathbb{Z} \qquad B_i^0(x) = \begin{cases} 1, & \text{if} \quad t_i \leq x < t_{i+1}, \\ 0, & \text{otherwise.} \end{cases}$$

The graph of the B-splines of degree 0 is shown in Figure 1.5.1. Among the most important properties of this family we highlight:

a) $B_i^0(x) = 0$, $\forall x \notin [t_i, t_{i+1})$.

b) is continuous on the right across the real line $\mathbb{R}$.

c) $B_i^0(x) \geq 0$, $\forall x \in \mathbb{R}$, $\forall i \in \mathbb{Z}$.

d) $\sum_{i=-\infty}^{+\infty} B_i^0(x) = 1$, $\forall x \in \mathbb{R}$.

Definition 1.5.1. *The B-Splines $B_i^k(x)$, in general are defined by:*

$$B_i^k(x) = \left(\frac{x - t_i}{t_{i+k} - t_i}\right) B_i^{k-1}(x) + \left(\frac{t_{i+k+1} - x}{t_{i+k+1} - t_{i+1}}\right) B_{i+1}^{k-1}(x), \quad k \geq 1. \quad (1.1)$$

where $k \in \mathbb{N}$.

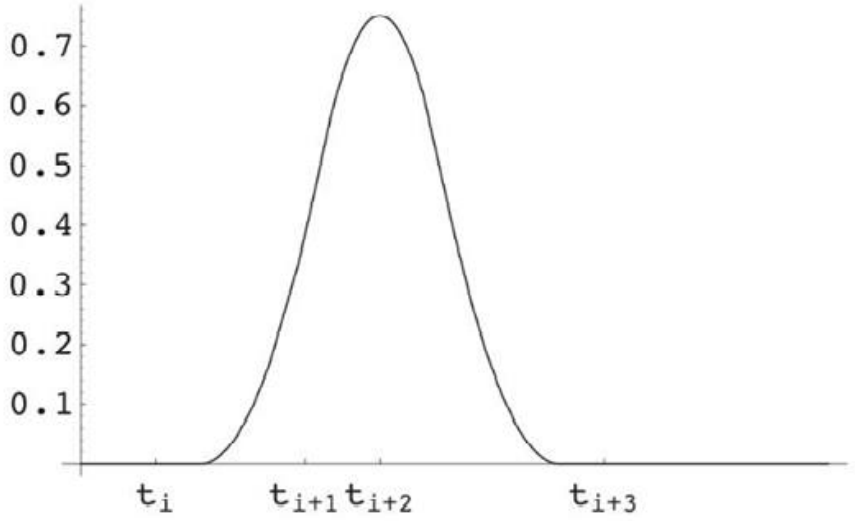

Figure 1.5.2: Graph of the quadratic B-spline of class $\mathcal{C}^1$

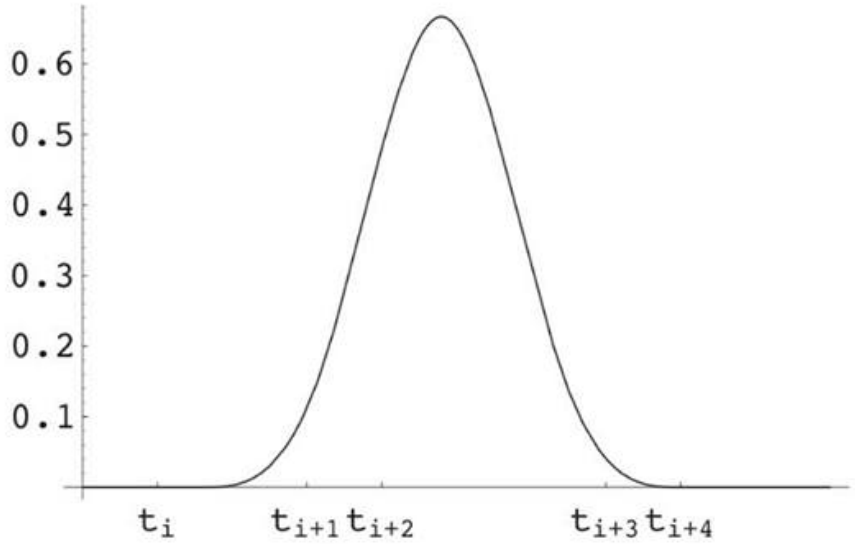

Figure 1.5.3: Graph of cubic B-spline of class $\mathcal{C}^2$

For B-splines of order k, we have for each x not belonging to the interval $[t_i, t_{i+k+1})$, $B_i^k(x)$ equal 0 (compact support), while $B_i^k(x) > 0, \forall x \in (t_i, t_{i+k+1})$ (positivity), and $\forall x \in \mathbb{R}$,

$$\sum_{i=-\infty}^{+\infty} B_i^k(x) = 1.$$

Besides, if $k \geq 2$, then the first derivative of $B_i^k(x)$ respect to x is defined by

$$\frac{d}{dx} B_i^k = \left(\frac{k}{t_{i+k} - t_i}\right) B_i^{k-1} - \left(\frac{k}{t_{i+k+1} - t_{i+1}}\right) B_{i+1}^{k-1}.$$

Also if $k \geq 1$, then B_i^k belongs to $\mathcal{C}^{k-1}(\mathbb{R})$(continuity), and

$$\int_{-\infty}^{x} B_i^k(s)ds = \left(\frac{t_{i+k+1} - t_i}{k+1}\right) \sum_{j=i}^{\infty} B_j^{k+1}(x), \quad \forall k.$$

Meanwhile the set of B-splines $\left\{B_i^k, B_{i+1}^k, \ldots, B_{i+k}^k\right\}$ is linearly independent in (t_{i+k}, t_{i+k+1}). It is usual to work with the B-splines of degree 2 and 3, whose graphs are represented in figures 1.5.2 and 1.5.3

1.5.2 Interpolating and approximating splines

In many applications, one wishes to interpolate or approximate univariate data by spline functions possessing certain geometric properties or shapes, like monotonicity, convexity or nonnegativity. An efficient method for interpolation and approximation of both curve and surface points using B-splines is described. Given $\Omega = (a, b)$ and $\Delta_m = a = t_0 < t_1 < \ldots < t_m = b \subset \overline{\Omega}$, then

$$S(k, r; \Delta_m) = \{s \in \mathcal{C}^r(\overline{\Omega}) : s|_{[t_{i-1}, t_i]} \in \mathbb{P}_k[t_{i-1}, t_i], \quad i = 1, \ldots, m\}.$$

is the vector space of functions with global class r that are piecewise polynomials of degree $\leq k$ in every sub-interval of the partition Δ_m.

Theorem 1.5.2. *A basis of the vector space $S(k, k-1; \Delta_m)$ is*

$$\{B_i^k|_{\overline{\Omega}} : -k \leq i \leq m-1\}.$$

As a consequence, the dimension of $S(k, k-1; \Delta_m)$ is $m + k$.

Let us suppose a series of data $\{y_1, ..., y_m : y_i \in \mathbb{R}\}$; then the Lagrangian interpolation problem is stated in the following way:
Find $s \in S(k, k-1; \Delta_m)$ such that $s(t_i) = y_i,\ 0 \leq i \leq m$. It is shown that the linear forms associated with this problem are linearly independent. As the dimension of space $S(k, k-1; \Delta_m) = m+k$ and there are $m+1$ ligatures or conditions, we have $k-1$ possibilities of election to assure the unisolvency of the proposed problem.
In the case of linear piecewise splines, we have to interpolate in the space $S(1, 0; \Delta_m)$, $k-1=0$, so that we do not have degrees of freedom. When working with quadratic splines of class 1; that is, in the space $S(2, 1; \Delta_m)$, $k-1=1$; and to obtain the unisolvency, an additional data must be entered. It is usual to add the value of the derivative at any of the extremes t_0 or t_m of the interval, or to minimize some amount related to the spline, such as an estimate of its bending energy.
One of the most common situations is that of the space of splines of degree $2k-1$ and class $2k-2$; that is, $S(2k-1, 2k-2; \Delta_m)$ for a certain $k \in \mathbb{N}$; to guarantee the unisolvency, in this case $2k-2$ additional conditions must be imposed (see [27]).

Proposition 1.5.3. *Unisolvent interpolation problems:*

1) The general Hermite interpolation problem, with conditions given on the derivatives at the end points by:

let $g \in \mathcal{C}^k(\overline{\Omega})$, *find* $s \in S(2k-1, 2k-2; \Delta_m)$ *such that :*

$$s(t_i) = g(t_i) \quad i = 0, \ldots, m, \tag{1.2}$$

$$s^{(\mu)}(a) = g^{(\mu)}(a) \qquad s^{(\mu)}(b) = g^{(\mu)}(b), \quad \mu = 1, \ldots, k-1. \tag{1.3}$$

2) The problem of natural interpolation, with natural conditions at the end points:

given $g \in \mathcal{C}^k(\overline{\Omega}), \quad 2 \leq k \leq m-1$ *find* $s \in S(2k-1, 2k-2; \Delta_m)$ *such that:*

$$s(t_i) = g(t_i), \quad i = 0, \ldots, m, \tag{1.4}$$

$$s^{(\mu)}(a) = s^{(\mu)}(b) = 0, \quad \mu = k, \ldots, 2k-2. \tag{1.5}$$

3) The interpolation problem with periodic conditions:

given $g \in \mathcal{C}^k(\overline{\Omega})$, *with* $g^{(\kappa)}(a) = g^{(\kappa)}(b)$ *for* $\kappa = 0, \ldots, k-1$, *find* $s \in S(2k-1, 2k-2; \Delta_m)$ *such that:*

$$s(t_i) = g(t_i) \quad i = 0, \ldots, m,$$

$$s^{(\mu)}(a) = s^{(\mu)}(b), \quad \mu = 1, \ldots, 2k-2.$$

In any case, it will always be necessary to verify that the set of all the linear forms associated with the problem, the initial ones and those of the conditions that are introduced, are linearly independent. We also remember the Schoenberg-Whitney Theorem (1953), which is fundamental in the study of polynomial interpolation.

Theorem 1.5.4. *The spline* $s \in S(k, k-1; \Delta_m)$ *is the unique spline that interpolates the values* $\{y_1, \ldots, y_{m+k}\}$ *at the points* $\{\gamma_1 < \ldots < \gamma_{m+k}\}$ *if and only if*

$$B^k_{j-m-1}(\gamma_j) \neq 0, \quad 1 \leq j \leq m+k.$$

Note 1.5.5. *When interpolating values where the function is known,* $g \in \mathcal{C}([t_0, t_m])$, *where* $x_1, \ldots, x_n \in [t_0, t_m]$, *the Lagrange interpolation operator of* g *is defined as:*

$$\mathcal{L}f(x) = \sum_{i=1}^{m+k} f(x_i) B^k_i(x), \quad \forall x \in [t_0, t_m]. \tag{1.6}$$

Theorem 1.5.6. *Given* $\Omega = (a, b)$ *and* $\Delta_m = \{a = t_0 < t_1 < \ldots < t_m = b\}$. *Suppose* s *the cubic spline* $s \in S(2k-1, 2k-2; \Delta_m)$ *with Hermite conditions (1.2) and (1.3) interpolates* $g \in \mathcal{C}^2(\overline{\Omega})$ *in* Δ_m. *Then*

$$|g(x) - s(x)| \leq h^{\frac{3}{2}} \left(\int_\Omega \left(g''(t)\right)^2 dt \right)^{\frac{1}{2}}, \quad \text{where} \quad h = \max_{0 \leq j \leq m-1} (t_{j+1} - t_j).$$

Theorem 1.5.7. *Given a uniform partition* Δ_m *of step* h *of* $\overline{\Omega}$, *and* $\Omega = (a, b)$, *if* s *the cubic spline with condition (1.2) and (1.3) interpolates that*

$g \in \mathcal{C}^4(\overline{\Omega})$ in Δ_m. Then we have

$$|g - s|_{l,\overline{\Omega}} \leq K_l h^{4-l}, \quad l = 0, 1, 2, 3,$$

where K_l, $l = 0, 1, 2, 3$, depends only on g and $h = \max_{0 \leq j \leq m-1}(t_{j+1} - t_j)$.

Theorem 1.5.8. *Let $\Omega = (a, b)$, $h = \dfrac{b-a}{m}$ and consider a uniform partition with step-size h of $\overline{\Omega}$. Suppose s is the cubic spline $s \in S(2k-1, 2k-2; \Delta_m)$, with condition (1.4) and (1.5) that interpolates $g \in \mathcal{C}^4(\overline{\Omega})$ in the nodes of the partition considered. Then we have*

$$\|g - s\|_{\infty,\Omega} \leq \frac{5}{384} h^4 \|g^{(4)}\|_{\infty,\Omega}.$$

The following theorem gives us information about the error for the three interpolation cases presented in Proposition 1.5.3, and also offers information about the convergence of spline derivatives, but it is not optimal in terms of error bounds.

Theorem 1.5.9. *Let $\Omega = (a, b)$ and $\Delta_m = \{a = t_0 < t_1 < \ldots < t_m = b\}$. Let's consider $s \in S(2k-1, 2k-2; \Delta_m)$ the spline that interpolates $f \in \mathcal{C}^k(\overline{\Omega})$, $k \geq 2$ with Δ_m. Then, for $0 \leq l \leq k-1$ we have*

$$\|f^{(l)} - s^{(l)}\|_{\infty,\Omega} \leq \frac{k!}{l!\sqrt{k}} h^{k-l-\frac{1}{2}} \, \|f^{(k)}\|_{2,\Omega}, \quad h = \max_{0 \leq j \leq m-1}(t_{j+1} - t_j).$$

1.6 Radial basis functions

Radial basis functions (RBF), which return a value as a function of the radius from a given center, came into the attention of the engineering community quite a while ago. The (RBF) is a method for multivariate approximation, which is one of the most applied approaches in modern approximation theory when the task is to approximate scattered data in several dimension. There are many interesting references in radial basis function approximation; such as [26], [16], [48], [50], [8], [4], just to mention a few.

Definition 1.6.1. *A radial function is a function defined on a Euclidean space $\mathbb{R}^d$ whose value at each point depends only on the distance between that*

point and the origin. For example, a radial function Φ *in two dimensions has the form*

$$\Phi(x, y) = \varphi(r), \quad \text{where } r = \sqrt{x^2 + y^2}.$$

Definition 1.6.2. *A radial basis function, RBF,* $\Phi(x)$ *is a function with respect to the origin or a certain point c, ie,* $\Phi(x) = f(\| x - c \|)$ *where the norm is usually the Euclidean norm.*

There are two main groups of Radial Basis Functions: global RBFs and Compactly Supported RBFs (CS-RBFs). Fitting scattered data with CS-RBFs leads to a simpler and faster computation, but techniques using CS-RBFs are sensitive to the density of scattered data. Global RBFs lead to a linear system of equations with a dense matrix and their usage is based on sophisticated techniques such as the fast multiple method. Global RBFs are useful in repairing incomplete datasets and they are insensitive to the density of scattered data.

Definition 1.6.3. *A function* $f : [0, \infty) \longrightarrow \mathbb{R}$ *will be called positive definite on a Hilbert space (HPD), if the matrix*

$$\left(f \parallel x_j - x_k \parallel^2\right)_{j,k=1}^{n} \tag{1.7}$$

is non-negative definite for every positive integer n and any points $x_1, ..., x_n \in H$.

There are several ways to demonstrate, or create, positive definite radial basis functions. In Table 1.6, we give common strictly positive RBFs and SPD-RBFs.

1.6.1 Compactly Supported RBFs

Compactly supported radial basis functions have been invented for the purpose of getting finite-element type approximations. They give rise to sparse interpolation matrices and can be used to solve numerically partial differential equations

Some well known definitions and results about general radial basis functions are the following, that can be consulted in some of the references mentioned above: [26], [16], [48], [50], [8], [4].

Table 1.6.1: common strictly positive definite radial basis functions ($r = r_i = \| P - P_i \|$, u and K_ν are respectively the Heaviside and the spherical Bessel functions.

Name	$\Phi(r)$	**Parameters**
Gaussian	$exp(-\epsilon r)^2$	$\epsilon > 0$
Sobolev Splines	$r^\nu K_\nu(r)$	$\nu > 0$
Inverse Multiquadrics	$(r_2 + c^2)^{-\epsilon/2}$	$\epsilon > n/2$
Compact Support	$u(1-r)^l P(r)$	$l \geq \lfloor n/2 \rfloor + 1$

Definition 1.6.4. *Let ϑ be such that $x \mapsto t\vartheta(t) \in \mathcal{L}^1[0,\infty)$, then we define*

$$I\vartheta(r) = \int_r^\infty t\vartheta(t),\ r \geq 0.$$

so that, for even functions $\vartheta \in \mathcal{C}^2(\mathbb{R})$ we define

$$(D\vartheta)(r) = -\frac{1}{r}\vartheta'(r),\ \ r \geq 0.$$

Theorem 1.6.5.

a. If both D and I preserve compact support, then so do $D\vartheta$ and $I\vartheta$.

b. If $\vartheta \in \mathcal{C}(\mathbb{R})$ and $t \mapsto t\,\vartheta(t) \in \mathcal{L}^1[0,\infty)$, then $DI\vartheta = \vartheta$.

c. If $\vartheta \in \mathcal{C}^2(\mathbb{R})$ is even and $\vartheta' \in \mathcal{L}^1[0,\infty)$, then $ID\vartheta = \vartheta$.

Theorem 1.6.6.

1. Suppose $\vartheta \in \mathcal{C}(\mathbb{R})$. If $t \mapsto t^{s-1}\vartheta(t) \in \mathcal{L}^1[0,\infty)$ and $s \geq 3$ then ϑ is strictly positive definite and radial on $\mathbb{R}^s$ if and only if $I\vartheta$ is strictly positive definite and radial on $\mathbb{R}^{s-2}$.

2. If $\vartheta \in \mathcal{C}^2(\mathbb{R})$ is even and $t^s \mapsto t\,\vartheta'(t) \in \mathcal{L}^1[0,\infty)$, then ϑ is strictly positive definite and radial on $\mathbb{R}^s$ if and only if $D\vartheta$ is strictly positive definite and radial on $\mathbb{R}^{s+2}$.

1.6.2 Wendland Compactly Supported Functions

Wendland functions are compactly supported Radial Basis Functions (CS-RBFs) that are used for interpolation of data or for solving Partial Differential Equations with mesh-free collocation methods. Wendland constructs a popular family $\vartheta_{s,k}$ of CS-RBFs, expressed in a polynomial form, whose degree is minimal for a given dimension space s, and whose continuity is $\mathcal{C}^{2k}$. Wendland also gives a recursive formula for the functions $\vartheta_{s,k}$ for all s and k.

The functions $\vartheta_{s,k}$ $k = 0, 1, 2, 3,$, $l = \lfloor s/2 \rfloor + k + 1$, have the form
$\vartheta_{s,0}(r) = (1-r)^l_+$,
$\vartheta_{s,1}(r) \doteq (1-r)^{l+1}_+ \left((l+1)+1\right)$,
$\vartheta_{s,2}(r) \doteq (1-r)^{l+2}_+ \left((l^2+4l+3)r^2+(3l+6)r+3\right)$,
$\vartheta_{s,3}(r) \doteq (1-r)^{l+3}_+ (l^3+9l^2+23l+15)r^3+(6l^2+36l+45)r^2+(15l+45)r+15)$,

Note 1.6.7. *The symbol $\doteq$ denotes equality up to a multiplicative positive constant and $\lfloor \cdot \rfloor$ the integer part function.*

Definition 1.6.8. *Let $\vartheta_l(r) = (1-r)^l_+$, then we define $\vartheta_{s,k} = I^k \vartheta_{\lfloor s/2 \rfloor + k + 1}$.*

It turns out that the functions $\vartheta_{s,k}$ are all supported on $[0, 1]$ and have a polynomial representation there.

Theorem 1.6.9. *Given univariate polynomials $p_{s,k}$ of degree $\lfloor s/2 \rfloor + k + 3$, and $\vartheta_{s,k} \in \mathcal{C}^{2k}(\mathbb{R})$, the function:*

$$\vartheta_{s,k}(r) = \begin{cases} P_{s,k}(r), & r \in [0,1], \\ 0, & r > 1, \end{cases}$$

are strictly positive definite and radial on $\mathbb{R}^s$.

1.7 Variational splines

Spline functions have proved to be very useful in numerical analysis, in numerical treatment of differential, integral, and partial differential equations; and have found applications in Science, Engineering, Biology, Medicine, etc. It is well known that interpolating/approximating polynomial splines can be derived as the solution of certain variational problems. In this section we present both smoothing and interpolating variational splines.

1.7.1 Smoothing variational splines

Smoothing splines are well known for providing nice curves and surfaces from exact or noisy data. In Geology and Structural Mechanics, the reconstruction and/or approximation of a curve or surface from a scattered data set is a commonly encountered problem.

