IMPROVED INTERPOLATION IN SPH IN CASES OF LESS SMOOTH FLOW

by

ODDNY H. BRUN
B.S. Nursing, Bergen Deacon School of Nursing, Norway, 1976
B.S. Applied Mathematics/Statistics, University of Bergen, Norway, 1982
M.S. Applied Mathematics/Statistics, University of Bergen, Norway, 1983

A thesis submitted in partial fulfillment of the requirements
for the degree of Master of Science
in the Department of Modeling & Simulation
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Fall Term
2016

Major Professor: R. Paul Wiegand

ABSTRACT

We introduced a method presented in Information Field Theory (IFT) [Abramovich et al., 2007] to improve interpolation in Smoothed Particle Hydrodynamics (SPH) in cases of less smooth flow. The method makes use of wavelet theory combined with B-splines for interpolation. The idea is to identify any jumps a function may have and then reconstruct the smoother segments between the jumps. The results of our work demonstrated superior capability when compared to a particular challenging SPH application, to better conserve jumps and more accurately interpolate the smoother segments of the function. The results of our work also demonstrated increased computational efficiency with limited loss in accuracy as number of multiplications and execution time were reduced. Similar benefits were observed for functions with spikes analyzed by the same method. Lesser, but similar effects were also demonstrated for real life data sets of less smooth nature.

SPH is widely used in modeling and simulation of flow of matters. SPH presents advantages compared to grid based methods both in terms of computational efficiency and accuracy, in particular when dealing with less smooth flow. The results we achieved through our research is an improvement to the model in cases of less smooth flow, in particular flow with jumps and spikes. Up until now such improvements have been sought through modifications to the models' physical equations and/or kernel functions and have only partially been able to address the issue.

This research, as it introduced wavelet theory and IFT to a field of science that, to our knowledge, not currently are utilizing these methods, did lay the groundwork for future research

ideas to benefit SPH. Among those ideas are further development of criteria for wavelet selection, use of smoothing splines for SPH interpolation and incorporation of Bayesian field theory.

Improving the method's accuracy, stability and efficiency under more challenging conditions such as flow with jumps and spikes, will benefit applications in a wide area of science. Just in medicine alone, such improvements will further increase real time diagnostics, treatments and training opportunities because jumps and spikes are often the characteristics of significant physiological and anatomic conditions such as pulsatile blood flow, peristaltic intestine contractions and organs' edges appearance in imaging.

This work is dedicated to my husband Thorvald and our children Thorvald Edward, Karen Elisabeth, and Helene Christine. Thank you for inspiring me, this research would not have been any fun without you.

ACKNOWLEDGMENTS

I hereby extend my warmest gratitude to my committee members: Dr. Wiegand, Dr. Pensky and Dr. Kider. Dr. Wiegand, thank you for taking an interest in my research problem and guiding me through this journey. I truly appreciate your time, your recommendations and corrections to my work. I am humbled to be able to benefit from your professional guidance. Dr. Pensky, thank you for your encouragement and guidance from the very beginning, introducing me to wavelet theory and advising me along this "sparse" way. Dr. Kider, thank you for taking such an enthusiastic interest in my work at such a short notice. I have a lot to learn about SPH and I am so grateful to find the resources here at IST.

I would also like to thank Mr. Amit Goel for introducing me to the subject of SPH as we worked together at a graduate course project at UCF Institute for Simulation and Training.

Finally, I would like to extend my gratitude to US Geological Survey and PhysioNet for making their data public and available for my research.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1
INTRODUCTION

1.1 Objective

The objective of this research was to optimize interpolation and kernel selection criteria used in smoothed particle hydrodynamics (SPH). In particular, the work focused on SPH applied to less smooth types of flows such as flows with $jumps$ and $spikes$. Plunging water waves and flow of matters with partial restrictions are examples where jumps and spikes occur. Currently, modeling flow characterized by jumps and spikes results in less accuracy [Cabezón et al., 2008], [Khayyer et al., 2007], [Müller et al., 2004], because current interpolation methods incorrectly smooth over these discontinuities, which reduces the value of the SPH analysis. Further, some of the methods currently attempted for such cases are less computational efficient as they tend to introduce additional computational requirements. In our work we used a combination of empirical simulations and analytic tools to evaluate the effectiveness of the method. Our objective was to improve the interpolation and selection criteria for SPH kernel functions and demonstrate improved accuracy and stability of the analysis and hence contribute to the simplification of the kernel selection process. Depending on the outcome, we also aimed at providing constructive advice for increased computational efficiency for the SPH analysis, together with when and how these methods can lead to increase the capability of providing results in real time, a need particularly underlined in medical applications such as Müller et al. [2004] and Pyo et al. [2013].