Let be $k, r, s \in \mathbb{N}^*$, $D \subset \mathbb{R}^d$, a non empty open set and $\langle ., . \rangle_k$, $\langle . \rangle_k$, the Euclidean inner product and corresponding norm in $\mathbb{R}^k$, as usual.

We denote $H^s(D; \mathbb{R}^k)$ the corresponding space of Sobolev functions with k components and we can also define:

$$\|u\|_{s,D,\mathbb{R}^k} = \left(\sum_{|\boldsymbol{\alpha}| \leq s} \int_D \langle \partial^{\boldsymbol{\alpha}} u(\mathbf{x}) \rangle^2_k \, d\mathbf{x} \right)^{\frac{1}{2}}$$

the semi-norms

$$|u|_{l,D,\mathbb{R}^k} = \left(\sum_{|\boldsymbol{\alpha}| = s} \int_D \langle \partial^{\boldsymbol{\alpha}} u(\mathbf{x}) \rangle^2_k \, d\mathbf{x} \right)^{\frac{1}{2}} \quad 0 \leq l \leq s,$$

with their corresponding inner semi-products:

$$(u, w)_{l,D,\mathbb{R}^k} = \sum_{|\boldsymbol{\alpha}| = l} \int_D \langle \partial^{\boldsymbol{\alpha}} u(\mathbf{x}), \partial^{\boldsymbol{\alpha}} w(\mathbf{x}) \rangle_k \, d\mathbf{x}, \quad 0 \leq l \leq s.$$

For all $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_{M_2})$, (with $\theta_i \geq 0$, for all $i = 1, \ldots, M_2$; $\varepsilon > 0$), and suppose $J_{\varepsilon,\theta}$ functional on $H^s(D, \mathbb{R}^k)$ as follow:

$$J_{\varepsilon,\boldsymbol{\theta}} = \langle Lu - Lg \rangle^2_{M_1,n} + \langle \theta, \beta(u, u) \rangle_{\mathbb{R}^{M_2}} + \varepsilon \, |u|^2_{s,D,\mathbb{R}^k}$$

Then from [36] we have the following linear system:

$$\mathbf{B}\boldsymbol{\alpha} = \mathbf{d}$$

where

$$d_j = \langle L^b g, L^b u_j \rangle_{M_1,n}$$

and

$$b_{i,j} = \langle L^b u_i - L^b u_j \rangle_{M_1,k} + \langle \theta, \beta(u_i, u_j) \rangle_{\mathbb{R}^{M_2}} + \varepsilon \, (u_i, u_j)_{s,D,\mathbb{R}^k}$$

Where Σ^b is a set of linear forms $L^b : \mathcal{C}^\mu(\bar{D}, \mathbb{R}^k) \to \mathbb{R}^{M_1,n}$, $M_1 = card(\Sigma^b)$ and Θ^b is ordered finite set in $H^s(D, \mathbb{R}^k)$ with $M_2 = card(\Theta^b)$, A^b is an ordered finite subset of distinct points of $\bar{D}$.

The minimization problem is to find $\sigma^{dh}_{\varepsilon,\theta}$ such as:

$$\begin{cases} \sigma^{dh}_{\varepsilon,\theta}, \\ u,\ J_{\varepsilon,\theta}(\sigma_{\varepsilon,\theta}) \leq J_{\varepsilon,\theta}(u). \end{cases}$$

Theorem 1.7.1. *The problem 1.7.1 has a unique solution relative to A^b, L^b, β^b, L^b, and ε, which also has unique solution of the following variational problem:*

$$\begin{cases} \sigma_{\varepsilon,\theta} \in H^s(D,\mathbb{R}^n), \\ u \in H^s(D,\mathbb{R}^n),\ \langle L\sigma_{\varepsilon,\theta}, Lu\rangle_{M_1,n} + \langle \theta, \beta(\sigma_{\varepsilon,\theta}, u)\rangle_{\mathbb{R}^n} + \varepsilon(\sigma_{\varepsilon,\theta})_{s,D,\mathbb{R}^n} = \langle Lg, Lv\rangle_{M_1,n} \end{cases}$$

1.7.2 Interpolating variational splines

To find the suitable function that interpolates certain data points, many researchers such as Cauchy, Lagrange and Hermite, worked to develop the classic approach, through different techniques and methodologies, to find the best correspondig polynomials.

Proposition. Z is a non-empty closed and convex subset of $H^s(D,\mathbb{R}^k)$. Also there is an affine variety associated with the orthogonal space

$$Z_0 = \{u \in H^s(D,\mathbb{R}^n) \ \mid Lu = 0\}$$

To study the interpolating variational spline problems, we will use the corresponding semi-norm defined as

$$|v|_{l,D,\mathbb{R}^k} = (v,v)^{\frac{1}{2}}_{l,D,\mathbb{R}^k}$$

and the norm

$$\|v\|_{s,D,\mathbb{R}^k} = (\sum_{l\leq m} |v|^2_{l,D,\mathbb{R}^k})^{\frac{1}{2}}$$

by [39] we get the following minimization problem to find the function σ such that:

$$\begin{cases} \sigma \in Z, , \\ \forall u \in Z, J(\sigma) \leq J(u), \end{cases}$$

Where

$$J(u) = (\theta, (u,u)_{\mathbb{R}M_2}) + \varepsilon|u|^2_{s,D,\mathbb{R}^n}$$

and $\theta = (\theta_1, \ldots, \theta_{M_2})$, (with $\theta_i \geq 0$). The solution of this problem is called the interpolating variational spline in D, associated with $A, L, \alpha, \beta, \theta$ and ε

Lemma 1.7.2. *The application* $(((.,.)))\ : H^s(D;\mathbb{R}^n) \times H^s(D;\mathbb{R}^n) \longrightarrow \mathbb{R}$ *defined by*

$$(((v,u))) = \langle Lv, Lu\rangle_{M_1,n} + \langle \theta, \beta(v,u)\rangle_{\mathbb{R}^{M_2}} + \varepsilon(v,u)_{s,D,\mathbb{R}^n}$$

is an inner product in $H^s(D;\mathbb{R}^n)$ *and its related norm is* $[[u]] = (((u,u)))^{\frac{1}{2}}$ *is equivalent to the norm of the Sobolev space* $\|.\|_{s,D,\mathbb{R}^n}$.

Theorem 1.7.3. *This problem has a unique solution called the smoothing interpolating spline which is also the unique solution of the following variational problem:*

$$\begin{cases} \sigma \in Z,, \\ \\ \forall u \in Z_0,\ \ \langle \theta, \beta(\sigma,u)\rangle_{\mathbb{R}^{M_2}} + \varepsilon(\sigma,u)_{s,D,\mathbb{R}^n} = 0 \end{cases}$$

Theorem 1.7.4. *There exists a unique element* $(\sigma,\omega) \in H^s(D,\mathbb{R}^n)$ *that satisfy the following variational problem:*

$$\begin{cases} \sigma \in Z \ \ \textit{such that} \\ \\ \forall u \in H^s(D,\mathbb{R}^n,\ \langle \theta, \beta(\sigma,u)\rangle_{\mathbb{R}^{M_2}} + (\omega, Lu)_{\mathbb{R}^{M1}} = 0 \end{cases}$$

and σ *is the solution of the original problem.*

1.8 Multiobjective evolutionary algorithms

1.8.1 Basic definitions and ideas about multi-objective optimization problems

Multi-objective optimization problems (MOOP, also known as vector or Pareto optimization, multi-criteria or multiattribute programming, etc.) involve mathematical optimization problems where more than one objective functions has to be optimized simultaneously. They are very important in many

scientific, engineering and economic and logistic areas, where usually the optimal decisions need to be taken in the presence of conflicting, or even antagonistic, objectives. However, many applications that were traditionally been tackled using a single objective function where in origin a multiobjective one, that was reformulated assigning a numerical weigthed function to each original objective function in order to make it more tractable. But this simplified approach with these fixed weighting would assume that the preferences between objectives remain static, even before the knowledge of the range of possible solutions. Another characteristic that it is common to these multiobjective problems is the fact that we will need in general a diversity of posible solutions, representing a sufficient range of trade-offs between the objectives. For all these reasons, the optimization methods that simultaneously find a diverse set of high-quality solutions have been attracting more and more interest during last years.

We formulate the definition of a general multi-objective optimization problem by the following definitions [71].

Definition 1.8.1. *In general, a multi-dimensional optimization problem consists in searching for the minimum of a function* $f : \mathbb{R}^N \longrightarrow \mathbb{R}$ *such that there exists* $s \in \mathbb{R}^N$ *with* $f(s) \leq f(x)$ *for all* $x \in \mathbb{R}^N$*. The function* $f(x)$ *is called the fitness or cost function for that optimization problem.*

Definition 1.8.2. *A multi-objective optimization problem is a quadruple* (X, Y, f, ξ) *where* X *indicates the search or decision space,* Y *denotes the objective space, being* $f : X \longrightarrow Y$ *a function that assigns to each* $x \in X$ *a corresponding objective vector* $y = f(x) \in Y$*, and* ξ *is a binary relation over* Y *that characterizes a partial order on the objective space.*

So, the main idea is trying to minimize all the objective functions in Y at the same time to obtain the optimal solution. At the end, the goal is to get at least one solution that is one of the best, because not always it will be clear what is the "best" one, and we can have an entire set (called Pareto front) of possible best solutions, its necessary to define the dominance and pareto -optimality in general as the following definition [**?**].

Definition 1.8.3. *In general a multiobjective optimization problem is defined*

as:

$$min_{\mathbf{x}\in X}\mathbf{g}(\mathbf{x})$$

where $\mathbf{x} \equiv (x_1, \ldots x_n)$ *are the decision variables and*

$$\mathbf{y} = \mathbf{g}(\mathbf{x}) \equiv (g_1(x_1, \ldots x_n), \ldots g_m(x_1, \ldots x_n))$$

are considered the objective functions (vector or variables).

Definition 1.8.4. *Without loss of generality suppose the following multiobjective optimization problem with* n *decision variables and* m *objectives* y*:*

$$Minimize \;\; y = \mathbf{g}(\mathbf{x}) = (g_1(x_1, \ldots, x_n), \ldots, g_m(x_1, \ldots, x_n)), \; \mathbf{x} \in X$$

where X*,*Y *are also called parameter space and objective space, respectively.*

A decision vector $\alpha \in X$ *is called a dominante decision vector, if and only if :*

$$\forall i \in \{1, \ldots, m\} \quad g_i(\alpha) \geq g_i(\beta)$$

$$\exists j \in \{1, \ldots, m\} \quad g_i(\alpha) > g_i(\beta)$$

where $\beta \in X$*, and also*

$$\alpha \succeq \beta \;\; \textit{if and only if } \alpha \succ \beta \;\; \textit{or } g(\alpha) = g(\beta).$$

Definition 1.8.5. *Let* $\alpha \in X$ *be an arbitrary decision vector, then* α *is called nondominated regarding the set*

$$X\prime \subseteq X \;\; \textit{if and only if } \nexists \alpha\prime \in X\prime \; (\alpha\prime \neq \alpha) : \; \alpha\prime \succ \alpha.$$

Also α *is said to be Pareto-optimal, if and only if* α *is nondominated regarding the whole parameter space* X*.*

So, all nondominated solutions possess the attribute that their quality cannot be improved with respect to any of the objective functions without affecting detrimentally at least one of the others. Such optimal solutions

usually lie on the edge of the feasible regions of the so called search space and this is called the Pareto set (or front).

This Pareto front in many real situations may be even discontinuous or enclose a non-convex region of feasible solutions, but the Evolutionary Approach of the Multi-objective Optimization Procedures have proven good abilities to deal with these difficulties, coupled with their population based nature and depending of their implemented bias to preserve diverse sets of good solutions. Fo all these reasons, it is not surprising that this evolutionary methods are currently the state of the art in many multiobjective optimization problems.

1.8.2 Non-dominated Sorting Genetic Algorithms

The Non-dominated Sorting Genetic Algorithm is a Multiple Objective Optimization algorithm whose objective is to improve the adaptive fit of a population of candidate solutions to a Pareto front constrained by a set of objective functions.

Amongst the first algorithms to explicitly exert certain selection pressure towards the discovery of nondominated solutions, we could mention for example the Fonseca and Fleming's MOGA [22]; it uses fitness sharing amongst solutions of the same rank, coupled with fitness-proportionate selection to help promoting diversity.

The authors Srinivas and Deb's [66] original Nondominated Sorting Genetic Algorithm (NSGA) works in a similar way, but assigns fitness based on dividing the population into a certain number of fronts of equal domination. To achieve this, the algorithm iteratively seeks all the nondominated points in the population that still have not been labelled as belonging to the current front, and increments the front count, repeating until all solutions have been labelled finally. So, each point in a given front gets as its raw fitness the count of all solutions in inferior fronts. After this, Deb and his coworkers proposed a revised NSGA-II algorithm [17] which still uses the idea of non-dominated fronts, but also incorporates the following important changes:

- A crowding distance metric is defined for each point as the average side length of the cuboid defined by its nearest neighbours in the same front. The larger the value, the fewer solutions reside in the vicinity of the point.

- An appropriate survivor selection strategy, where the new population is obtained by accepting the individuals from pregressively inferior fronts

until it is full, so that if not all the individuals in the last front considered can be accepted, they are chosen on the basis of their crowding distance.

- Parent selection uses a modified tournament operator that considers first dominance rank, then crowding distance as the second criterium.

We can see then that for the approprite use of NSGA-II, it is necessary to describe these fundamental issues:

(1) Population initialization: the size of population depend on the problem range.

(2) Non-dominated sort : before talking about this stage we define
- p and P the individual and the main population respectively.
- S_p is a set of all individuals dominated by p.
- n_p equal the number of individuals dominated by p
- Q is the set for storing the individuals in the $(i+1)$-th. front.

We describe then the non-dominated procedure as follow :
$\star$ do the following

$$\forall\, p \in P$$

$$Initialize\ S_p = \emptyset.$$

$$Initialize\ n_p = 0.$$

$$\forall\, q \in P$$

$\cdot$ if p dominated q then $S_p = S_p \cup \{q\}$.
$\cdot$ else if q dominates p then $n_p = n_p + 1$

$$if n_p = 0 \ \text{ then } F_1 = F_1 \cup \{p\}$$

$\star$ Do this for all individuals.
$\star$ Begin initializing the new front from the first one with $i = 1$.
$\star$ next do that while $F_i \neq \emptyset$

$$\forall q \in S_p$$

$$n_q = n_q - 1\ then\ Q = Q \cup \{q\}$$

Augment the front counter by one.
At this stage $Q = F_i$

(3) Crowding distance : crowding distance is very important to compare between individuals and find the Euclidian distance between each individual in a front, so each individual have a crowding distance. The method of calculating the crowding distance is summarized as follow: For all front F_i, first initialize the distance $F_i(d_j) = 0$, and for each objective function m , $sort(F_i, m)$.
In order to calculate the distance for each individual inF_i
for k= 1 : (n-1)

$$I(d_k) = I(d_k) + \frac{I(k+1).m - I(k-1).m}{{f_m}^{max} - {f_m}^{min}}$$

(4) Selection: in order to select individuals we use a crowded comparison operator that depends on a binary tournament selection; the operation depends on F_i and p_{rank}, we see $p \prec_n q$ if

$$p_{rank} < q_{rank}$$

$$or \;\; if \; p \; and \; q \in F_i$$

$$then \quad F_i(d_p) > F_i(d_q)$$

(5) Genetic operator: now we describe the two parts of the simulated binary crossover, and that of polynomial mutation − Binary crossover : the following expressions represent the simulates binary crossover :

$$c_{1,k} = \frac{1}{2}[(1 - \beta_k)_{p_1,k} + (1 + \beta_k)_{p_2,k}]$$

$$c_{2,k} = \frac{1}{2}[(1 - \beta_k)_{p_1,k} + (1 + \beta_k)_{p_2,k}]$$

with $\beta \geq 0$, $p_{i,k}$ *selected parent, and* $c_{i,k}$ *represent to* i^{th} *child with* k^{th} *component*

But the density of any random sample that generated define by:

$$p(\beta) = \frac{1}{2}[(\eta_c + 1)\beta^{\eta_c}, \;\; if \; 0 \leq \beta \leq 1]$$

$$p(\beta) = \frac{1}{2}[(\eta_c + 1)\frac{1}{\beta^{\eta_c+2}}, \;\; if \; \beta > 1.]$$

where η_c defined as the distribution index of crossover. Let u be a uniform sample random number lie on $(0, 1)$, then we have:

$$\beta(u) = (2u)^{\frac{1}{\eta+1}}$$

$$\frac{1}{[2(1-u)]^{\frac{1}{\eta+1}}}$$

− Polynomial Mutation: first we define important thing as follow:
∘ c_k is a child.
∘ p_k is the parent.
∘ r_k is uniform sample random in$(0, 1)$.
∘ δ is mutation distribution index.
then the polynomialmutation is defined by :

$$c_k = p_k + ({p_k}^u - {p_k}^l)\delta_k$$

where

$$\delta_k = (2r_k)^{\frac{1}{\eta_m+1}} - 1, \;\; if \; r_k < 0$$

$$\delta_k = 1 - [2(1-r_k)]^{\frac{1}{\eta_m+1}}, \;\; if \; r_k \geq 0$$

(6) Recombination and Selection: The offspring population is combined with the current generation population and selection is performed to set the individuals of the next generation, and the elitism is ensured because the best individuals are added sequentially to the population. The selection of parents for the next generation is also based on their crowding distance, by selecting the individual at random, which specify by tournament size, then choosing the best individual out of that set to be a parent.

Chapter 2

Evolutionary computation of optimal knots allocation in smoothing splines of two variables

2.1 Introduction

Function approximation, and curve/surface fitting to data, are major and important problems in many applied technical and scientific fields: as in CAD design, virtual reality and computer animation, data visualization of geological and medical imaging, artificial vision or scientific computing in general [54], [76]. On the other hand, there are several different methodologies that have been used for selecting knots for best approximating or fitting curves/surfaces to data using B-splines. For instance, the methodology in [42] uses a least squares fitting method with B-spline surfaces, depending on a sensitive parametrization, in connection with a uniform distribution of knots. In the last decades, an increasing number of works and algorithms have been developed for this problem of approximating or fitting a surface to data using splines. For example, the authors in [72] developed a genetic algorithm to optimize both the number and the allocation of knots, in spline

approximation of bio-medical data. Later, a new approach was introduced to improve the knot adjustment for B-spline curves, by using an appropriate elitist clonal selection algorithm [24]. The methodology in [82] describe an iterative algorithm, that can also use heuristic criteria, for the optimal knots distribution in B-spline surface fitting. Kozera et al. in [40] gave a new method for choosing unknown knots that minimize the cost function, by using a special scale. Also, the development of surface fitting include several types of splines, for example some techniques without grids for medical image registration, by modification of several selected knots in P-splines [56]. The method in [77] describes the unified averaging technique, for both approximation and interpolation problems, concerning knots placement for B-spline curves. In [25] is presented a method to solve the curve fitting problem, using a hierarchical genetic algorithm (HGA). A computational methodology, based on the so called Artificial Immune Systems (AIS) paradigm, is used to solve the knots' adjustment problem for B-spline curves. The goal of the application of this algorithm is to determine the location of knots automatically and efficiently. In [73] the authors describe a powerful methodology, based on simulated annealing (SA), for optimal placement of the knots. Also the authors in [35] use sparse optimization techniques for this purpose. However, as pointed out in [71] and many other related works, a final and definitive methodology for optimal placement and selection of knots, for approximating/fitting curves or surfaces to data, using smoothing splines, is not still completely investigated, even in the one dimensional case.

In this chapter we propose a new methodology for optimal placement of random knots, for approximating/fitting curves or surfaces to data, using cubic smoothing splines of one or two independent variables. A new technique is presented to optimize both the number of knots and its optimal placement by cubic spline approximations, using a developed multi-objective genetic algorithm. Our scheme give very accurate results, even for curves or surfaces with discontinuities and/or cusps; it also determines the optimal number of knots automatically.

This chapter is organized as follows: after this Introduction, Section 2.2 presents the basic definitions and concepts related with cubic and bicubic spline spaces, necessary for the corresponding methodology used in the sequel for the mathematical formulated problem (see Subsection 2.2.3). Concerning the associated multi-objective optimization problems, see in Section 2.3 the proposed optimization strategy for knots' placement in the cases of cubic and bicubic smoothing splines; in Section 2.4 we show some results, both in one (see Subsection 2.4.1) and two variables (see Subsection 2.4.2) that confirm the performance of the proposed methodology; and some final conclusions are drawn in Section 2.5.

2.2 Proposed Methodology

2.2.1 Introduction.

In this section we extend the classical cubic B-splines of class $\mathcal{C}^2$ in one independent variable, to the bicubic B-splines of class $\mathcal{C}^2$ of two independent variables. For this, let $a, b, c, d \in \mathbb{R}$ and consider $R = [a, b] \times [c, d]$.

2.2.2 Bicubic spline spaces of class $\mathcal{C}^2$.

Bicubic splines of two variables are an extension of cubic splines in one variable. We start with two partitions or knot sequences of $[a, b]$ in m subintervals and $[c, d]$ in n subintervals; i.e. increasing sequences, that can be uniform or not.

Let denote $\triangle_m$ and $\triangle_n$ the two partitions of $[a, b]$ and $[c, d]$ respectively, where $\triangle_m$ and $\triangle_n$ are respectively

$$\triangle_m \equiv \{a = x_0 < x_1 < ... < x_m = b\}$$

$$\triangle_n \equiv \{c = y_0 < y_1 < ... < y_n = d\}.$$

We define the bicubic spline functions

$$S : [a, b] \times [c, d] \longrightarrow \mathbb{R}$$

such that :

i) $S \in \mathcal{C}^2([a, b] \times [c, d])$,

ii) $S\,|_{[x_i,x_{i+1}]} \times_{[y_j,y_{j+1}]} \in \mathbb{P}_3([x_i, x_{i+1}] \times [y_j, y_{j+1}])$,
$i = 0, ..., m-1, \quad j = 0, ..., n-1$,

where $\mathbb{P}_3([x_i, x_{i+1}] \times [y_j, y_{j+1}])$ is the space of all restrictions of two-variable polynomials of partial degree less than or equal to three on the rectangle $[x_i, x_{i+1}] \times [y_j, y_{j+1}]$.

Given $x_{-3}, x_{-2}, x_{-1}, x_{m+1}, x_{m+2}, x_{m+3}$ and $y_{-3}, y_{-2}, y_{-1}, y_{n+1}, y_{n+2}, y_{n+3}$ real numbers, such that,

$$x_{-3} \leq x_{-2} \leq x_{-1} \leq a < b \leq x_{m+1} \leq x_{m+2} \leq x_{m+3},$$

$$y_{-3} \leq y_{-2} \leq y_{-1} \leq c < d \leq y_{n+1} \leq y_{n+2} \leq y_{n+3},$$

then

$$B_i^0(x) = \begin{cases} 1, & x_{i-3} \leq x < x_{i-2} \\ 0, & otherwise \end{cases}, \; i = 0, \ldots m+5 \qquad (2.1)$$

and $B_i^k(x), \quad k = 1, 2, 3, ...$ can be defined from the recursive relation, for $i = 0, \ldots, m + 5 - k$:

$$B_i^k(x) = \frac{x - x_{i-3}}{x_{i+k-3} - x_{i-3}} B_i^{k-1}(x) + \frac{x_{i+k-2} - x}{x_{i+k-2} - x_{i-2}} B_{i+1}^{k-1}(x). \tag{2.2}$$

These functions verify the following properties:

i) they are positive in the interior of their support,

$$B_i^k(x) \geq 0, \ \forall x \in [a, b];$$

ii) they form a partition of unity,

$$\sum_{i=0}^{m+5-k} B_i^k(x) = 1, \ \forall x \in [a, b];$$

iii) $\left\{B_0^k, \ldots, B_{m+5-k}^k\right\}$ are linearly independent for all $k = 0, 1, 2, 3 \ldots$.

We consider the analogous definitions (2.1,2.2), in the variable y, of the functions $\left\{B_0^k, \ldots, B_{n+5-k}^k\right\}$ for the partition in $[c, d]$.

Meanwhile, if $\mathcal{S}_3(\triangle_m \times \triangle_n)$ represents the set of bi-cubic spline functions of degree less than or equal to three and class $\mathcal{C}^2$, then dim $\mathcal{S}_3(\triangle_m \times \triangle_n) = (m+3)(n+3)$; and if $\left\{B_0^3(x), ..., B_{m+2}^3(x)\right\}$ and $\left\{B_0^3(y), ..., B_{n+2}^3(y)\right\}$ are the basis of $\mathcal{S}_3(\triangle_m)$ and $\mathcal{S}_3(\triangle_n)$ respectively, then a basis of $\mathcal{S}_3(\triangle_m \times \triangle_n)$ will be

$$\left\{B_q^3(x, y) \equiv B_i^3(x) B_j^3(y), \begin{smallmatrix} i=0,...,m+2, \\ j=0,...,n+2, \\ q=(m+3)j+i+1 \end{smallmatrix}\right\}.$$

2.2.3 Formulation of the problem

The main goal of this section is to solve the following problem: Given the data sets

$$\{u_l : l = 1, ..., N\} \subset R \subset \mathbb{R}^2$$

and

$$\{\beta_l : l = 1, ..., N\} \subset \mathbb{R},$$

find $S \in \mathcal{S}_3(\triangle_m \times \triangle_n)$ such that

$$S(u_l) \approx \beta_l, \quad \forall l = 1, \ldots, N$$

i.e. S approximates the data set

$$\{(u_l, \beta_l) \in R \times \mathbb{R}\}_{l=1}^{N}.$$

We define the smoothing variational bicubic spline associated with $U = \{u_l\}_{l=1,...,N} \subset R$, $\{\beta_l\}_{l=1,...,N} \subset \mathbb{R}$, and $\varepsilon \in (0, \infty)$ as the unique $S \in \mathcal{S}_3(\triangle_m \times \triangle_n)$ minimizing the functional

$$J : \mathcal{S}_3(\triangle_m \times \triangle_n) \longrightarrow [0, \infty)$$

defined by

$$J(v) = \sum_{l=1}^{N} (\beta_l - v(u_l))^2 \tag{2.3}$$

$$+\varepsilon \sum_{|\alpha|=2} \int_a^b \int_c^d (\partial^\alpha v(x,y)\,)^2 dxdy \tag{2.4}$$

where $|\alpha| = \alpha_1 + \alpha_2, \forall \alpha = (\alpha_1, \alpha_2) \in \mathbb{N}^2$, and $\partial^\alpha v(x,y) \equiv \dfrac{\partial^{|\alpha|} v}{\partial x^{\alpha_1} \partial y^{\alpha_2}}$.

This problem has a unique solution (see [37]) that can be expressed as a linear combination of our bi-cubic basis functions

$$S(x,y) \equiv \sum_{q=1}^{(m+3)(n+3)} \alpha_q B_q^3(x,y), \;\; (x,y) \in R.$$

where $\alpha = (\alpha_1, ..., \alpha_{(m+3)(n+3)})$ is the unique solution of the following linear system

$$(AA^\top + \varepsilon R)\alpha^\top = A\beta^\top$$

being $A \in \mathbb{R}^{(m+3)(n+3),N}$ and $R \in \mathbb{R}^{(m+3)(n+3),(m+3)(n+3)}$

$$A = \left(B_q^3(x_l, y_l)\right)_{\substack{q=1,...,(m+3)(m+3) \\ l=1,...,N}},$$

$$R = \left(\sum_{|\alpha|=2} \int_a^b \int_c^d \partial^\alpha B_r^3(x,y)\, \partial^\alpha B_s^3(x,y)\, dx\, dy \right)_{r,s}.$$

The corresponding formulation in the one variable case is just evident and may be written easily from this general formulation in two variables. It is also well known, and may be consulted in [71] and the references therein.

2.3 Optimization strategy for knots' placement in the cases of cubic and bicubic smoothing splines

In order to verify the ability of the new multi-objective strategy for the determination of the knots placement for the proposed bicubic smoothing splines, we will use at least two approximate discretizations of normalized error estimations, in $\mathcal{C}(R)$ and $\mathcal{L}^2(R)$ for example, given by the expressions:

$$E_c \equiv \frac{\max\limits_{i=1,\ldots,M} |f(a_i) - S(a_i)|}{\max\limits_{i=1,\ldots,M} |f(a_i)|} \tag{2.5}$$

$$E_l \equiv \sqrt{\frac{\sum\limits_{i=1}^{M} (f(a_i) - S(a_i))^2}{\sum\limits_{i=1}^{M} (f(a_i))^2}} \tag{2.6}$$

where $f \in \mathcal{C}^2(R)$ is the function we want to approximate, $S \in \mathcal{S}_3(\triangle_m \times \triangle_n)$ is the smoothing variational bicubic spline associated with the given data sets and $\{a_1, \ldots, a_M\} \subset R$ is a given random point set where the errors will be computed. In the case of functions of only one variable (or 2D point sets), the expressions are totally equivalent, changing the rectangle R by the corresponding interval of real numbers.

We will be using a Non-dominated Sorting Genetic Algorithm (NSGA); that is, a Multiple Objective Optimization Genetic Algorithm (MOGA) whose objective is to improve the adaptive fit of a population of candidate solutions to a Pareto front, constrained by a set of objective functions. In this paper the algorithm developed to obtain optimal knots placement of B-spline basis functions uses an evolutionary methodology, with replacement and the usual operators, including: selection, genetic crossover, and mutation.

For the correct use of the NSGA-II algorithm (an important improvement of the original NSGA one, see [17]) it is necessary to describe three fundamental issues:

(i) Solution Representation: The chosen representation characterizes each individual of interest to be evaluated. The representation determines how the problem is structured in the NSGA-II and also determines its behavior.

(ii) Genetic Operators: We must define some important parameters before running any MOGA algorithm. The important agents in this part are [17]:

(a) Number of Generations: We can use 5 or more generations, but in this work we usually take 20 to 40 generations as stopping criteria of the algorithm, depending on the case.

(b) Population size: This number determines the number of possible good candidates to be taken into account in all the genetic process. Depending on the problem, we are using a population size of about 20 or more members.

(c) Selection function: The selection function chooses the individuals, called parents, that contribute to the population at the next generation.

(d) Crossover and mutation functions: in this algorithm, we use binary crossover function which combines two parents to form children for the next generation; the crossover fraction is 0.9 and use polynomial mutation.

(e) Pareto fraction: It keeps the others than most fit population down to the specified fraction in order to maintain a diversity of the population; a value of 0.4 is of common use in this case.

(iii) Fitness function: different forms can be used in a NSGA-II algorithm; but the main goal is to minimize the errors E_c and E_l, between the original function and the smoothing variational bicubic spline constructed from each population of random knots.

2.4 Simulation examples

The objective of this study is to analyze the performance of the explained MOGA procedure for the optimization of the knots placement for the construction of B-spline basis functions in order to obtain an optimal approximating cubic or bicubic spline.

Different experiments have been carried out, both with functions of one and two variables. In this section, we will show the results of approximations for each of these functions, and we present the evolution of the optimal distribution of knots, together with the related Pareto fronts, using the proposed methodology.

2.4.1 Examples of one independent variable

The examples below are mainly chosen from the existent literature on the subject in the 1D case: example 1 coincide with the example 3 in [71], and the examples 3 and 4 have been also extracted from the simulations included in the papers [75] and [46] , respectively.

Example 1: $f_1 : [0,1] \longrightarrow \mathbb{R}$

$$f_1(x) = 90/(1 + e^{100(x-0.4)}) \ .$$

Example 2: $f_2 : [0,\pi] \longrightarrow \mathbb{R}$

$$f_2(x) = 0.12 + 0.25e^{-4(x-\frac{\pi}{4})^2}\cos(2x)\sin(2\pi x) \ .$$

Example 3: $f_3 : [0,4\pi] \longrightarrow \mathbb{R}$

$$f_3(x) = 0.2e^{-.05x}\sin(5x) + 0.4 \ .$$

Example 4: $f_4 : [-1,1] \longrightarrow \mathbb{R}$

$$f_4(x) = 1 - e^{-50|x|} \ .$$

To test the behavior of the presented methodology for the optimization of knots placement of the B-spline basis in this case, all the procedure have been also adapted to this one-dimensional case. Not a large number of knots have been necessary to obtain sufficiently good results, as you can see in the corresponding figures.

We can also see clearly, in the right column of Figure 2.4.1, how the evolution of the knots' distribution tends to be located in the region where the function $f_1(x)$ change the most within its domain. The left column also shows the results of the approximating function, compared with the original one, using this B-spline function approximation. The corresponding Pareto front, taking also into account one of the errors in the approximations vs. the number of knots used, is shown in Figure 2.4.2.

We can view clearly in all these cases that our results are quite better than the corresponding results included in the specified articles, with less interior knots; and always obtaining lower errors when using the same number of them or less number of generations.

Now we show in Figure 2.4.5 our results (with discrete errors of orders of 10^{-4} and 10^{-5} in any case) for the Example 3, also used as test example in [75].

And finally, in Figure 2.4.6 some graphical and numerical result for the Example 4, also considered in a classical paper [46] about this subject.

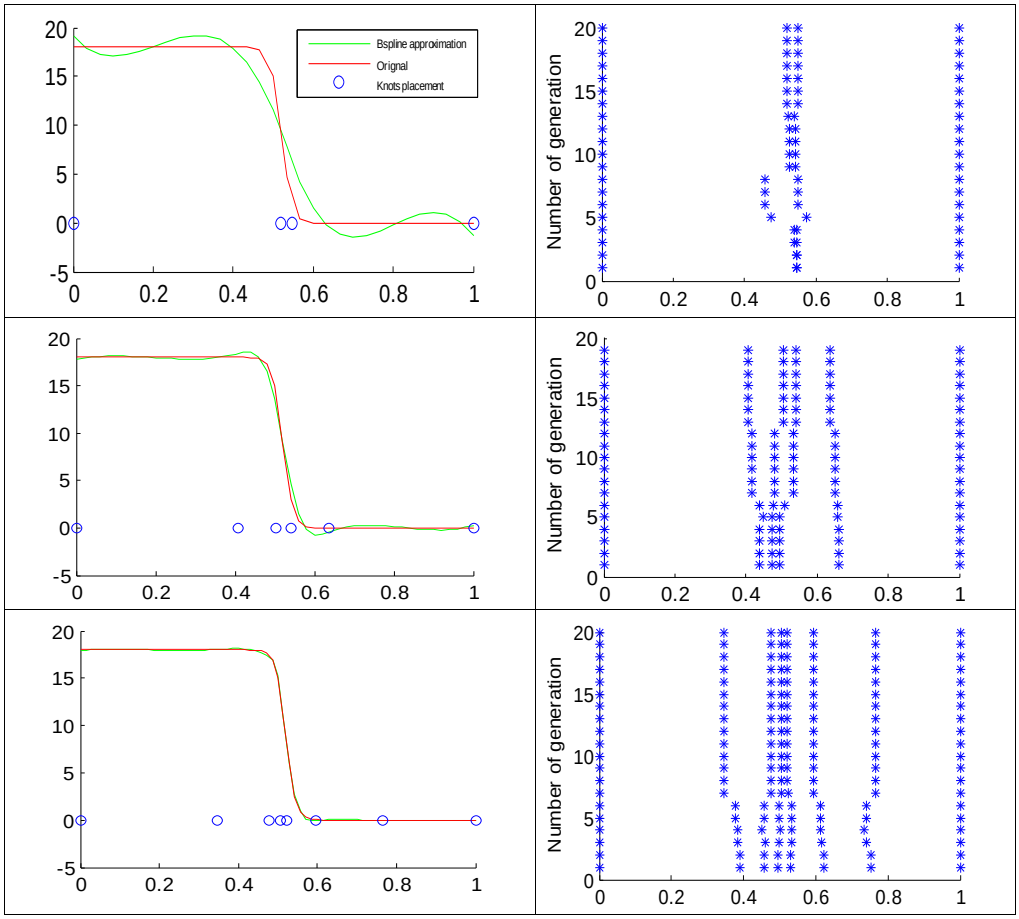

Figure 2.4.1: Approximating spline vs. actual function $f_1(x)$ (left) at final generation and distribution of knots in each generation (right) for 2, 4 and 6 interior knots respectively

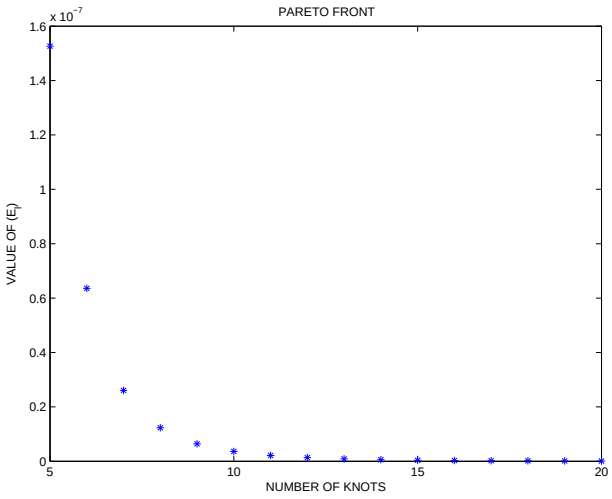

Figure 2.4.2: Pareto front of E_l error vs. number of knots for $f_1(x)$.

2.4.2 Examples of functions of two independent variables

We begin with a quite simple example of a paraboloid of revolution in the following example

Example 5: $F_5 : [0,1] \times [0,1] \longrightarrow \mathbb{R}$

$$F_5(x,y) = x^2 + y^2 \ .$$

In order to analyze the behavior of the B-spline approximation in these cases, not too many knots will be used (just 10 to 20 knots in each variable interval). In Figure 2.4.7 it can be seen some particular results of the approximation of function $F_5(x,y)$ using bi-cubic smoothing variational splines.

Similar results can be obtained for a more complicated Example 6, where the well-known Franke's function is considered. Graphics in Figure 2.4.9 show the results of the approximation with Bi-cubic smoothing variational splines in this case.

Example 6: $F_6 : [0,1] \times [0,1] \longrightarrow \mathbb{R}$

$$\begin{aligned} F_6(x,y) := \ & \tfrac{3}{4} e^{-((9x-2)^2+(9y-2)^2)/4} \\ & + \tfrac{3}{4} e^{-((9x+1)^2/49-(9y+1)/10)} \\ & + \tfrac{1}{2} e^{-((9x-7)^2+(9y-3)^2)/4} \\ & - \tfrac{1}{5} e^{-((9x-4)^2+(9y-7)^2)} \end{aligned} \ .$$

Figures 2.4.8 and 2.4.10 represent the Pareto fronts for the application of this MOGA procedure to these last two examples. It is clear that the errors

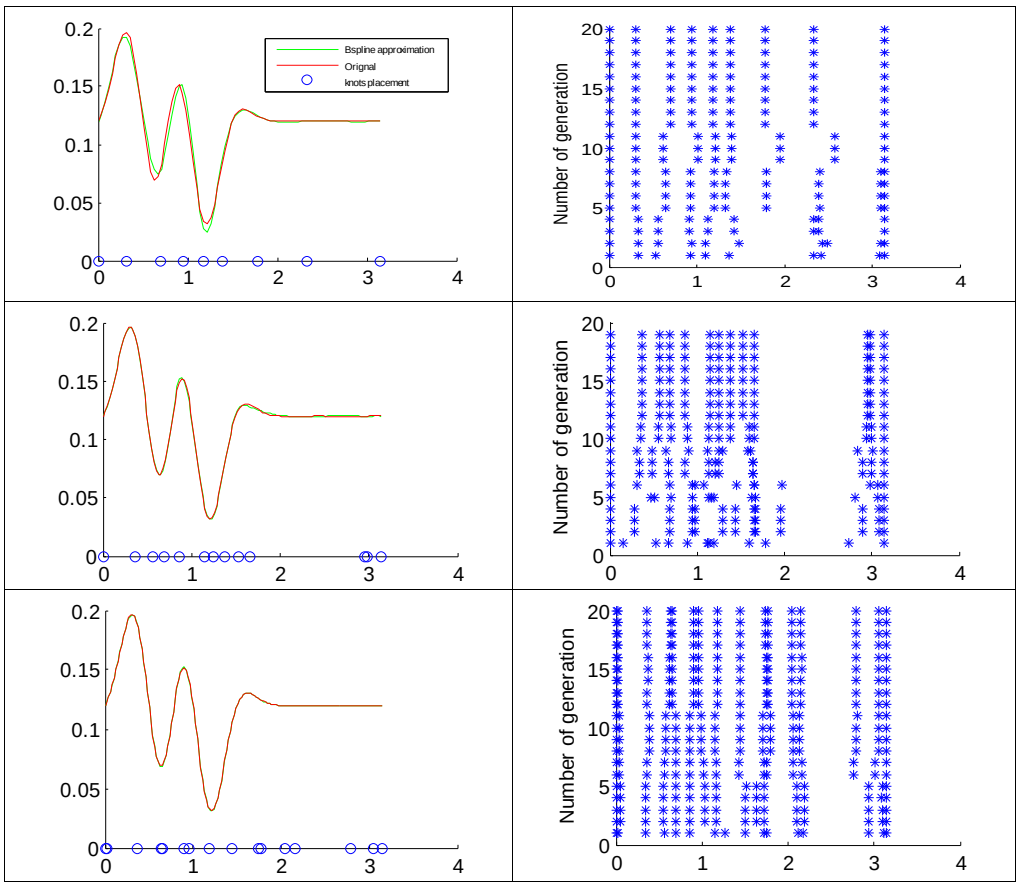

Figure 2.4.3: Approximating spline vs. actual function $f_2(x)$ (left) at final generation and distribution of knots in each generation (right) for 8, 11 and 14 interior knots respectively

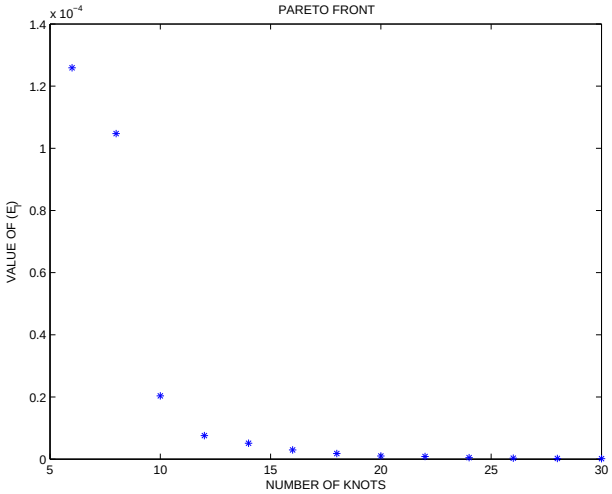

Figure 2.4.4: Pareto front of E_l error vs. number of knots for $f_2(x)$.