With more accurate, stable and computational efficient methods for flow modeling, the contribution of our research will benefit applications in a wide area of science by more realistic and real-time results to increase the understanding of the the phenomena studied. In medicine, this will improve the understanding of physiological and pathological conditions and how to deal with such conditions. It will reduce cost and improve tools for training. Such effects will also be true for other areas of science where modeling flow of less smooth nature is used. Examples are animation, geophysics, hydrodynamics, just to mention a few. Last, but not least, our work will also increase interdisciplinary problem solving among different disciplines in science as we introduce wavelet theory and information field theory to an area of science that, to our knowledge, has not been done before.

1.2 Smoothed Particle Hydrodynamics (SPH)

SPH was first introduced in astrophysics by the work of Lucy [1977] and Gingold and Monaghan [1977]. Due to its modeling flexibility and computational efficiency, SPH soon gained interest from additional fields such as medicine, [Müller et al., 2004], interactive blood simulation for virtual surgery, [Chui and Heng, 2010], a meshless rheological model for blood-vessel interaction in endovascular simulation, [Hosseini and Feng, 2009], a particle-based model for the transportation of erythrocytes in capillaries, [Shahriari et al., 2012], pulsatile flow inside a rigid two-dimensional model of left heart cavity, [Basafa et al., 2013], presiction of cement infiltration of cancellous bone in osteoporotic bone augmentation, [Ho et al., 2014], 3D swallowing simula-

tion, [Pyo et al., 2013], physically based nonrigid registration applied to hepatic metastasis follow up, [Sinnott et al., 2012], investigating the relationship between peristaltic contraction and fluid transport in the human colon, in animation, [Xu et al., 2014], real-time generation of special effects in character animation, [Takamatsu and Kanai, 2011], fast and practical method for animating particle-based viscoelastic fluids, in geophysics, [Fakhimi and Lanari, 2014], simulation of rock blasting, [Boso et al., 2013], simulation of solute transport in highly heterogeneous formations, [Prakash et al., 2014], modeling the impact of dam failure scenarios on flood inudation, [Dalrymple and Rogers, 2006], numerical modeling of water waves, [Huang and Dai, 2014], large deformation and failure simulations for geo-disasters, and materials processing, [Cleary et al., 2013], prediction of industrial, biophysical and extreme geophysical flows.

SPH is a numerical method for approximating the evolution of systems described by differential equations. It is a mesh-free alternative to traditional methods such as finite elements methods (FEM) and finite difference method (FDM) where grids are used to mark the locations for calculating the physical values that describe the flow. In SPH the matter in question is represented as a sample of its particles. These particles, with their physical characteristics and governing physical equations, represent the movement of the material, i.e. the movement of the physical system [Liu and Liu, 2003].

The SPH method is a Lagrangian based particle method for flow analysis that ensures the three fundamental laws of physics are met:

- Conservation of mass

- Conservation of momentum
- Conservation of energy

Interpolation functions are used in a wide area of applications where the purpose is to draw conclusions about an entire field based on a finite number of observations. This is in particular the case of SPH, where a kernel function is used to interpolate the discrete particle values into a continuous function representing the flow of the matter. A finite number of particles of the matter to be analyzed and their characteristics such as density, mass and velocity are used to draw conclusions regarding the flow of all of the matter that these particles represent.

The physical characteristics of a given particle are calculated as a weighted average of the physical characteristics of the particles within a given neighborhood of the particle, and the kernel function represents the actual weights. The selection of weights and neighborhood extension, i.e. range of support, to a large degree determine the results. Unfortunately, such selection is, in general, non-trivial for most simulation applications.

1.3 The Need for Optimizing Interpolation and Kernel Selection Criteria

The purpose of interpolation is to recreate the continuous flow of the matter. The degree of accuracy achieved depends on kernel function characteristics such as smoothness and range of support. If a highly smooth kernel function is used, the result will be rather smooth. If the kernel function is less smooth, the results will be less smooth. Also, a larger range of support means a

higher number of particles are included in the calculations, which generally has a higher smoothing effect than a more narrow range of support. Over-smoothing will lead to poor recreation of peaks and jumps, whereas under-smoothing will introduce artificial jumps and peaks into the flow.