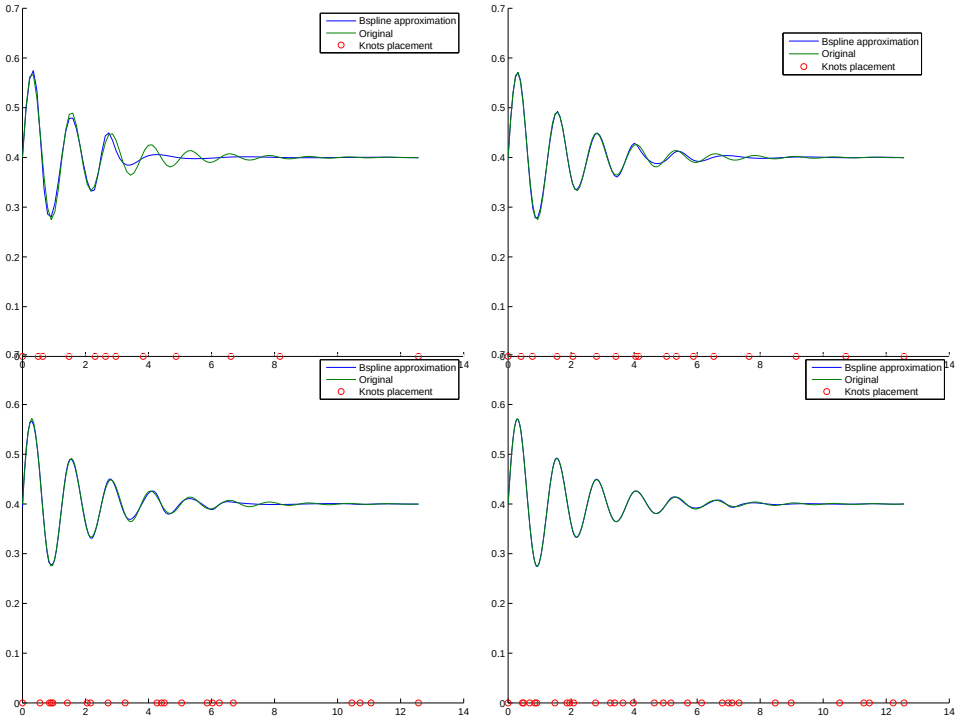

Figure 2.4.5: Approximating splines vs. actual function $f_3(x)$ at final generation for 10, 15, 20 and 30 interior knots (marked on the x-axis) respectively.

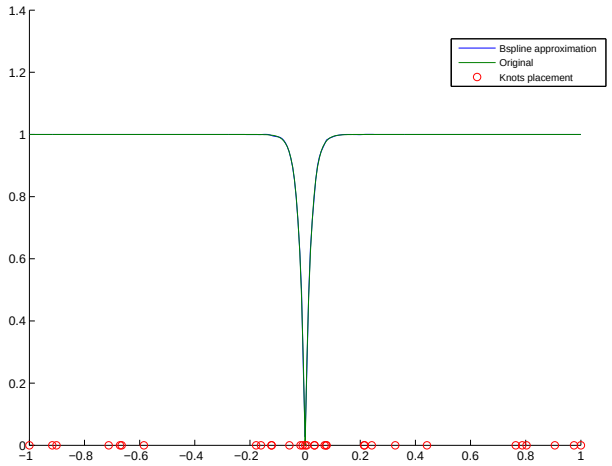

Figure 2.4.6: Graphical results for $f_4(x)$ with 29 interior knots (marked on the x-axis) and a discretized error of order 10^{-4}

with the MOGA tends to be reduced when the number of knots to construct the smoothing B-spline basis increases, but also increases (significantly in this case) the corresponding computational effort.

2.5 Conclusions

In this paper, a novel evolutionary MOGA methodology is presented for knots placement in the problem of cubic spline approximation of functions of one or two variables, showing the effectiveness of the strategy for different types of functions.

So, the goal of using a MOGA for placement of the knots in such cases of approximating functions by cubic or bi-cubic splines, of one or two variables, respectively, can be summarized as follows:

(1) It has been sufficiently proven that the placement of the knots in spline approximation has an important and considerable effect on the behavior of the final results, but the optimal placement or location of knots is not known a priori.

(2) The number of knots to be used in classical approaches, should be selected a priori by the designer; but using MOGA, a Pareto front for

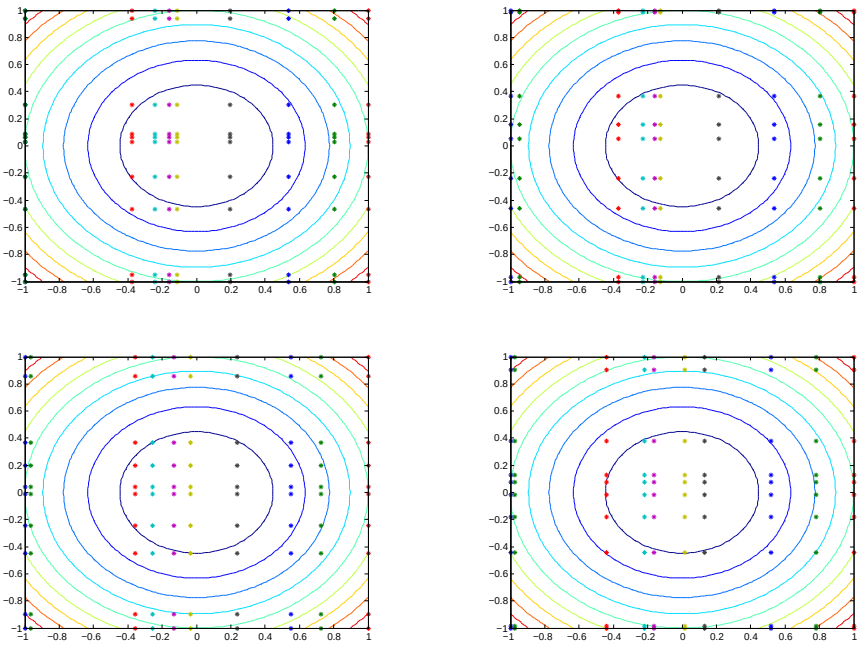

Figure 2.4.7: From top to down and left to right, evolution of the knots distribution every 5 generations for the function $F_5(x, y)$.

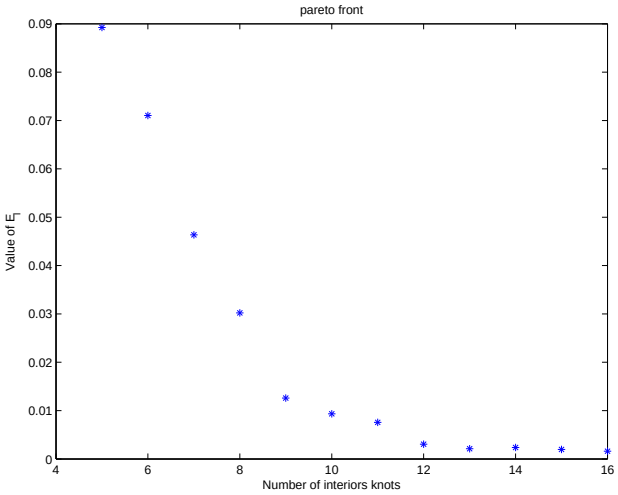

Figure 2.4.8: Pareto front of E_l error vs. number of knots for $F_4(x)$.

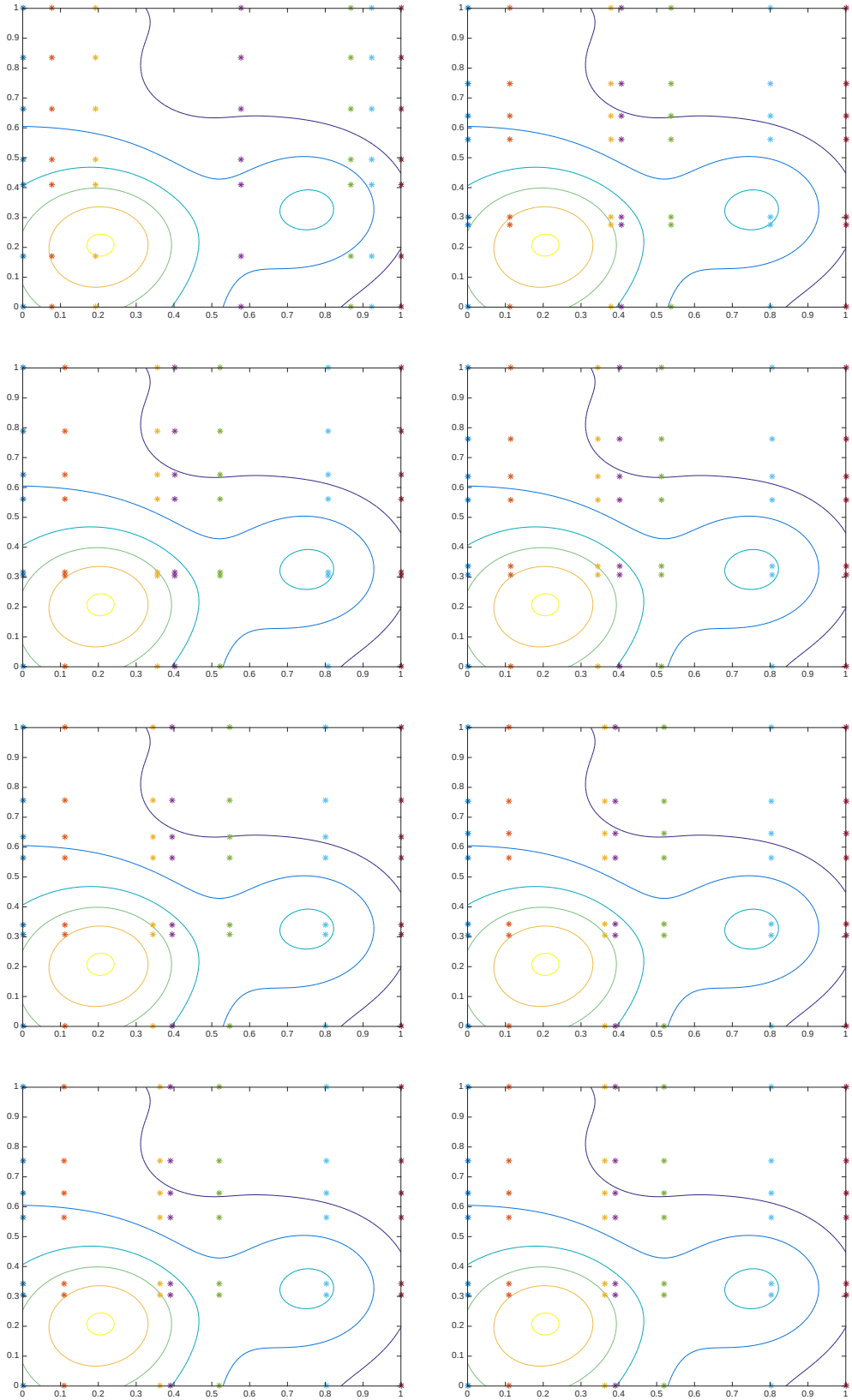

Figure 2.4.9: From top to down and left to right, evolution of the knots distribution every 5 generations for the function $F_6(x, y)$.

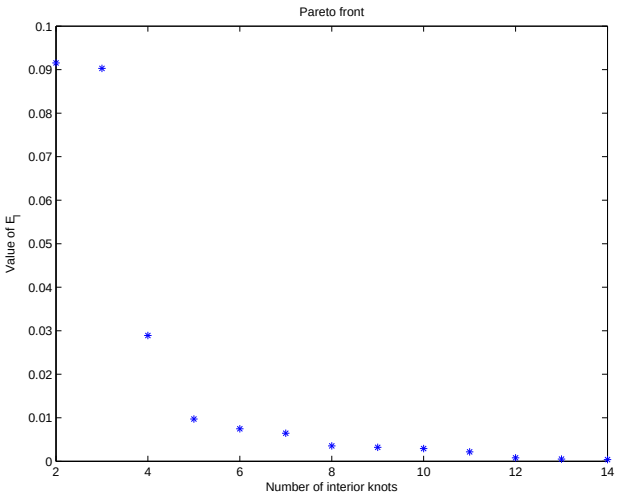

Figure 2.4.10: Pareto front of E_l error vs. number of knots for $F_6(x)$.

different or variable number of knots used can also be directly optimized.

Many simulations have been carried out, showing that increasing the number of knots in the definition of the cubic, or bi-cubic B-splines, also increases the accuracy of the approximation, but only up to a certain limit, where the corresponding computation effort is not worth enough comparing with the small possible reduction in the corresponding errors. So we can affirm without any doubt, that this proposed procedure have much better performance and results than such as the selected knots within an equally spaced distribution of them, and many other classical approaches.

Chapter 3

Optimal knots allocation in the bicubic spline interpolation problem

3.1 Introduction

Interpolation and approximation are very important problems, in Mathematics and also in many applied fields. For example, they are the key technology in every reverse engineering procedure, systematic function or data analysis, signal processing, data representation, stockage and/or compression, etc. In our case, the main aim is to obtain a structured and suitable digital representation of curves and surfaces, related to some environmental and/or natural resources practical problem, by using cubic or bicubic splines, respectively.

There exist several methodologies for approximating or interpolating a surface or some given data. In any case, multiple authors, like those in [42, 45, 79] describe the important and significant effect of the placement of the knots in spline approximation or interpolation on the performance of the final results.

Many methodologies have been presented for the selection and optimization of parameters within B-splines, using techniques based on selecting some special knots, called dominant points [51, 52], or by a data selection process as in [53]. Also the methodology in [24] gave a technique for automatic

knots modification using an elitist clonal selection algorithm. Other procedures using some least squares methods use uniform knots distributions, in connection with a possible sensitive parametrization [42]. In this sense it is worth to take into account the works [45] and [79].

In [40] the authors present a method for selecting the unknown knots by minimizing a certain cost function, and this is one of the key ingredients in the implementation of the so-called Genetic Algorithms (GA). Other methodologies, as for example that in [25], describe techniques for using a computational Artificial Immune System (AIS), based on a hierarchical paradigm structure to determine the number and knots location automatically. AIS are adaptive computational algorithms inspired by theories and models of real biological inmune systems when applied to mathematical or engineering problem solving. Some key aspects on the designing of AIS are:

- The chosen internal representation
- Mechanisms to evaluate the immune interactions
- The adaptation processes and mechanisms

Following the ideas of Man et al. in [49] also the computational representation of the chromosomes in a GA can also be arranged in more complex hierarchical structure, to emulate the formation of the biological DNA in living beings. In this way, the computational chromosomes of these Hierarchical Genetic Algorithms (HGAs) consists of two types of genes, known as *control* and *parametric* genes, that typically are encoded as binary digits and real numbers (or any other particular adequate representation), respectively. Both types of genes can be simultaneously optimized using this HGAs but in general are much more complicated to be implemented.

Authors in [31] pointed out a different approach for curve and surface fitting using implicit polynomials, where an adaptive heuristic finding algorithm allows to optimize the final degree of these polynomials. In [82] also a heuristic algorithm is presented to approximate scattered data with a B-spline surface. But there are many other related works, for instance the method in [35] optimizes the location and the number of knots in curve fitting with splines by a sparse optimization model; and in [74], a fast and computationally efficient methodology for optimal placement of the knots by using simulated annealing (SA) is also presented. In this stochastic procedure, a combination of hill-climbing and random walks is followed in order to simulate a well known physical process in tempered metallurgy. So, SA is a descent algorithm modified by random ascent moves in order to escape local

minima which are not global ones. Other more or less "intelligent" Teaching and Learning Based Optimization (TLBO) [58] and self-adaptive multi-population based Jaya algorithms [59] for Engineering problems have also been developed more recently, but we do not find them specially well-suited for our particular problem, taking also into account the more specialized implementation and applications that inspired them.

You can see [17] and the references therein for more details about the ideas concerning multi-objective optimization problems and the fundamental issues for GAs in general and NSGA-II algorithm in particular.

The first GAs were used mainly to optimize single cost (or mono-objective) function problems; but soon, the necessity of solving also several or many cost (or multi-objective) optimization ones was completely compulsory, due to the great diversity of real-life problems in such situation. In fact, the first multi-objective GA was proposed by Schaffer [67] called Vector Evaluated Genetic Algorithm (VEGA). Afterward, several important Multi-Objective Evolutionary or Genetic Algorithms (MOGA) were developed (see for example [22], Niched Evolutionary Algorithms [41], Random Weighted Genetic Algorithms (RWGA) [78] and many others, that can be consulted in the website [15]. But one of the most reliable and succesful estrategies is the application of the Goldberg's notion of *nondominated sorting* in the GA along with a niche and speciation method to find multiple Pareto-optimal points simultaneously [66]. Also appropriate extensions and improvements are accomplished in the NSGA-II and NSGA-III versions (see [33] for a performance comparison between these two versions). The key points concerning these two new versions of the original one are:

- A fast *nondominated sorting* procedure where he population is sorted into a hierarchy of sub-populations based on the ordering of Pareto dominance. Similarity between members of each sub-group is evaluated on the Pareto front, and the resulting groups and similarity measures are used to promote a diverse front of non-dominated solutions.

- The appropriate consideration of the *elitism approach* in order to enhance the convergence properties of the algorithm.

- A *parameterless niching operator, in order to maintain certain level of multiple solutions.*

More recently, a unified approach for single, multiple and many objective optimization algorithm have been proposed and analyzed in [64]. In this unified version of these algorithms the authors also show how it works with usual Evolutionary Algorithms (EAs) parameters and no additional tunable

ones are needed. They also emphasize that certain MOGA procedures, as the NSGA-II, runs very well for bi-objective problems but does not scale up to solve many-objective optimization problems efficiently, although they can work quite well solving mono-objective ones. On the other hand, for three and many-objective problems, the more recent NSGA-III algorithm is preferred, but they propose a single unified efficient procedure capable of handling one to many-objective problems without having to reimplement or change the optimization algorithm if we have to deal with different objective dimensions of the original problem.

In this chapter, a new multi-objective NSGA-II methodology for optimal placement of random knots, when interpolating a function of one or two variables, using cubic or bicubic splines, is developed. But, we have to emphasize that, although some of the previous more basic evolutionary algorithms presented have been applied in the existing literature to the knots placement in the fitting or approximating 1D problem, we have not found references where the interpolating case is also treated, and neither the 2D approximating or interpolating cases. We also have to mention that in most of the cases, the GAs encountered in the literature were applied to a certain combinatorial binary problem of selecting the knots among a particular uniform partition of the interval, whereas we are aplying a more sophisticated NSGA-II MOGA with real representation of the chromosomes, in such a way that these interpolation knots/points can freely move, or even be deleted when necessary, until a certain threshold of errors and number of knots is accomplished. So, this is the main originality of the present work, together with the ability to treat with functions of two variables as well.

3.2 Background material

3.2.1 Cubic interpolation splines in one variable

In this subsection, just to fix the notation used in the sequel of the article, the usual cubic splines will be presented for this interpolation problem. In principle, we denote $\mathcal{S}_3(\triangle_n)$ the space of cubic spline functions of degree less than or equal to three and class $\mathcal{C}^2$, on the partition of $[a, b]$ in n subintervals, from an increasing sequence of (uniform or non uniform) points or knots $\triangle_n = \{a = x_0 < ... < x_n = b\}$, then $dim\, \mathcal{S}_3(\triangle_n) = n + 3$.