The criteria for choosing optimal kernel functions and their parameters in terms of accuracy and stability for the particular SPH analysis at hand are not well defined. This is in particular the case when using SPH to model mixed media, viscosity variations, flow limited by edges/borders or turbulent flow, or a combination of such factors. Jumps and spikes that typically occur in such flow have proven hard to model as they get over-smoothed by the kernel function, [Müller et al., 2004], [Cabezón et al., 2008], [Khayyer et al., 2007], [Zhang and Batra, 2004].

By practice, the typical characteristics of the interpolation functions most frequently used in SPH seem to differ from the wavelet functions proven optimal for IFT applications, which has a more advanced theoretical background with respect to issues of interpolation. So far, it has not been possible to find any examples from use of wavelets for interpolation in SPH applications.

Indeed, this type of challenge for SPH is also found in IFT where wavelets are often used for interpolation. In IFT dealing with, among others, images and signal theory, wavelet functions are used for interpolation and reconstruction of the image or signal of interest. Research in these fields has resulted in improved detection and reconstruction of spikes and jumps (Mallat [2009], Chapter 6.3, Multiscale edge detection). Images containing objects with sharp edges are examples of such non-smooth conditions. Similar non-smooth conditions are also typical for signals from other sources such as medicine (pulsatile blood flow, electrocardiograms, human body organ

scans), shock waves in homogeneous media, geophysical observations and explorations (recording of seismic signals) and ocean waves, among others.

One major advantage of wavelets is that they can represent functions using few coefficients. With an optimal selection of wavelet family, a non-smooth function may be fairly accurately represented by a few large coefficients compared to other type of transforms, including Fourier transforms that dominated signal theory until wavelet theory started to develop.

A Fourier transform defines a sparse representation (i.e. many coefficients with value equal to zero) of uniformly regular functions, whereas a wavelet transform defines a sparse representation of piece-wise regular signals [Mallat, 2009]. Fourier transform may be used to represent smooth functions. For less smooth functions, the number of Fourier coefficients will increase to the point of being computational inefficient. Wavelets can often be constructed as orthogonal basis, which is typically computationally more efficient compared to non-orthogonal basis.

The characteristics of flow analysis of mixed media, viscosity variations, flow limitations and turbulent flow are expected to represent less smooth flow than if these factors were not present. Hence, wavelets, or kernel functions with properties close to the properties of wavelets, may be better suited to represent this kind of flow as compared to more smooth kernel functions.

One area of non-smoothness characterized as jumps has in particular been identified as causing less accurate results in SPH [Müller et al., 2004], [Khayyer et al., 2007], [Cabezón et al., 2008]. A $jump$ is observed as a sudden, vertical increase or decrease in the flow and in mathematical terms described as a finite discontinuety in a function's range at a given domain location. A

spike is different from a jump in the sense that the derivative of the function is discontinuous, but the function itself is continuous, (Figure 1.1). In general, there is a loss of accuracy when interpolating functions containing jumps and spikes, as they tend to get over-smoothed and/or increased ripple occurs at these discontinuities, (Figure 1.2).

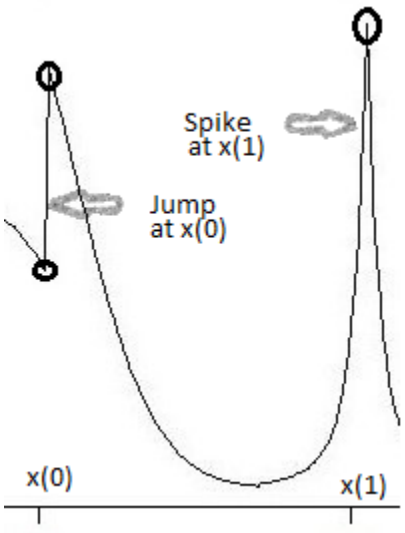

Figure 1.1: A function with a jump at x(0) and a spike at x(1)

A mathematical definition for jump is included in the methods chapter. An example of four functions with jumps are included, (Figure 1.3). In dealing with non-smooth flow in their analysis of breaking water waves, Khayyer et al. [2007], focused on modifications applied to the Navier-Stokes equations but still reported insufficient accuracy related to the jumps locations, (Figure 1.4). Müller et al. [2004] in their interactive blood simulations for virtual surgery used modifications of the physical equations combined with a spiky kernels to attempt stronger blood pulsation from an injured artery, but cited increased computational demand reduced the speed and made it hard to acheive the real time results needed for their simulation purpose. Cabezón et al. [2008] in their

analysis of shock fronts and thermal waves focused on the kernel function by introducing a set of functions where the spikiness of the kernel function could be adjusted to better preserve jumps, but still experienced inaccuracy (Figure 1.5). They further found their method of limited use due to increased numerical noise.