If now we have:

$x_{-3} \leq x_{-2} \leq x_{-1} \leq x_0 = a < \cdots < x_n = b \leq x_{n+1} \leq x_{n+2} \leq x_{n+3}$,

we can define for each $x \in [a, b]$

$$B_i^0(x) = \begin{cases} 1, & if \quad x_{i-3} \leq x < x_{i-2}, \\ 0, & otherwise \end{cases}, \ for \quad i = 0, \ldots, n+5$$

and $B_i^k(x)$, $k = 1, 2, 3$, from the very well known recursive relation:

$$B_i^k(x) = \frac{x - x_{i-3}}{x_{i+k-3} - x_{i-3}} B_i^{k-1}(x) + \frac{x_{i+k-2} - x}{x_{i+k-2} - x_{i-2}} B_{i+1}^{k-1}(x), \ i = 0, \ldots, n+5-k.$$

so that $\left\{B_0^k, \cdots, B_{n+5-k}^k\right\}$ are linearly independent for all $k = 0, 1, 2, 3$ and have nice support and positivity properties (see for example [**?**, 68]) and a basis of $\mathcal{S}_3(\triangle_n)$ would be $\left\{B_0^3, \cdots, B_{n+2}^3\right\}$, called B-spline basis functions of fourth order, and degree three.

Now, given $u_i \in \mathbb{R}$ (for $i = 0, \ldots, n$) and $\{a = x_0 < ... < x_n = b\}$, we want to obtain the usual natural cubic spline (with vanishing second derivatives at the endpoints of the interval $[a, b]$) interpolating these values at these specified knots, but using now these B-spline basis functions of $\mathcal{S}_3(\triangle_n)$, $\left\{{B_i}^3\right\}_{i=0,\ldots,n+2}$ instead of the usual piecewise representation on each one of the subintervals. Thus $s : [a, b] \to \mathbb{R}$ such that $s(x_i) = u_i, i = 0, \ldots, n$ can be expressed as

$$s(x) = \sum_{i=0}^{n+2} \alpha_i B_i^3(x), \ x \in [a, b], \tag{3.1}$$

where $\alpha \equiv (\alpha_0, \ldots, \alpha_{n+2})$ is the solution of the linear system $A\alpha = B$ obtained under the following conditions

1. $s(x_i) = u_i, \quad i = 0, ..., n,$
2. $s''(x_0) = 0 = s''(x_n)$;

where the matrix $A = \begin{pmatrix} A_1 \\ \hline A_2 \end{pmatrix} \in \mathbb{R}^{(n+3)\times(n+3)}$ and the vector $B = \begin{pmatrix} b_1 \\ \hline b_2 \end{pmatrix} \in \mathbb{R}^{n+3}$ are composed of two submatrices and subvectors, respectively:

$$A_1 = ({B_j}^3(x_i))_{\substack{0 \leq i \leq n \\ 0 \leq j \leq (n+2)}} \in \mathbb{R}^{(n+1)\times(n+3)}$$

$$A_2 = \begin{pmatrix} (B_j^3)''(x_0) \\ (B_j^3)''(x_n) \end{pmatrix}_{0 \leq j \leq n+2} \in \mathbb{R}^{2\times(n+3)}$$

$$b_1 = (u_i)_{0 \leq i \leq n} \in \mathbb{R}^{n+1}, \quad b_2 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \mathbb{R}^2.$$

3.2.2 Bicubic interpolating splines of class $\mathcal{C}^2$.

We will also extend the classical cubic univariate B-splines to bicubic B-splines in two variables. Let $R = (a,b)\times(c,d) \subset \mathbb{R}^2$ be a rectangular open set. We start with two partitions with knot sequences $\triangle_n$ of $[a,b]$ in n subintervals, and $\triangle_m$ of $[c,d]$ in m subintervals: $\triangle_n = \{a = x_0 < x_1 < \ldots < x_n = b\}$ and $\triangle_m = \{c = y_0 < y_1 < \ldots < y_m = d\}$; then, $\triangle_n \times \triangle_m$ is a grid partition of R, and we define the bicubic spline of class $\mathcal{C}^2$ on the partition $\triangle_n \times \triangle_m$ as every function $S : R \longrightarrow \mathbb{R}$ such that

i) $S \in \mathcal{C}^2(R)$

ii) $S\,|_{[x_i,x_{i+1}]\times[y_j,y_{j+1}]} \in \mathbb{P}_3([x_i,x_{i+1}]\times[y_j,y_{j+1}])$, for every $i = 0,\ldots,n-1, j = 0,\ldots,m-1$

where $\mathbb{P}_3([x_i,x_{i+1}]\times[y_j,y_{j+1}])$, $i = 0,\ldots,n-1, j = 0,\ldots,m-1$, is the space of all restrictions of polynomial functions of two variables and partial degree less than or equal to three to the sub-rectangle $[x_i,x_{i+1}]\times[y_j,y_{j+1}]$. Given
$x_{-3}, x_{-2}, x_{-1}, x_{n+1}, x_{n+2}, x_{n+3} \in \mathbb{R}$ and $y_{-3}, y_{-2}, y_{-1}, y_{m+1}, y_{m+2}, y_{m+3} \in \mathbb{R}$, such that, $x_{-3} \le x_{-2} \le x_{-1} \le x_0 < \cdots < x_n \le x_{n+1} \le x_{n+2} \le x_{n+3}$, and $y_{-3} \le y_{-2} \le y_{-1} \le y_0 < \cdots < y_m \le y_{m+1} \le y_{m+2} \le y_{m+3}$, we can construct, as presented in the previous subsection, the corresponding cubic splines of class $\mathcal{C}^2$ in each one of the variables, x and y, on the partitions $\triangle_n$ and $\triangle_m$ respectively.

Meanwhile, if $\mathcal{S}_3(\triangle_n \times \triangle_m)$ represents the set of bicubic spline functions of degree less than or equal to three and class $\mathcal{C}^2$ in both variables, then $dim\,\mathcal{S}_3(\triangle_n \times \triangle_m) = (n+3)\times(m+3)$ and if the basis of $\mathcal{S}_3(\triangle_n)$ and $\mathcal{S}_3(\triangle_m)$ are denoted by $\left\{B_0^3(x),...,B_{n+2}^3(x)\right\}$ and $\left\{B_0^3(y),...,B_{m+2}^3(y)\right\}$, respectively, then a basis of $\mathcal{S}_3(\triangle_n \times \triangle_m)$ will be

$$\left\{B_q^3(x,y) \equiv B_i^3(x)B_j^3(y), \begin{array}{c} i=0,\ldots,n+2, \\ j=0,\ldots,m+2, \\ q=(n+3)j+i+1 \end{array}\right\}.$$

Now, given a function f, well defined and smooth enough (at least of class $\mathcal{C}^1$); in this section we deal with the following problem: Find another smooth function (mainly a bivariate polynomial or spline) $S : R \longrightarrow \mathbb{R}$ to interpolate the following $N \equiv (n+1)\times(m+1)$ points in $3D$:

$$\{(x_i, y_j, u_l) : i = 0,...,n, j = 0,...,m, l = (n+1)j + i + 1\}. \tag{3.2}$$

where we will also denote $U \equiv \{u_l = f(x_i,y_j)\}_{l=1,\ldots,N} \subset \mathbb{R}$

In fact here, we want to obtain $S \in \mathcal{S}_3(\triangle_n \times \triangle_m)$ such that, we can obtain for every $i = 0, \ldots, n-1$, $j = 0, \ldots, m-1$

$$S(x_i, y_j) = u_l, \ l = (n+1)j + i + 1.$$

Thus, we would write

$$S(x, y) = \sum_{k=1}^{(n+3)(m+3)} \alpha_k B_k(x, y), \ \forall (x, y) \in R, \tag{3.3}$$

where $\alpha \equiv (\alpha_1, \ldots, \alpha_{(n+3)(m+3)})$ is the solution of the linear system $A\alpha = B$ obtained under the following conditions

1) $S(x_i, y_j) = u_l, \ i = 0, ..., n, \ j = 0, ...m, \ l = (n+1)j + i + 1,$

2) $\dfrac{\partial^2 S}{\partial y^2}(x_i, c) = 0 = \dfrac{\partial^2 S}{\partial y^2}(x_i, d), \quad i = 0, ..., n,$

3) $\dfrac{\partial^2 S}{\partial x^2}(a, y_j) = 0 = \dfrac{\partial^2 S}{\partial x^2}(b, y_j), \quad j = 0, ..., m,$

4) $\dfrac{\partial^4 S}{\partial x^2 \partial y^2}(a, c) = \dfrac{\partial^4 S}{\partial x^2 \partial y^2}(a, d) = \dfrac{\partial^4 S}{\partial x^2 \partial y^2}(b, c) = \dfrac{\partial^4 S}{\partial x^2 \partial y^2}(b, d) = 0,$

Where $A = (\dfrac{A_1}{A_2})$,
$A_1 = (B_k(d_l)), \quad l = 1, ..., (n+1)(m+1), \quad k = 1, ..., (n+3)(m+3)$, and
$d_l = (x_i, y_j), \quad i = 0, ..., n, \quad j = 0, ..., m, \quad l = (n+1)j + i + 1,$

$$(A_2)_{\hat{l}k} = \begin{cases} \dfrac{\partial^2}{\partial y^2} B_k(x_i, c) \\ \dfrac{\partial^2}{\partial y^2} B_k(x_i, d) \\ \dfrac{\partial^2}{\partial x^2} B_k(a, y_j) \\ \dfrac{\partial^2}{\partial x^2} B_k(b, y_j) \\ \dfrac{\partial^4}{\partial x^2 \partial y^2} B_k(a, c) \\ \dfrac{\partial^4}{\partial x^2 \partial y^2} B_k(a, d) \\ \dfrac{\partial^4}{\partial x^2 \partial y^2} B_k(b, c) \\ \dfrac{\partial^4}{\partial x^2 \partial y^2} B_k(b, d) \end{cases} \qquad i = 0, \cdots, n, \ j = 0, \cdots, m,$$

where $\hat{l} = 1, ..., 2(n+1) + 2(m+1) + 4, \quad k = 1, ..., (n+3)(m+3)$, and $B = (\dfrac{b_1}{b_2})$, $b_1 = (u_l), \quad l = 1, ..., (n+1)(m+1)$, and $b_2 = (0) \in \mathbb{R}^K$, with $K = (n+3)(m+3) - (n+1)(m+1)$.

3.2.3 Optimization paradigm for placement of knots in cubic splines interpolation.

At this stage, it is important to clarify more in detail this methodology of using MOGA as an optimization strategy for the determination of the knots placement for bicubic interpolation splines. It is well-known that many mathematical and realistic models can be formulated as multi-objective optimization problems and that customized genetic algorithms (GAs) have been demonstrated to be particularly effective obtaining very satisfactory solutions to these type of problems, where usually there is not a unique *best* solution. The concept of GA was introduced in the last sixties and fully developed by

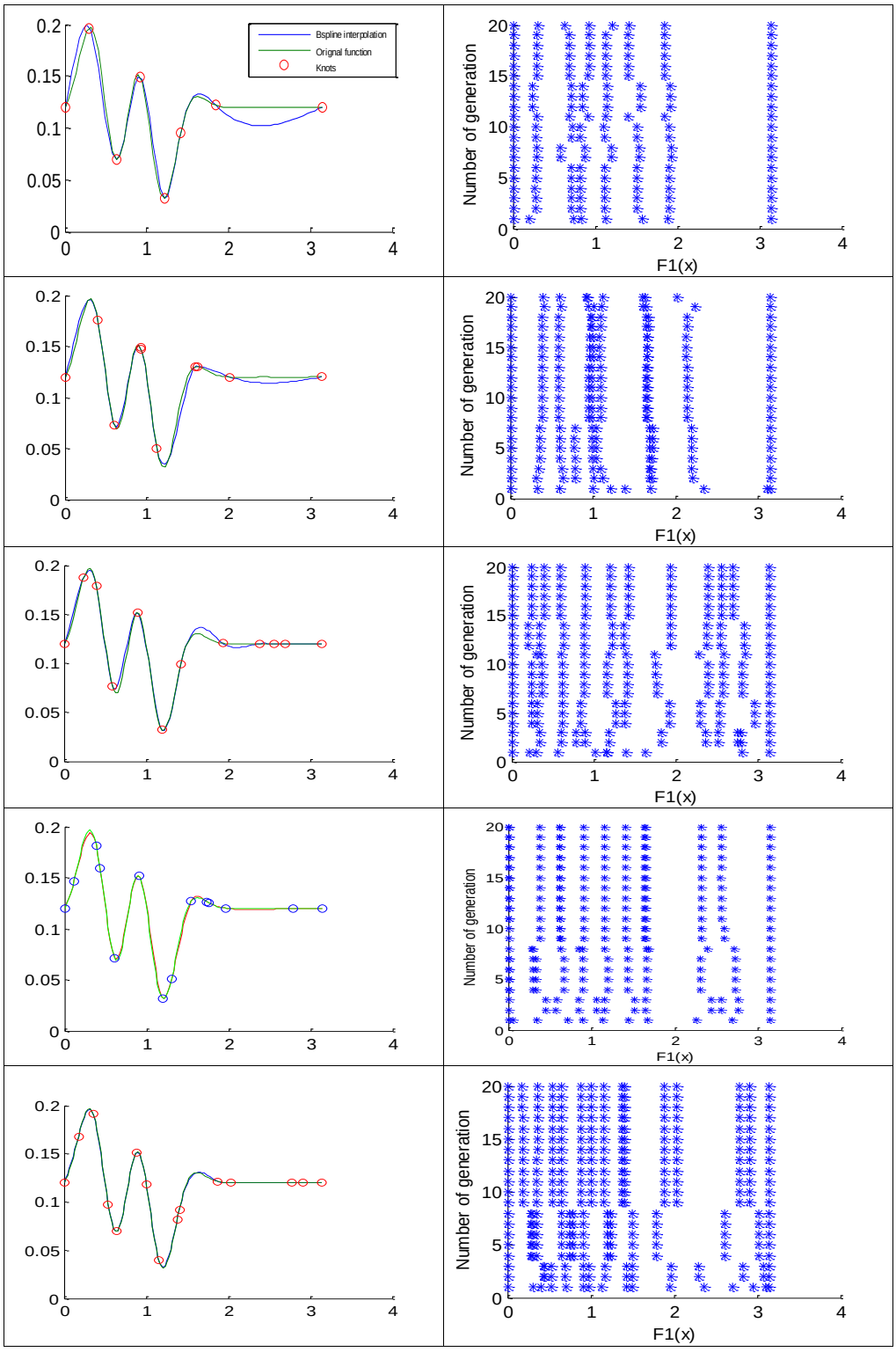

Figure 3.2.1: We can see in the left column the cubic interpolating splines corresponding to the last knots' distribution, whose evolution is also showed in the right column, with increasing number of interior knots for the function $F_1(x)$.

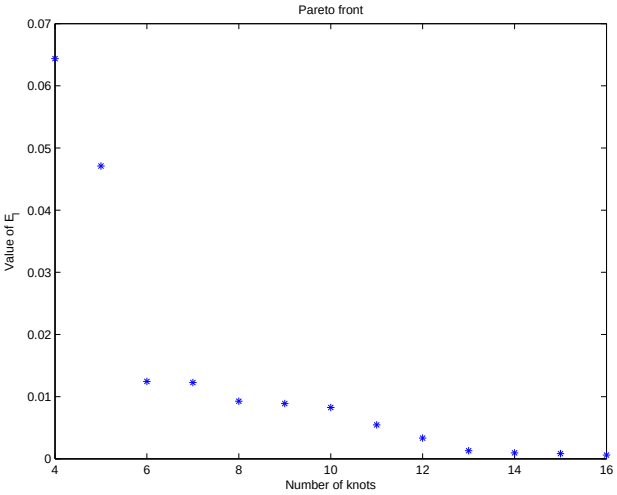

Figure 3.2.2: Pareto front of E_l error vs. number of knots for $F_1(x)$.

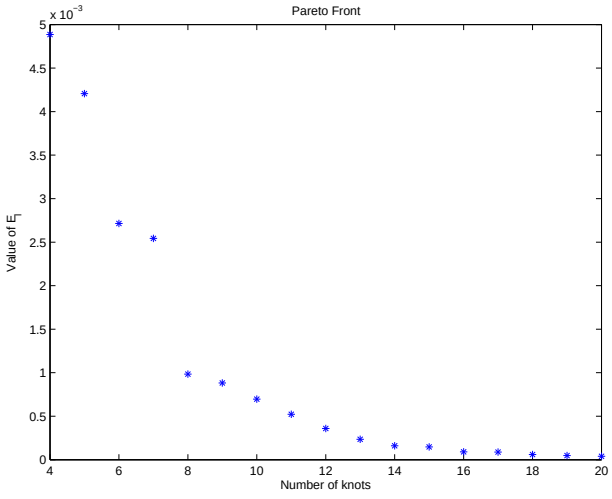

Figure 3.2.3: Pareto front of E_l error vs. number of knots for $F_2(x)$.

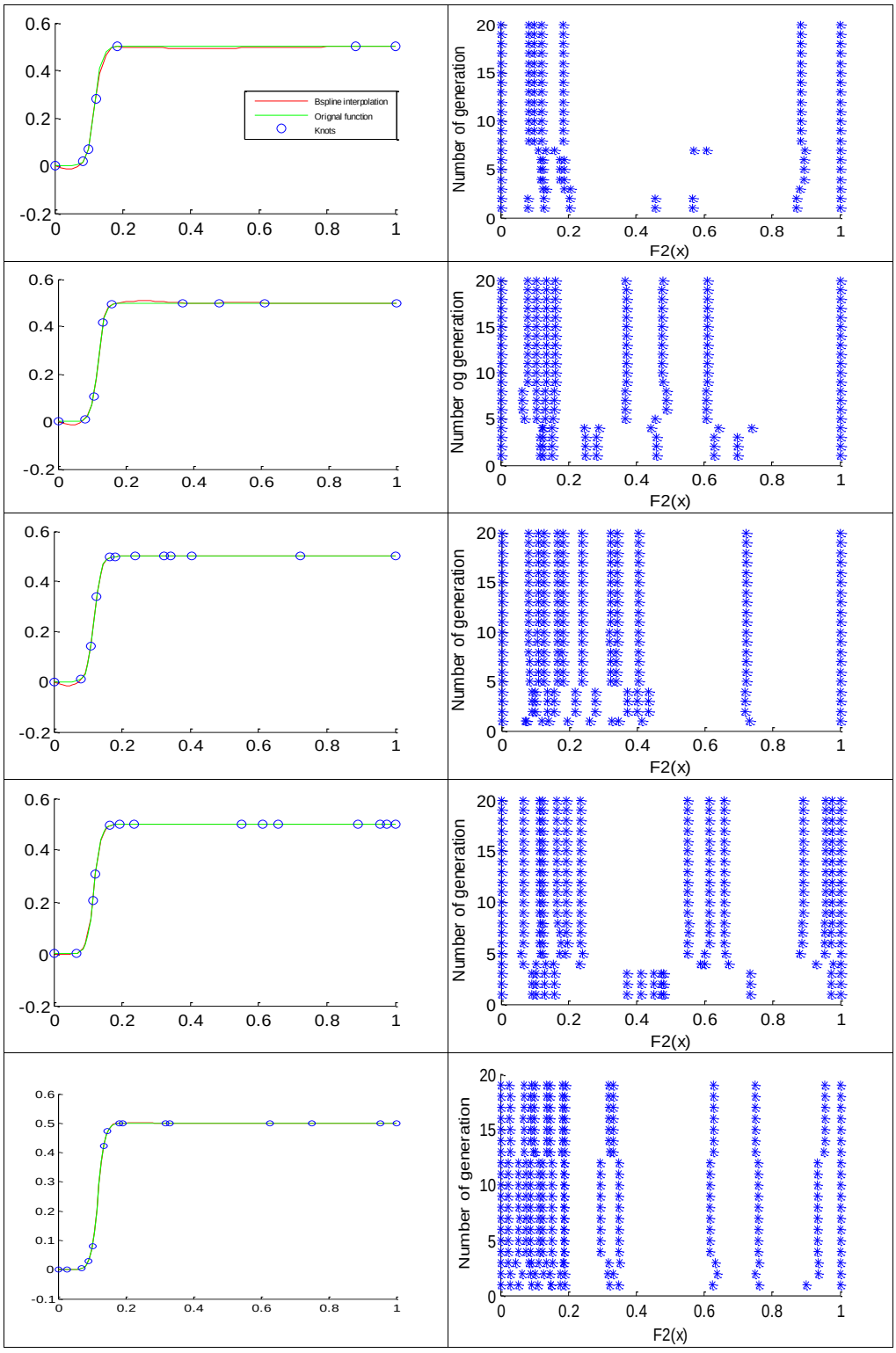

Figure 3.2.4: We can see in the left column the cubic interpolating splines corresponding to the last knots' distribution, whose evolution is also showed in the right column, with increasing number of interior knots for the function $F_2(x)$.

Holland [28] and many others during several decades and is fully inspired in the usual selection process of most adapted individuals in natural evolution of species. GA operates then with a collection (called the *population*) of randomly chosen *individuals*, with their particular *chromosomes*, that can be appropriately combined (using the so called *crossover* operator) and/or mutated (using the corresponding *mutation* operator). So the parents must to be adequately chosen in order that their corresponding chromosomes would be combined to produce *better* offsprings (taking into account the *objective/cost* functions considered). Also the mutation operator may introduce some particular random little changes into these chromosomes, although the *mutation rate* (probability of changing the properties of a gene) is usually quite small. In this way, a slow process of change/evolution it is also implemented so that unsuccessful changes are automatically eliminated by this "natural" selection process, while good or adapted changes will prevail.

A simplified pseudocode of the NSGA-II algorithm, well suited for continuous function multiple objective optimization problems is the following (see for example [7], [17]):

```
Input: PopulationSize, ProblemSize, P_crossover, P_mutation
Output: Children

Population <- InitializePopulation(PopulationSize, ProblemSize)
EvaluateAgainstObjectiveFunctions(Population)
FastNondominatedSort(Population)
Selected <- SelectParentsByRank(Population, PopulationSize)
Children <- CrossoverAndMutation(Selected, P_crossover,P_mutation)

While (StopCondition())
    EvaluateAgainstObjectiveFunctions(Children)
    Union   <- Merge(Population, Children)
    Fronts  <- FastNondominatedSort(Union)
    Parents <- EmptySet
```

```
    Front_L <- EmptySet

    For ( Front_i \in Fronts)
        CrowdingDistanceAssignment(Front_i)
        If (Size(Parents)+Size(Front_i) > PopulationSize)
            Front_L <- i
            Break()
        Else
            Parents Merge(Parents, Front_i)
        End
    End

    If (Size(Parents) < PopulationSize)
        Front_L <- SortByRankAndDistance(Front_L)
        For (P_1 To PopulationSize-SizeFront_L)
            Parents <-  P_i
        End
    End

    Selected   <- SelectParentsByRankAndDistance(Parents, PopulationSize)
    Population <- Children
    Children   <- CrossoverAndMutation(Selected, P_crossover, P_mutation)
End

Return (Children)
```

As it is also well known, both the classical NSGA and the more efficient widely accepted version NSGA-II (Non-dominated Sorting Genetic Al-

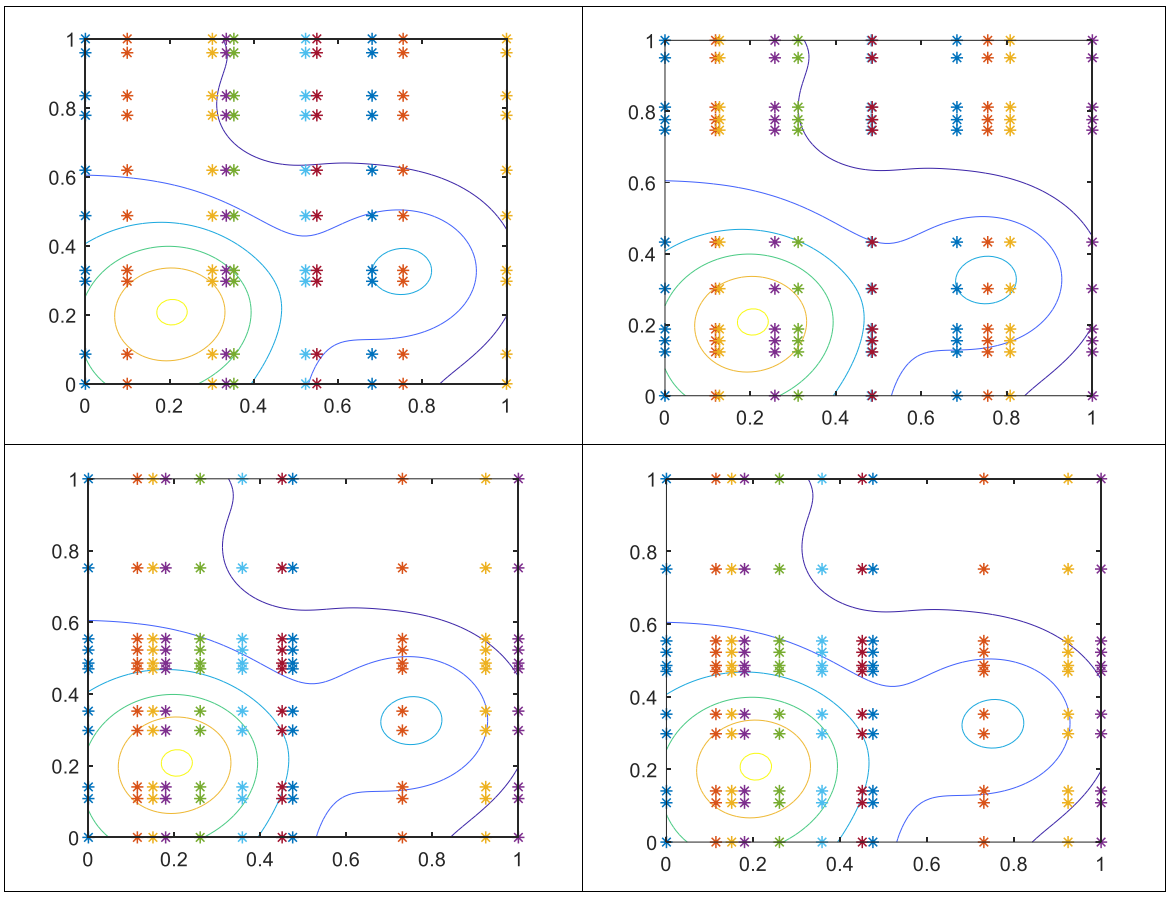

Figure 3.2.5: From left to right and top to down we show the evolution of the distribution of 11 points on each axis for $F_3(x, y)$.

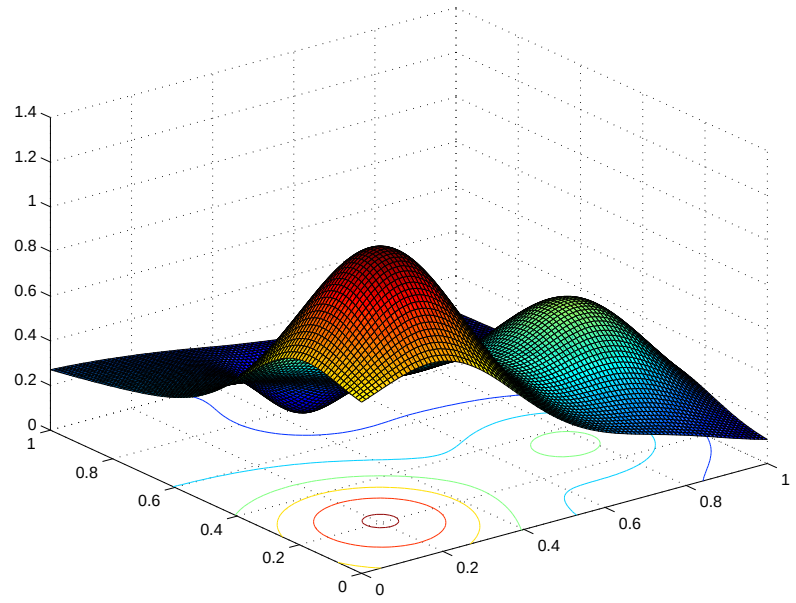

Figure 3.2.6: 3D graphic and iso-lines of the Franke's function $F_3(x, y)$.

gorithms) are both MOAs and EAs for Multiple Objective Optimization functions. So, the principal objective of these methods is always to improve the adaptive fit of a population of candidate solutions to a Pareto front constrained by these, often conflicting objectives. In our case, these objective functions are to minimize several normalized discrete versions of some approximate errors, as explained below. But also the intention to maintain the number of knots as small as possible will be taken into account, so that we allow the procedure to delete some of the obtained knots, when they become sufficiently close to each other and the same level of approximation can be retained.

Both, the classical NSGA and the updated and improved NSGA-II version always emphasizes non-dominated solutions, making a non-dominated sorting with the combination of parent and offspring population and classifying them in different fronts according to an ascending level of non-domination, but the main differences are that this last one:

- uses an elitist principle, so that there are some elites in the population that are given more opportunities to have offsprings, and even be carried out to the next generation,

- uses explicit diversity preserving mechanism, taking into account a denominated *crowding distance*,

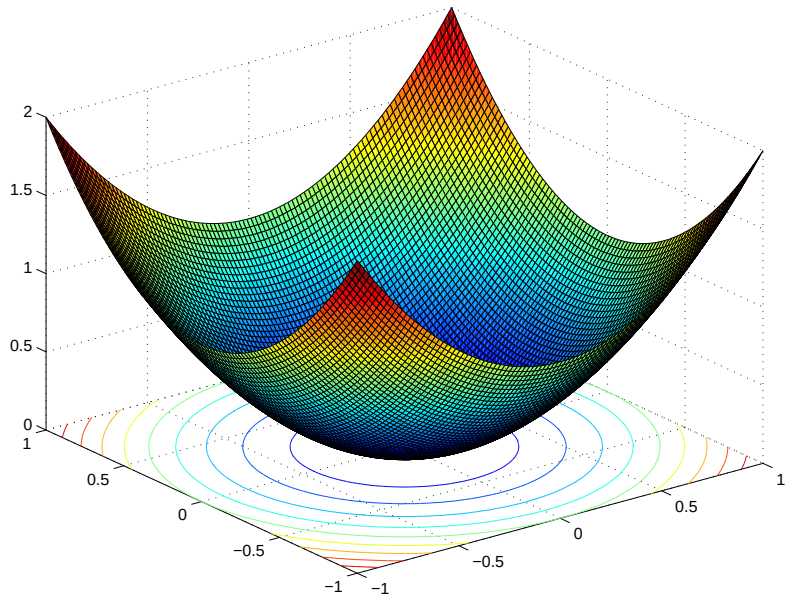

Figure 3.2.7: 3D graphic and iso-lines of the $F_4(x, y)$ function.

- avoid some inherent time complexity and necessary sharing parameter in the previous classical version.

In order to verify the ability of generalization of the new multi-objective strategy for the determination of the knots placement for the proposed interpolating bicubic splines, a test data set (TDS) is used and denoted by $X^{Test} = [(x_1, y_1)^{Test}, ..., (x_{ntest}, y_{ntest})^{Test}]$, composed of the nodes and its corresponding output data set: $TDS = [X^{Test}; Z^{Test}]$. With the data of TDS, it is possible to obtain the output data approximation using the presented interpolating bicubic splines methodology. This output set is termed as $\hat{Z}^{Test}$, being $\hat{Z}^{Test} = [\hat{z}_1^{Test}, ..., \hat{z}_{ntest}^{Test}]$, with $ntest$ being the number of nodes in the set X^{Test}.

There are some important parameters that should be defined before running any MOGA algorithm. The most important parameters for the evolutionary strategy are presented in Table 3.2.1, but usually are problem-dependent and have to be chosen empirically.

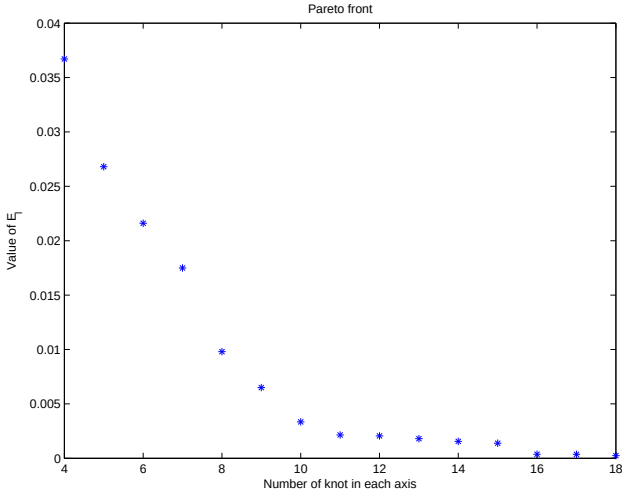

Figure 3.2.8: Pareto Front of E_l error vs. number of knots for $F_3(x)$.

Different forms of fitness, or objective functions, can be used in a NSGA-II procedure, but the main goal is to minimize some of the usual errors between the original function and the interpolating bicubic spline constructed from each population of random knots. We consider two approximation error estimations that are appropriate normalizations of the discrete version of the usual norms in $\mathcal{C}(R)$ and $\mathcal{L}^2(R)$, and are given by the expressions:

$$E_c = \frac{\max\limits_{i=1,\ldots,M} |f(a_i) - S(a_i)|}{\max\limits_{i=1,\ldots,M} |f(a_i)|} \tag{3.4}$$

$$E_l = \sqrt{\frac{\sum\limits_{i=1}^{M} (f(a_i) - S(a_i))^2}{\sum\limits_{i=1}^{M} (f(a_i))^2}} \tag{3.5}$$

where $f \in \mathcal{C}^2(R)$ is a given function, $S \in \mathcal{S}_3(\triangle_n \times \triangle_m)$ is the interpolating bicubic spline associated with the given data set (3.2), and $\{a_1, \ldots, a_M\} \subset R$ is a given scattered random point set where the errors are computed. In the case of functions of only one variable, the expressions are totally equivalent, but the points will be chosen inside the corresponding interval.

Also, in this case of interpolation, we need to consider some deleting or

Table 3.2.1: Parameters and functions used by MOGA in the simulations.

Parameters of the MOGA	Value
Number of generations	20
Population size	40
Crossover function	Binary crossover
Selection function	Binary tournament
Crossover fraction	0.9
Pareto fraction	0.4
Mutation function	polynomial mutation
Mutation rate	0.01
Fitnes functions	E_c and E_l
Knots' deletion tolerance	$0.3 * 10^{-2}$

colliding procedure, in order to can remove some of the interpolation knots when they come too close to each other, making instable the associated interpolation procedure (because the matrix of the corresponding linear system become almost singular). So, a certain tolerance parameter is also introduced, in this interpolation case, in order to avoid the possible instabilities and/or bad conditioning of the matrices involved in this problem. This issue could also be avoided if we choose the interpolation points independently of the knots, but this is not the case here, and could be the subject of a much more general procedure, where some Shoenberg-Whitney conditions would also must be taken into consideration [65].

3.3 Simulation results

To study the behavior of the approximation for the presented methodology, performed by optimization of the knots placement of bicubic spline functions by MOGA, different experiments have been carried out. In order to perform the interpolation, using the proposed methodology, we present the most important parameters for the evolutionary strategy in Table 3.2.1, and the following functions are used: Example 1: $F_1 : [0, \pi] \longrightarrow \mathbb{R}$

$$F_1(x) = 0.12 + .25 \exp^{-4(x-\frac{\pi}{4})^2} \cos(2x) \sin(2\pi x) \ .$$

We can see clearly in Figure 3.2.1 that the evolution of the distribution of knots is located where the function $F_1(x)$ change the most within its

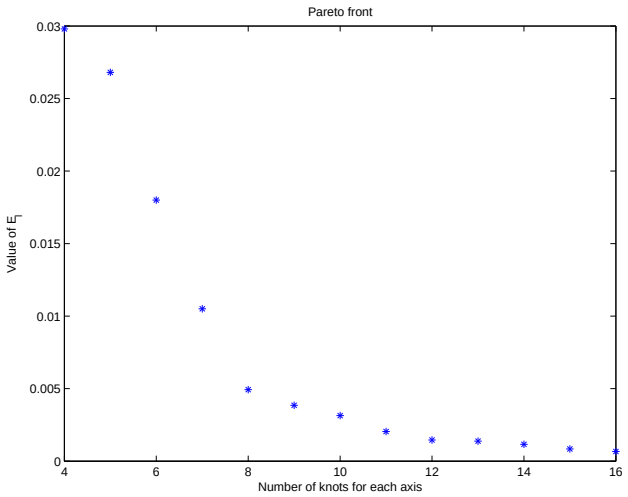

Figure 3.3.9: Pareto front of E_l error vs. number of knots for $F_4(x, y)$.

domain, and shows the results of the approximating function compared with the original one using B-spline function interpolation. The corresponding Pareto front, taking also into account the error in the interpolation vs. the number of knots used is shown in Figure 3.2.2. Similar results (see Figures 3.2.4 and 3.2.3) are obtained for the second example $F_2(x)$ below.

For experiments with two independent variables, we will use some very well known functions, as the Franke's one in Example 3, and just a paraboloid in Example 4. We can see their graphics in the Figures 3.2.6 and 3.2.7.

Example 2: $F_2 : [0, 1] \longrightarrow \mathbb{R}$

$$F_2(x) = (2 + e^{-50(2x-0.25))^{-1}} \ .$$

Example 3: $F_3 : [0, 1] \times [0, 1] \longrightarrow \mathbb{R}$

$$\begin{aligned} F_3(x, y) := \ & \tfrac{3}{4} e^{-((9x-2)^2+(9y-2)^2)/4} + \tfrac{3}{4} e^{-((9x+1)^2/49-(9y+1)/10)} \\ & + \tfrac{1}{2} e^{-((9x-7)^2+(9y-3)^2)/4} - \tfrac{1}{5} e^{-((9x-4)^2+(9y-7)^2)} \end{aligned} \ .$$

Example 4: $F_4 : [-1, 1] \times [-1, 1] \longrightarrow \mathbb{R}$

$$F_4(x, y) := \ x^2 + y^2 \ .$$

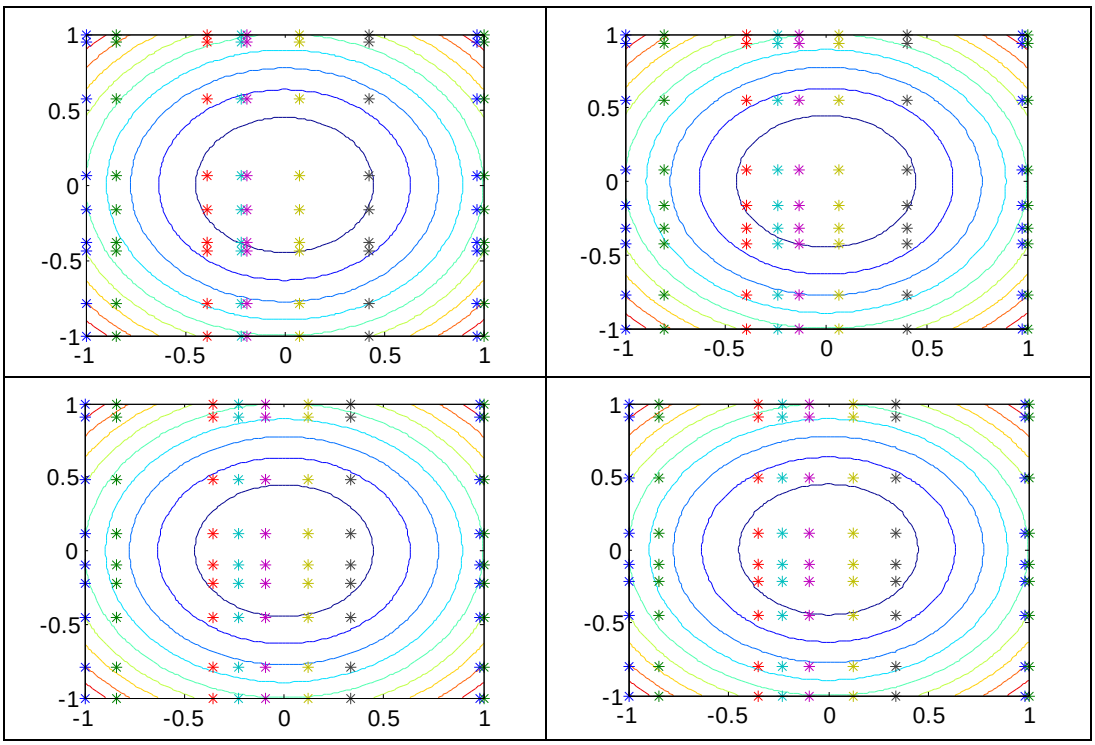

Figure 3.3.10: From left to right and top to down we show the evolution of the distribution of interpolating knots for $F_4(x, y)$.

In order to analyze the behavior of the bicubic spline interpolation procedure, a TDS with a large number of knots is not necessary. In Figures 3.2.5 and 3.3.10 we can see the evolution of interpolating knots in both cases, whereas Figures 3.2.8 and 3.3.9 show the Pareto fronts for the functions $F_3(x, y)$ and $F_4(x, y)$, respectively.

3.4 Conclusions

In this chapter, a novel methodology is presented for knots placement for cubic and bicubic splines interpolation of functions of one or two variables, respectively, showing the effectiveness of the strategy for different types of functions.

So, the goal of using a MOGA for placement of the knots in such case of interpolating functions can be summarized as follows:

(1) It has been sufficiently proven that the placement of the knots in spline

interpolation has an important and considerable effect on the behavior of the final results, but the optimal placement or location of knots is not known a priori.

(2) The number of knots to be used in classical MOGA approaches, should be selected a priori by the designer; but using our procedure, a Pareto front for different or variable number of knots used can also be directly optimized when less knots are necessary to obtain the same level of approximation.