So far, the focus in SPH to deal with such non-smooth flow has been to modify the physical equations, the kernel function, or a combination of the two. The approach in this research was to build upon the results from Abramovich et al. [2007]. They used a two-step method to recreate functions with potential jumps by first identifying any jumps using Discrete Wavelet Transforms (DWT) and secondly by interpolation recreating the smoother segments of the function. Their analytic evaluation proves that the method can be as accurate as wanted as the number of sample points approaches infinity.

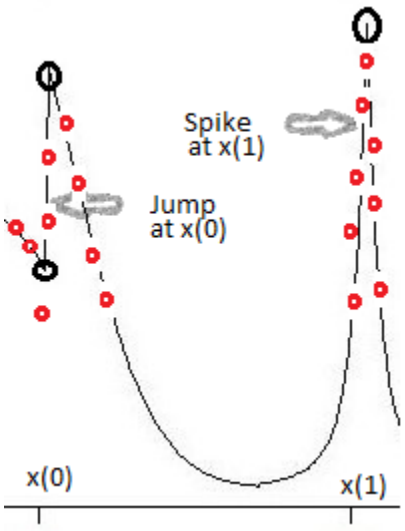

Figure 1.2: Illustration of over-smoothing (small red circles) from interpolating a function with jump and spike

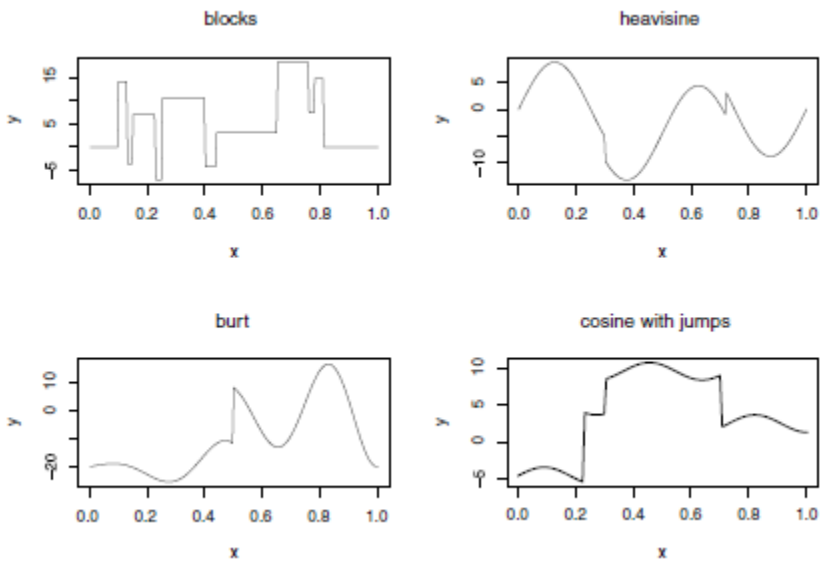

Figure 1.3: Four standard functions with jumps (from Abramovich et al. [2007])

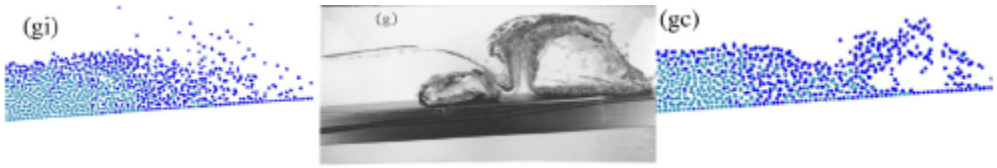

Figure 1.4: Picture of water flow with jumps (g) and SPH reconstruction results (gi) and (gc), (from Khayyer et al. [2007])

Our idea was that for SPH, if successful, this two-step method would reduce or eliminate the risk of over-smoothing jumps and spikes while recreating the smoother parts of the function between any jumps or spikes more accurate and efficient.

Our hope was that while focusing on functions with jumps, additional knowledge on how to deal with rather spiky functions would be gained by this research. While spikes, like jumps represent sudden changes in the flow of the matter, spikes differ from jumps by the fact that they

represent continuity whereas jumps are discontinuous in nature. Electrocardiogram (ECG) recordings of a normal human heartbeat (Figure 3.1) is an example of signal with several spikes but no jumps. Both spikes and jumps are areas of high relevance to SPH applications.