As can be seen, in all the examples the mean square error (MSE) with the MOGA tends to be reduced when the number of knots to construct the interpolating B-spline increase, but after the appropriate evolution of the interpolating knots, not too many of these points are needed to obtain acceptable, or even good, results. In our particular case, we are seeking to minimize at the same time only two normalized discrete versions of some approximate errors, obtained from the usual norms in $\mathcal{C}(R)$ (4.6) and $\mathcal{L}^2(R)$ (4.7), that are not opposite convex functions of the involved variables of the MOGA; so the framework of the NSGA-II setting is the more convenient. In fact our main goal minimizing at the same time these discretized errors with our interpolation procedure is not only solve this interpolation issue, but also obtain a good fitting between the original and the interpolated curve or surface ir order to capture the maximum information of it with the representation of the obtained interpolating curve or surface.

Chapter 4

Optimal centers allocation in smoothing radial basis functions

4.1 Introduction

In the last decades, Radial Basis Functions (RBFs) methods has been developed in a way that contributed to their great spread and emergence, mainly for their simplicity and having been lauded for the ability in solving multivariate scattered data approximation problems [8]. The main advantages of these methods are their no need for a mesh or triangulation, their simple implementation, dimension independence, and no staircasing or polygonization for boundaries [12, 9].

Many authors have presented several methodologies for approximating or fitting a surface to data: such as the methodology that is based on a polynomial interpolation theorem of Lagrange [34, 80], and the other important approach that is based on Hermite polynomials [69].

In [70] the methodologies for Single Objective Problems (SOP) have been presented for selection and optimization of centers, mainly based on techniques for the selection of center points through non-dominated sorting, where promising points are selected as centers. On the other hand, the authors in [43] developed techniques to do centers selection and estimation

simultaneously by a componentwise boosting algorithm. And the centers of RBFs are selected among the vertices of the Voronoi diagram of the sample data point in [44]. The methodology in [1] has been optimizing the shape parameter of radial basis functions for the solution of ordinary differential equations. The authors in [2] use an approach based on genetic algorithms to optimize the centers and Radial Basis Function Neural Networks (RBFNNs). But they used one type of neural network where the basis function have the following expression:

$$\phi(\mathbf{x}, \mathbf{c}, r) = exp(\frac{\|\mathbf{x} - \mathbf{c}\|}{r})$$

where $\mathbf{c}$ is the central point of function ϕ, r is the radius and $\mathbf{x}$ is the input vector.

In this chapter, a new methodology for optimal placement of random centers, for approximating or fitting a curve or surface to data, using radial basis functions of one or several variables, is developed. A new technique is presented to optimize, both the number of radial basis centers and its optimal placement, using multi-objective genetic algorithms.

4.2 Proposed methodology

Let Ω be an open bounded connected nonempty subset of $\mathbb{R}^d$ (usually $d = 1, 2, 3$), with Lipschitz-continuous boundary; we will also use the classical notation $H^k(\Omega)$ to denote the usual Sobolev space of all distributions u, whose partial derivatives, up to order k, are in the classical Lebesgue space $\mathcal{L}^2(\Omega)$.

4.2.1 Spaces of radial functions of class $\mathcal{C}^2$.

Let $m > 1$ be a positive integer, and let $\Pi_{m-1}(\mathbb{R}^d)$ denote the space of polynomials on $\mathbb{R}^d$ of degree at most $m - 1$, whose dimension is denoted by $d(m)$ and let $\{q_1, \cdots, q_{d(m)}\}$ be the standard basis of $\Pi_{m-1}(\mathbb{R}^d)$. Let us give an arbitrary finite set $\{\mathbf{b}_1, \cdots, \mathbf{b}_M\} \subset \mathbb{R}^d$ of distinct approximation points and a set of real values $U = \{f_1, \ldots, f_M\}$. We will also use a set of knots $\{\mathbf{a}_1, \cdots, \mathbf{a}_N\}$; and for each of them, a radial function $\phi(\cdot - \mathbf{a}_i)$.

The main goal of this section is to approximate, in the best possible way, the points $\{(\mathbf{b}_1, f_1), \ldots, (\mathbf{b}_M, f_M)\} \subset \mathbb{R}^{d+1}$.

To conclude this section, we define the radial basis functions that will be

used in the sequel: we consider the following function, ($\varepsilon \in \mathbb{R}^+,\ t \geq 0$)

$$\phi_\varepsilon(t) = -\frac{1}{2\varepsilon^3}\left(e^{-\varepsilon\sqrt{t}} + \varepsilon\sqrt{t}\right) \tag{4.1}$$

and the corresponding radial functions ($\varepsilon \in \mathbb{R}^+,\ \mathbf{x} \in \mathbb{R}^d$)

$$\phi_\varepsilon(\mathbf{x}) = \phi_\varepsilon(< \mathbf{x} >_d^2) = -\frac{1}{2\varepsilon^3}\left(e^{-\varepsilon < \mathbf{x} >_d} + \varepsilon < \mathbf{x} >_d\right)$$

where $< \cdot >_d$ denote the Euclidean norm on $\mathbb{R}^d$.

So, the idea is to use a finite dimensional space H^N, whose basis is of this type

$$H^N \equiv span\left\{q_1, \cdots, q_{d(m)}, \phi(\cdot - \mathbf{a}_1), \cdots, \phi(\cdot - \mathbf{a}_N)\right\}.$$

Remark. When $M = N$ and $\mathbf{b}_i \equiv \mathbf{a}_i$ for $i = 1, \ldots N$, we could also consider an interpolating problem at the knots, where the interpolating function S_N has explicitly the following form

$$S_N(\mathbf{x}) = \sum_{i=1}^{N} \lambda_i \phi_\varepsilon(\mathbf{x} - \mathbf{a}_i) + \sum_{j=1}^{d(m)} \alpha_j\, q_j(\mathbf{x}), \quad \mathbf{x} \in \mathbb{R}^d, \tag{4.2}$$

where the coefficients $\boldsymbol{\lambda} = (\lambda_1, \cdots, \lambda_N)^\top$, and $\boldsymbol{\alpha} = (\alpha_1, \cdots, \alpha_{d(m)})^\top$ are the solution of the linear system

$$\begin{pmatrix} \mathbf{A}_\varepsilon & \mathbf{M} \\ \mathbf{M}^\top & \mathbf{O} \end{pmatrix} \begin{pmatrix} \boldsymbol{\lambda} \\ \boldsymbol{\alpha} \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ \mathbf{0} \end{pmatrix} \tag{4.3}$$

with $\mathbf{A}_\varepsilon = (\phi_\varepsilon(\mathbf{a}_i - \mathbf{a}_j))_{1 \leq i,j \leq N}$ be a $N \times N$ matrix, $\mathbf{M} = (q_j(\mathbf{a}_i))_{\substack{1 \leq i \leq N \\ 1 \leq j \leq d(m)}}$ be a $N \times d(m)$ matrix, $\mathbf{M}^\top$ denotes the transpose of $\mathbf{M}$, $\mathbf{O}$ be the $d(m) \times d(m)$ zero matrix, $\boldsymbol{\lambda} = (\lambda_1, \cdots, \lambda_N)^\top$ and $\mathbf{f} = (f_1, \cdots, f_N)^\top$ two vectors of $\mathbb{R}^N$ and $\boldsymbol{\alpha} = (\alpha_1, \cdots, \alpha_{d(m)})^\top$ and $\mathbf{0}$ is the zero vector of $\mathbb{R}^{d(m)}$.
Take into account that we are considering the interpolating conditions:

$$S_N(\mathbf{a}_i) = f_i, \quad i = 1, \cdots, N,$$

together with the constraints

$$\sum_{i=1}^{N} \lambda_i q_j(\mathbf{a}_i) = 0,\ j = 1, \cdots, d(m).$$

4.2.2 Smoothing radial basis functions

In this case, the problem to be resolved can be formulated as follows: given a set of real values and an approximation data set $\{(\mathbf{b}_i, f_i) : \ i = 1, \ldots, M\}$, now we still want to obtain $\sigma \in H^N$ such that

$$\sigma(\mathbf{b}_i) \approx f_i, \ \ i = 1, \cdots, M,$$

but now by means of the following minimization problem:

Find $\sigma \in H^N$ such that

$$\mathcal{J}(\sigma) \leq \mathcal{J}(\upsilon), \quad \forall \upsilon \in H^N \tag{4.4}$$

where $\tau \in (0, \infty)$ and

$$\mathcal{J}(\upsilon) = \sum_{i=1}^{M} (u_i - \upsilon(\mathbf{b}_i))^2 + \tau |\upsilon|_2^2,$$

where

$$|\upsilon|_2^2 = \sum_{|\boldsymbol{\alpha}|=2} \int_\Omega (\partial^\alpha \upsilon(\mathbf{x}))^2 d\mathbf{x}$$

with

$$|\boldsymbol{\alpha}| = \alpha_1 +, \ldots + \alpha_d, \forall \boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_d) \in \mathbb{N}^d, \ \ \partial^{\boldsymbol{\alpha}} \upsilon(x_1, \ldots, x_d) = \frac{\partial^{|\boldsymbol{\alpha}|} \upsilon}{\partial x_1^{\alpha_1} \cdots \partial x_d^{\alpha_d}},$$

Thus

$$\sigma(\mathbf{x}) = \sum_{l=1}^{N+d(m)} \alpha_l \, w_l,$$

where

$$w_l = \left\{ \begin{array}{ll} \phi_\varepsilon(\cdot - \mathbf{a}_l), & l = 1, \cdots, N \\ q_{l-N}, & l = N+1, \cdots, N+d(m), \end{array} \right\};$$

and $\alpha_1, \cdots, \alpha_{N+d(m)} \in \mathbb{R}$ are the control coefficients (i.e., the problem unknows) obtained as the solution of the variational problem formulated in the following Theorem (see for example [36, 38]):

Theorem 2. The minimization problem (4.4) has a unique solution that also is the only solution of the following variational problem:
Find $\sigma \in H^N$ such that, for all $\upsilon \in H^N$:

$$\sum_{i=1}^{M} \sigma(\mathbf{b}_i) \, \upsilon(\mathbf{b}_i) + \tau(\sigma, \upsilon)_2 = \sum_{i=1}^{M} \upsilon(\mathbf{b}_i) \, f_i \tag{4.5}$$

where

$$(\sigma, v)_2 = \sum_{|\boldsymbol{\alpha}|=2} \int_\Omega (\partial^{\boldsymbol{\alpha}} \sigma(\mathbf{x}) \;\; \partial^{\boldsymbol{\alpha}} v(\mathbf{x}) \,) d\mathbf{x}$$

Proof.
The expression

$$[[v]] := \left(\sum_{i=1}^{M} v(\mathbf{b}_i)^2 + (v, v)_2 \right)^{\frac{1}{2}}$$

constitutes a norm on H^N equivalent to its usual norm. Then, as a consequence, the continuous symmetric bilinear form

$$a : H^N \times H^N \longrightarrow \mathbb{R}$$

defined by

$$a(u, v) = \sum_{i=1}^{M} u(\mathbf{b}_i) v(\mathbf{b}_i) + \tau (u, v)_2$$

is H^N-elliptic. Besides, $\psi : H^N \longrightarrow \mathbb{R}$

$$\psi(v) := \sum_{i=1}^{M} f_i \, v(\mathbf{b}_i)$$

is clearly a continous linear form on this space.

Then, applying the Lax-Milgram Lemma (see [3] for example), there exists a unique function $\sigma \in H^N$ such that

$$a(\sigma, v) = \psi(v), \;\; \forall v \in H^N$$

that is, (4.5) holds. Moreover σ is the unique function of H^N that minimizes the functional

$$\widetilde{\mathcal{J}}(v) = 2 \left(\frac{1}{2} a(v, v) - \psi(v) \right) + \sum_{i=1}^{M} f_i^2,$$

concluding that σ also minimizes the functional $\mathcal{J}$. □
By linearity we can reduce the problem (4.5) to the following linear system

$$\left(\mathbf{A}\mathbf{A}^\top + \tau \mathbf{R} \right) \boldsymbol{\alpha} = \mathbf{A}\mathbf{f}$$

where $\mathbf{A} = (w_i(\mathbf{b}_j)), \;\; i = 1, \ldots, N + d(m), \;\; j = 1, \cdots, M.$

$$\mathbf{R} = \left(\sum_{|\boldsymbol{\alpha}|=2} \int_{\Omega} \partial^{\boldsymbol{\alpha}} w_i \partial^{\boldsymbol{\alpha}} w_j \, d\mathbf{x} \right)_{i,j=1,\cdots,N+d(m)}$$

$$\boldsymbol{\alpha} = (\alpha_i)_{i=1,\cdots,N+d(m)} \in \mathbb{R}^{N+d(m)}, \quad \mathbf{f} = (f_i)_{i=1,\cdots,M}$$

4.3 Simulation examples

The objective of this study is to analyze the performance of this procedure for different types of functions, optimizing the centers placement using radial basis functions by Multi Objective Genetic Algorithms (MOGA). Different experiments have been carried out, and we will show in this section the results for the approximation of several functions, also presenting the evolution of the optimal distribution of points, together with the related Pareto fronts.

At this step, it is also important to consider two approximations of error estimations that are appropriate normalizations of the discrete version of the usual norms in $\mathcal{C}(\Omega)$ and $\mathcal{L}^2(\Omega)$ respectively, and are given by the expressions:

$$E_c = \frac{\max\limits_{i=1,\ldots,M} |f(\mathbf{b}_i) - \sigma(\mathbf{b}_i)|}{\max\limits_{i=1,\ldots,M} |f(\mathbf{b}_i)|} \tag{4.6}$$

$$E_l = \sqrt{\frac{\sum\limits_{i=1}^{M} (f(\mathbf{b}_i) - \sigma(\mathbf{b}_i))^2}{\sum\limits_{i=1}^{M} (f(\mathbf{b}_i))^2}} \tag{4.7}$$

In order to analyze the behavior of the radial basis approximations, a TDS with a large number of knots will be used (400 knots), with a population number of 40 individuals, until 20 generations in each one of following examples.

4.3.1 One dimensional examples

Example 1: $f_1 : [0, \pi] \longrightarrow \mathbb{R}$

$$f_1(x) = 0.12 + .25 \exp^{-4(x-\frac{\pi}{4})^2} \cos(2x) \sin(2\pi x) \ .$$

To test the behavior of the approximation for the presented methodology, performed by optimization of the centers placement of radial basis functions by a NSGA-II algorithm Figure 4.3.1 shows the results of the approximation of function $f_1(x)$, using radial basis function approximation, while in Figure 4.3.2 we can see how the E_l error decreases clearly when the number of centers increases. We can also see in the left colum of Figure 4.3.1 the radial basis approximation corresponding to the last centers' distribution, whose evolution is also showed in the right column, with increasing number of interior centers for the function $f_1(x)$ with 7, 9 and 11 centers, respectively

Example 2: $f_2 : [0, 1] \longrightarrow \mathbb{R}$

$$f_2(x) = 0.5e^{-5(x-\pi/2)^2} sin(4\pi x)\, cos(4x) \ .$$

For this Example 2, the numerical approximation results for $f_2(x)$ and the evalution of centers are shown in Figure 4.3.3, on the other hand the two approximation discretization errors for $f_2(x)$ are also described in Table 4.3.1

We can see in the left column of Figure 4.3.3 the radial basis approximation corresponding to the last centers' distribution, whose evolution is also showed at the right column, with increasing number of interior centers for the function $f_2(x)$, using 2, 4 and 6 interior centers respectively.

Table 4.3.1: Two approximation discretization errors for $f_2(x)$.

centers	E_l	E_c
2	4.3978×10^{-2}	1.8918×10^{-3}
4	2.6664×10^{-2}	6.7993×10^{-4}
6	4.0691×10^{-3}	1.9208×10^{-4}
8	3.9539×10^{-3}	1.7241×10^{-4}

Example 3: $f_3 : [-1, 1] \longrightarrow \mathbb{R}$

$$f_3(x) = 1 - e^{-50|x|} \ .$$

Where the graphical result approximating $f_3(x)$ with 20 interior centers using 20 generations in the NSGA-II algorithm, using radial basis functions, is showed in the Figure 4.3.4

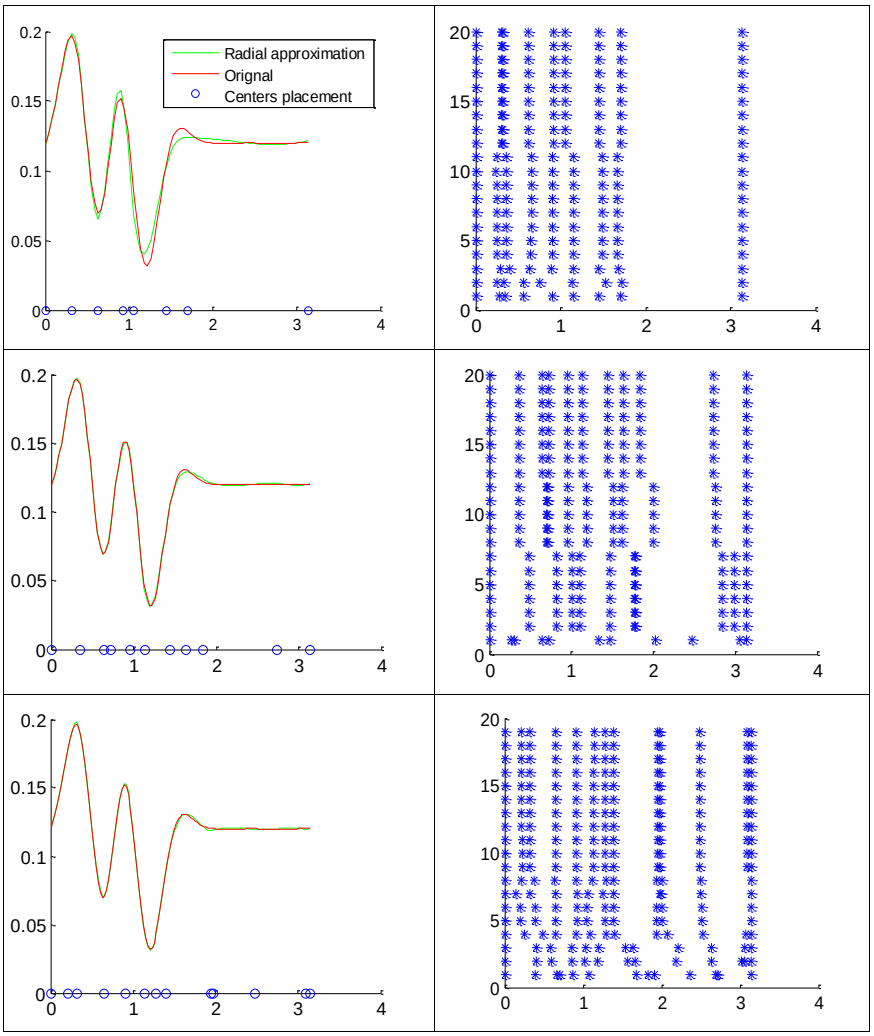

Figure 4.3.1: Radial basis approximation and evolution of final centers' placement for the function $f_1(x)$ with 7, 9 and 11 centers, respectively.

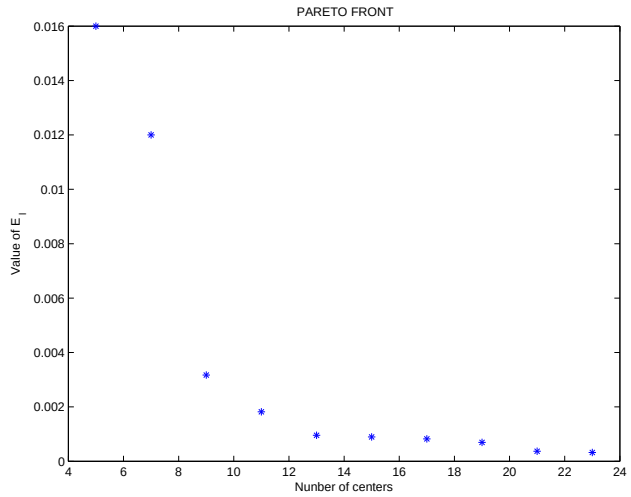

Figure 4.3.2: Pareto front of E_l error vs. number of centers for $f_1(x)$.

4.4 Conclusions

This chapter presents an important and interesting methodology for optimizing both the number and the corresponding center placement in the problem of radial basis function approximation of functions of one or several independent variables, where we can see clearly the effectiveness of the strategy for different types of functions. Also the preliminary numerical results of this MOGA strategy for centers placement in approximating functions of one variable show that the placement of the centres in radial basis approximation function has an important and significant impact on the behavior of the final results, because the ideal place or location of the centers is not known in advance.

On the other hand, in classic methods the designer should choose the number of centers that will be used. But using a MOGA procedure of the type we are using, the corresponding Pareto's fronts can also be directly improved using a different or variable number of centers at the same time the simulations are being performed, indicating that increasing the number of center of radial basis in the definition of radial basis function, usually also increases the accuracy of the approximation, but only up to a certain level, where the computational effort it is not worthy enough and we can admit the coalescence of very close knots maintaining the same level of accuracy.

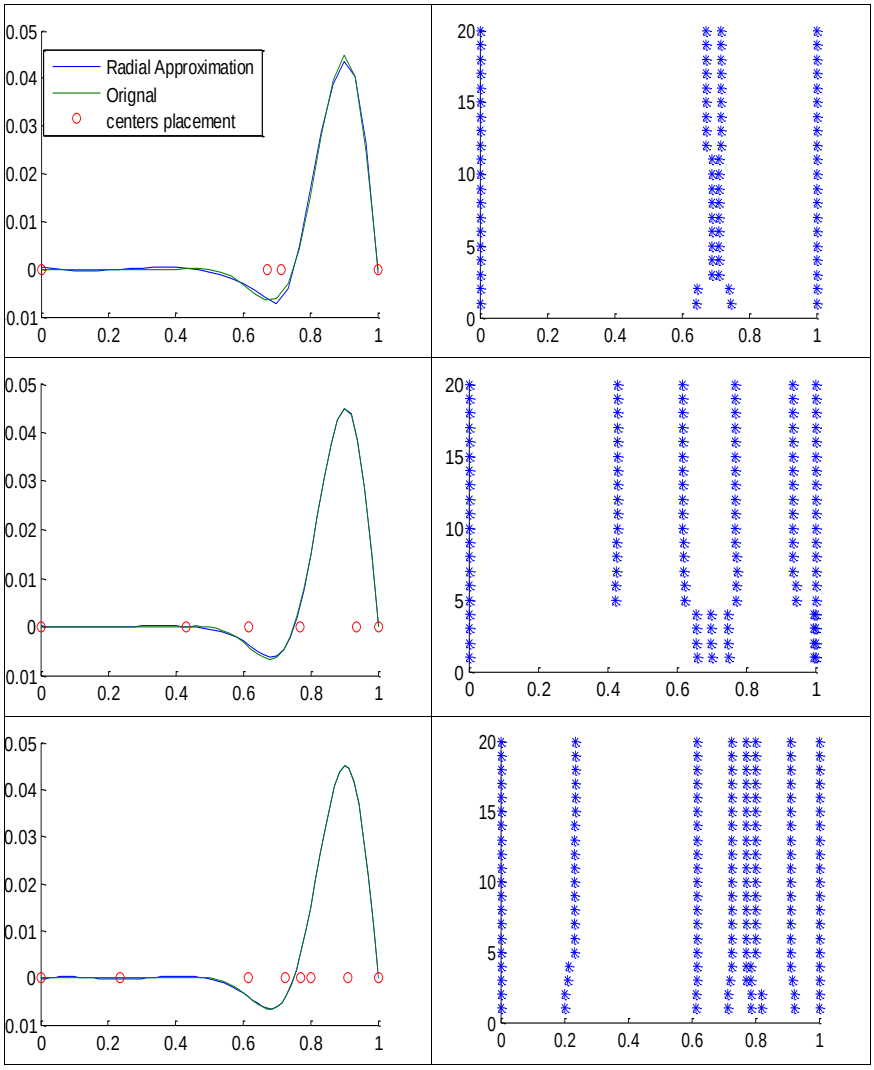

Figure 4.3.3: Radial basis approximation and evolution of final centers' placement for the function $f_2(x)$ with 2, 4 and 6 interior centers, respectively.

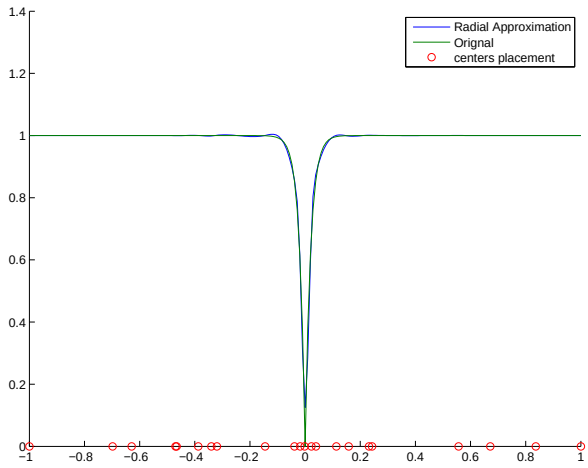

Figure 4.3.4: Graphical results for $f_3(x)$ with 20 interior centers (marked on the x-axis)

Bibliography

[1] F. Afiatdoust, M. Esmaeilbeigi, Optimal variable shape parameters using genetic algorithm for radial basis function approximation *Ain Shams Engineering Journal*, 6, 639–647 (2015). 80

[2] Awad M., Pomares H., Herrera L.J., González J., Guillén A., Rojas F. (2005) Approximating I/O Data Using Radial Basis Functions: A New Clustering-Based Approach. In: Cabestany J., Prieto A., Sandoval F. (eds) Computational Intelligence and Bioinspired Systems. IWANN 2005. Lecture Notes in Computer Science, vol 3512. Springer, Berlin, Heidelberg 80

[3] K. Atkinson, W. Han : *Theoretical Numerical Analysis: A Functional Analysis Framework*. Springer–Verlag, Nueva York (2001). 16, 83

[4] Carlos Argáez, Sigurdur Hafstein and Peter Giesl, Wendland Functions A C++ code to compute them, SIMULTECH 2017, Madrid (Spain). Published in: Proceedings of the 7th International Conference on Simulation and Modeling Methodologies, Technologies and Applications, (2017). 323-330. SCITEPRESS, ISBN 9789897582653. 4, 27, 28

[5] Carl de Boor, A Practical Guide to Splines, Revised Edition, Applied Mathematical Sciences, vol. 27. Springer (1978). 21

[6] Haim Brezis, Functional Analysis, Sobolev Spaces and Partial Differential Equations. Springer (2010). 16

[7] J. Brownlee, Clever Algorithms: Nature-Inspired Programming Recipes. `http://www.cleveralgorithms.com/` 68

[8] Buhmann M. D. Radial basis functions: theory and implementations. Cambridge monographs on applied and computational mathematics, *Cambridge: Cambridge University Press,* vol. 12, (2003). 4, 27, 28, 79

[9] Buhmann M, Dyn N. Spectral convergence of multiquadric interpolation. *Proc Edinb Math Soc* (2), 36(2), 319–333, (1993). 79

[10] Edwin Catmull, Raphael Rom, A Class of Local Interpolating Splines, Computer Aided Geometric Design 317—326 (1974).

[11] R.E. Carlson and T.A. Foley, The parameter R^2 in multiquadric interpolation, Comput. Math. Appl. 21 , 29–42 (1991).

[12] Cheng AH-D, Golberg M.A., Kansa E.J., Zammito G, Exponential convergence and $h-c$ multiquadric collocation method for partial differential equations. Numer Methods Partial Diff. Eq., 19(5), 571–94 (2003). 79

[13] W. Cheney, Analysis for Applied Mathematics. Springer (2001). 16

[14] C. K. Chui : *Multivariate splines.* CBMS-NSF Series Appl. Math. Vol. 54. SIAM Publications, Philadelphia (1988). 21

[15] List of References on Evolutionary Multiobjective Optimization http://delta.cs.cinvestav.mx/ ccoello/EMOO/emoopage.html 10, 59

[16] Darve E., The fast multipole method: Numerical implementation. Journal of Computational Physics, 160 (1), 195–240 (2000). 4, 27, 28

[17] K. Deb, S. Agrawal, A. Pratap, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, *IEEE Trans. Evol. Comput.* 6(2) 182–197 (2002). 9, 36, 46, 47, 59, 68

[18] M. Emmerich, A. Deutz, Multicriteria Optimization and Decision Making, LIACS Master Course: Autumn/Winter 2006.

[19] Esmaeilbeigi M, Hosseini M.M., Mohyud-Din Syed Tauseef. A new approach of the radial basis functions method for telegraph equations. *Int J Phys Sci,* 2011 ,6(6) 151-27.

[20] Esmaeilbeigi M., Hosseini M.M., Dynamic node adaptive strategy for nearly singular problems on large domains. *Eng Anal. Bound. Elem.* , 2012, (36) 1311-–1321.

[21] T.A. Foley, Near optimal parameter selection for multiquadric interpolation, J. Appl. Sci. Comput. 1 (1994), 54–69.

[22] C. M. Fonseca, P. J. Fleming, An overview of evolutionary algorithms in multiobjective optimization, Evol. Comput. 3 (1) (1995) 1–16. 10, 36, 59

[23] W. Freedn, Spherical spline interpolation-basic theory and computational aspects, Journal of Computational and Applied Mathematics, 11 (1984), 367–375. North-Holland.

[24] A. Gálvez, A. Iglesias, A. Avila, C. Otero, R. Arias, C. Manchado, Elitist clonal selection algorithm for optimal choice of free knots in B-spline data fitting *Applied Soft Computing* **26** (2015), 90–106. 6, 8, 42, 57

[25] C. H. García-Capulín, F. J. Cuevas, G. Trejo-Caballero, H. Rostro-González, A hierarchical genetic algorithm approach for curve fitting with B-splines, Genet. Program. Evolvable Machines, **16** (2015), 151–166. 6, 9, 42, 58

[26] A. Gelas, Compactly Supported Radial Basis Functions: Multidimensional Reconstruction and Applications, Institut National des Sciences Appliquées de Lyon, 2006-ISAL-0095. 4, 27, 28

[27] Hämmerlin G., Hoffman KH. (1991) Splines. In: Numerical Mathematics. Undergraduate Texts in Mathematics (Readings in Mathematics). Springer, New York, NY 21, 25

[28] J. H. Holland, Adaptation in Natural and Artificial Systems, 1975. 68

[29] J. Hoschek, D. Lasser, Fundamentals of computer aided geometric design, *A. K. Peters, Ltd., Wellesley*, 1993.

[30] M.C. Huang, C.C. Tai, The pre-processing of data points for curve fitting in reverse engineering, *Int. J. Adv. Manuf. Technol*, 16 (2000) 635–642.

[31] Ruben Interian, Juan M. Otero, Celso C. Ribeiro, Anselmo A. Montenegro, Curve and surface fitting by implicit polynomials: Optimum degree finding and heuristic refinement , *Computers and Graphics* 67 (2017) 14-–23. 9, 58

[32] Hisao Ishibuchi and Tadahiko Murata, Multi-objective genetic local search algorithm, in Proceeding of 1996 IEEE International Conference on Evolutionary Computation (ICEC96), 119–124. Piscataway, N.J. IEEE Press (1996).

[33] H. Ishibuchi, R. Imada, Y. Setoguchi, Y. Nojima, Performance comparison of NSGA-II and NSGA-III on various many-objective test problems, in: 2016 IEEE Congress on Evolutionary Computation (CEC-2016), 3045–3052. 10, 59

[34] H. Jeffreys, B.S. Jeffreys, Lagrange's Interpolation Formula, in Methods of Mathematical Physics, 3rd. Ed., *Cambridge University Press, Cabridge (England)* , 1988. vol. 59, p. 260. 79

[35] H. Kang, F. Chen, Y. Li, J. Deng, Z. Yang, Knot calculation for spline fitting via sparse optimization, *Computer-Aided Design* **58** (2015) 179–188. 7, 9, 42, 58

[36] A. Kouiba, M. Pasadas, Smoothing variational splines, Applied Mathematics Letters, 13 (2000), 71—75 . 4, 31, 82

[37] A. Kouibia, M. Pasadas, Approximation of surfaces by fairness bicubic splines, *Advances in Computational Mathematics.* Kluwer Ac. Publ. **20** (1-3)(2004), 87–103. 45

[38] A. Kouibia and M. Pasadas, Approximation of curves by fairness cubic splines, *Applied Mathematical Science* **1**(5) (2007), 227–240. 82

[39] A. Kouiba, M. Pasadas, Approximation by interpolating vatariational splines, Computational and Applied mathematics, 218, (2008), 342–349. 4, 32

[40] R. Kozera, L. Noakes, Optimal knots selection for sparse reduced data, in: F. Huang, A. Sugimoto (Eds.), Image and Video Technology – PSIVT 2015 Workshops, Springer International Publishing, Cham, (2016), 3–14. 6, 8, 42, 58

[41] O. Kramer, H. Danielsiek, DBSCAN-based multi-objective niching to approximate equivalent Pareto-subsets, in: Proceedings of the 12th Annual Conference on Genetic and Evolutionary Computation, GECCO '10, ACM, New York, NY, USA, 2010, pp. 503–510. 10, 59

[42] Y-K. Lai, S-M. Hu, H. Pottmann, Surface fitting based on a feature sensitive parametrization *Computer-Aided Design* **38**(7) (2006), 800–807. 6, 8, 41, 57, 58

[43] Leitenstorfer, Florian; Tutz, Gerhard (2006) : Knot selection by boosting techniques, Discussion paper // Sonderforschungsbereich 386 der

Ludwig-MaximiliansUniversität München, No. 481, Techn. Univ.; Sonderforschungsbereich 386, Statistische Analyse Diskreter Strukturen, München, http://nbn-resolving.de/urn:nbn:de:bvb:19-epub-1849-4 79

[44] M. Samozino, M. Alexa, P. Alliez, and M. Yvinec, Reconstruction with Voronoi Centered Radial Basis Functions, Eurographics Symposium on Geometry Processing (2006). 80

[45] T. Lyche, K. Mørken, The sensitivity of a spline function to perturbations of the knots, BIT Numerical Mathematics 39 (2) (1999) 305–322. 8, 57, 58

[46] T. Lyche, K. Mørken, A data-reduction strategy for splines with applications to the approximation of functions and data. *IMA Journal of Numerical Analysis* **8** (1998), 185–208. 48

[47] Mathworks, MATLAB - The Language of Technical Computing. Available from: <www.mathworks.com/products/matlab>, 2010.

[48] Z. Majdisova, V. Skala, A Radial Basis Function Approximation for Large Datasets, SIGRAD 2016. 4, 27, 28

[49] K. F. Man, K. S. Tang, S. Kwong, Genetic Algorithms: Concepts and Designs with Disk, 2nd Edition, Springer-Verlag, Berlin, Heidelberg, 1999. 9, 58

[50] Mark J.L. Orr , Introduction to Radial Basis Function Networks, Report of the Centre for Cognitive Science, University of Edinburgh (1996). 4, 27, 28

[51] H. Park, An error-bounded approximate method for representing planar curves in b-splines, Computer Aided Geometric Design 21 (5) (2004) 479 – 497. 8, 57

[52] H. Park, J.-H. Lee, B-spline curve fitting based on adaptive curve refinement using dominant points, Comput. Aided Des. 39 (6) (2007) 439–451. 8, 57

[53] M. Pasadas, M.L. Rodríguez, An approximation method with data selection process, Mathematics and Computers in Simulation 79 (12) (2009) 3567 – 3576, the International Conference on Approximation Methods and numerical Modeling in Environment and Natural Resources. 8, 57

[54] H. Pottmann, S. Leopoldseder, M. Hofer, T. Steiner, W. Wang, Industrial geometry: recent advances and applications in CAD, Comput. Aided Des. **37** (2005), 751–766. 6, 41

[55] M. J. D. Powell. The Theory of Radial Basis Functions Approximation, in Advances of Numerical Analysis, *Oxford: Clarendon Press,* pp. 105–210, 1992.

[56] S. Prahan, D. Patra, Non-grid medical image registration using adaptive knot selection in P-spline, *IEEE Students Technology Symosium* (2016), 250–254. 6, 42

[57] P. M. Prenter, *Splines and Variational Methods.* A Wiley-Interscience Publication, New York (1989). 21

[58] R. V. Rao, Teaching Learning Based Optimization Algorithm: And Its Engineering Applications, 1st Edition, Springer Publishing Company, Incorporated, 2015. 59

[59] R. V. Rao, A. Saroj, A self-adaptive multi-population based jaya algorithm for engineering optimization, Swarm and Evolutionary Computation 37 (2017) 1 – 26. doi:https://doi.org/10.1016/j.swevo.2017.04.008. 59

[60] S. Rippa, An algorithm for selecting a good value for the parameter c in radial basis function interpolation, Adv. Comput. Math. 11(2) 1999, 193–210.

[61] I. J. Schoenberg, Contribution to the problem of approximation of equidistant data by analytic functions, Parts A and B, Quart. Appl. Math. 4(1946), 45–99, 112–141. 21

[62] I.J. Schoenberg : *Cardinal Spline Interpolation.* SIAM (1973). 21

[63] L.L. Schumaker : *Spline Functions: Basic Theory.* Wiley-Interscience, New-York (1981). 21

[64] H. Seada, K. Deb, U-NSGA-III: A unified evolutionary optimization procedure for single, multiple, and many objectives: Proof-of-principle results, in: A. Gaspar-Cunha, C. Henggeler Antunes, C. C. Coello (Eds.), Evolutionary Multi-Criterion Optimization, Springer International Publishing, Cham, 2015, pp. 34–49. 10, 59

[65] M. Sommer, H. Strauss, A condition of Schoenberg–Whitney type for multivariate spline interpolation 5 (1) (1996) 381–397. 74

[66] N. Srinivas, K. Deb, Muiltiobjective optimization using nondominated sorting in genetic algorithms, Evol. Comput. 2 (3) (1994) 221–248. 10, 36, 59

[67] J. D. Schaffer, Multiple objective optimization with vector evaluated genetic algorithms, in: Proceedings of the 1st International Conference on Genetic Algorithms, L. Erlbaum Associates Inc., Hillsdale, NJ, USA, 1985, pp. 93–100. 10, 59

[68] J. Stoer, R. Bulirsch, Introduction to Numerical Analysis, third ed. (Texts in Applied Mathematics, vol. 12), Springer-Verlag, New York, 2002. 61

[69] G. Szego, M. Sakai, Orthogonal Polynomials, American Mathematical Society, Providence, RI, 1975. 79

[70] Tipaluck Krityakierne, Taimoor Akhtar, Christine A. Shoemaker, SOP: parallel surrogate global optimization with Paretocenter selection for computationally expensive single-objective problems, DOI 10.1007/s10898-016-0407-7, 2016. 79

[71] O. Valenzuela, B. Delgado-Márquez, M. Pasadas, Evolutionary computation for optimal knots allocation in smoothing splines, *Applied Mathematical Modelling*, **37** (2013) 5851–5863. 7, 34, 42, 45, 48

[72] O. Valenzuela, M. Pasadas, F. Ortuño, I. Rojas, Optimal Knots Allocation in Smoothing Splines using intelligent system *IWBBIO* 2013. 6, 41

[73] O. Valenzuela, M. Pasadas, Using simulated annealing for knot placement for cubic spline approximation, *Math. Models Eng. Sci.* (2010), 148–154. 7, 42

[74] O. Valenzuela, M. Pasadas, Using simulated annealing for knot placement for cubic spline approximation, in: Proceedings of the 2010 International Conference on Mathematical Models for Engineering Science, MMES'10, World Scientific and Engineering Academy and Society (WSEAS), Stevens Point, Wisconsin, USA, 2010, pp. 148–154. 9, 58

[75] O. Valenzuela, M. Pasadas, I. Rojas, A. Guillén, H. Pomares, Automatic Knot Asjustment For B-Spline Smoothing Approximation Using Improved Clustering Algorithm *IEEE International Conference on Fuzzy Systems*, 2013. 48

[76] T. Varady, R.R. Martin, Reverse engineering, in: G. Farin, J. Hoschek, M. Kim(Eds.), Handbook of Computer Aided Geometric Design, Elsevier Science, 2002. 6, 41

[77] Wei Wang, Fazhong Shi, Gang Zhao, Xiaoxiao Du, A new method for knot placement of B-Spline curve approximation and interpolation, *3rd International Conference on Materials Engineering, Manufacturing Technology and Control (ICMEMTC 2016)* 1527–1532. 6, 42

[78] S. Wang, S. Ali, A. Gotlieb, Random-weighted search-based multi-objective optimization revisited, in: C. Le Goues, S. Yoo (Eds.), Search-Based Software Engineering, Springer International Publishing, Cham, 2014, pp. 199–214. 10, 59

[79] R.-H. Wang, J.-X. Wang, The poisedness of interpolation problem for splines, Applied Numerical Mathematics 54 (1) (2005) 95 – 103. 8, 57, 58

[80] E.T. Whittaker, Robinson, Lagrange's formula of interpolation, Calculus of Observations: A Treatise on Numerical Mathematics, 4th ed., *Dover, New York* 1967. pp. 28–30. 79

[81] P. Yosida, Function Analysis. Springer-Verlag, Berlin (1974). 16

[82] Y. Zhang, J. Cao, Z. Chen, X. Li, X.-M. Zeng, B-spline surface fitting with knot position optimization, *Computer and Graphics* **58** (2016) 73–83. 6, 42, 58