More accurate, stable and computer efficient results in SPH modeling and simulation will benefit different areas of science in several ways. In the medical field a more realistic, real-time simulation of an artery puncture will emphasize the potential severity and underline the demand for a correct and timely response to the situation. In costal engineering a more realistic result of splashing and breaking waves from tsunamis will assist to improve planning and preparation to reduce the damaging consequences of such phenomenon.

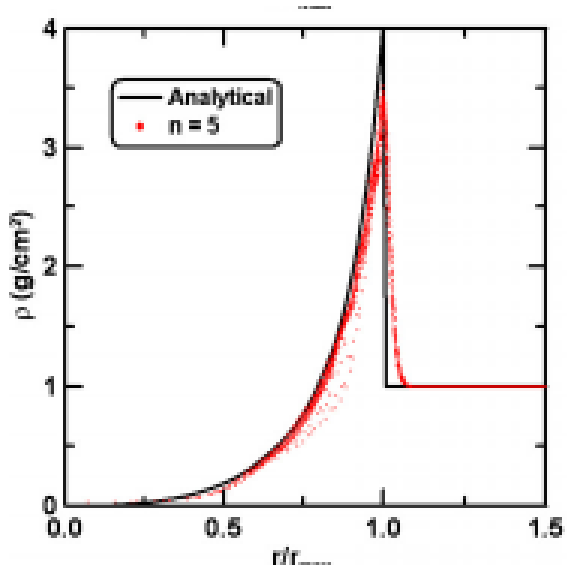

Figure 1.5: A shock wave front (black, solid line) and SPH attempted recreation (red, dotted line), (from Cabezón et al. [2008])

The need for better methods to more accurately model flow was clear. In the next chapter, we provide further details and examples on these topics to contextualize our approach and provide

foundation for our work by referring to historical elements of flow analysis, aspects of physics models and parameters, and mathematical methods of relevance to our chosen approach.

CHAPTER 2
BACKGROUND

2.1 Modeling Flow

The flow of matters is a phenomenon of relevance to a wide area of science. As found in our literature review, it is, among others, present in astrophysics, medicine, hydrodynamics, geology, metals processing, and animation. The matters in question represent gases, fluids, and solids or any combination of the three. The flow of such matters varies depending on the nature of the matters as well as the surrounding factors such as forces, temperature and physical restrictions. The flow of matters may be described as the changes taking place in the matters over time as they are impacted by one or more external factors. Such changes are typical described in terms of specific matter characteristics such as density, mass, and energy, over the time of the flow.

Mathematical modeling of the flow of matter has been used for a long time by many scientific disciplines. A variety of factors about the characteristics of the matter and its surroundings influence such modeling. This has lead to the development of different models for different applications as well as alternative models for similar applications.

This thesis was inspired by medical applications, ranging from fluid flow to mixed media and porous bones, and as such it represent models that have to take into account types of matters, Newtonian vs non-Newtonian fluids, viscosity, shear factors and turbulence, mixed media and boundary conditions. But the applications and models from medicine are just as relevant to the rest of science. In physics, several phenomena causing less smooth flow have been studied by

a combination of mathematical and physical models to analyze their consequences. Laboratory models for recreating changes in outcome as relevant input parameters are varied have been made and are still frequently used for verification purposes, including for SPH analysis. Examples of such models are Darcy flow models for flow through porous media. This was used by Basafa et al. [2013] for verification of their SPH results in case of cement infiltration of cancellous bone. A model for oscillating flow in a tube and flow in a lid-driven cavity were used for modeling and testing of SPH based simulations of pulsatile blood flow inside a left heart cavity [Shahriari et al., 2012]. The model for oscillating flow in a tube is also referred to as a Womersley type of flow. Both of these two latter models use the so-called Reynolds number (defined as the ratio between inertial force and viscous force) to vary the flow turbulence. This was used to mimic the flow conditions of a human heart [Shahriari et al., 2012].

Before SPH was introduced [Lucy, 1977, Gingold and Monaghan, 1977], the most common methods for modeling flow were finite-difference methods (FDM) and finite-elements method (FEM) based on grids, also called meshes. FDM uses a fixed grid with regular shaped grid cells covering the entire potential area where the matter may flow. FEM uses a grid of irregular shaped cells that covers the matter and moves with the matter. FEM was introduced to the industry early 1940 [Widas, 1997] for design and testing of structures prior to manufacturing. FDM is an Eulerian method where the flow of matter is observed from a fixed grid point in opposite to FEM where the flow is observed from a grid following the matter and hence an Lagrangian method. The Eulerian equations describing the flow in terms of mass, momentum and energy, respectively, are:

Mass

$$\frac{\partial \rho}{\partial t} + \nu^{\beta} \frac{\partial \rho}{\partial x^{\beta}} = -\rho \frac{\partial \nu^{\beta}}{\partial x^{\beta}} \tag{2.1}$$

Momentum

$$\frac{\partial \nu^{\beta}}{\partial t} + \nu^{\alpha} \frac{\partial \nu^{\beta}}{\partial \nu^{\alpha}} = -\frac{1}{\rho} \frac{\partial p}{\partial x^{\beta}} \tag{2.2}$$

Energy

$$\frac{\partial e}{\partial t} + \nu^{\beta} \frac{\partial e}{\partial x^{\beta}} = -\frac{p}{\rho} \frac{\partial \nu^{\beta}}{\partial x^{\beta}} \tag{2.3}$$

The Lagrangian equations for flow in terms of mass, momentum and energy, respectively, are:

Mass

$$\frac{D\rho}{Dt} = -\rho \frac{\partial \nu^{\beta}}{\partial x^{\beta}} \tag{2.4}$$

Momentum

$$\frac{D\nu^{\beta}}{Dt} = -\frac{1}{\rho} \frac{\partial p}{\partial x^{\beta}} \tag{2.5}$$

Energy

$$\frac{De}{Dt} = -\frac{p}{\rho} \frac{\partial \nu^{\beta}}{\partial x^{\beta}} \tag{2.6}$$

where ρ and e are density and internal energy, respectively, ν and x are velocity and position vectors, respectively, and α and β denote coordinate directions, [Liu and Liu, 2003].

In both cases change in physical properties to describe the flow of the matter are calculated by solving partial differential equations (PDE). Changes in location of the FEM nodes due to material movement requires recalculation of each node's position in the grid. When using FDM, the fixed grid may fail to accurately represent the matter as it flows, resulting in over-sampling or under-sampling as the nodes in the grid also are fixed. This, combined with the fact that FEM

and FDM both involve solving PDEs, makes it less favorable compared to SPH, in particular in case of less smooth flow. With SPH, the particle representation eliminates the need for a grid and the resulting differential equations are of ordinary form (ODE). The computational gain of solving ODEs as opposed to PDEs is due to the fact that the equation for the unknown function in an ODE depends on one independent variable whereas in a PDE it depends on two or more independent variables and hence requires two or more sets of equations to be solved simultaneously.

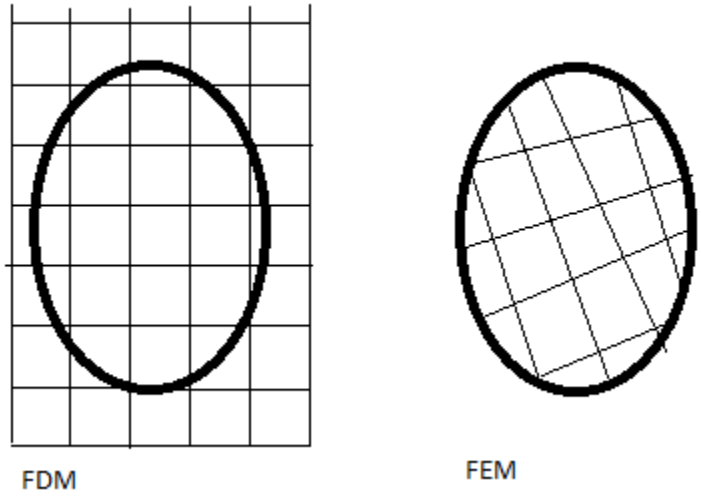

Figure 2.1: FDM (left) uses a regular stationary grid, and FEM (right) uses an irregular grid that moves with the matter

According to Monaghan [2005], the SPH method is superior to FDM in terms of accuracy, being Galilean invariant and bridging the gap between continuous and discrete representation. Being *Galilean invariant* means that the laws of physics are the same in all inertial frames of reference [Fowler, 2008], the reference system is stationary, and it does not change or introduce any force - as opposed to, for example, a rotating reference system. For non-smooth, shock type of

models, the computational benefits with SPH compared to FDM are significant [Monaghan, 2005], as a sample of 10^7 may be reduced to 10^3 for a one-dimensional model without loss of accuracy. Higher dimensions require far higher sample numbers and hence FDM tend to become prohibitive for efficient multi-dimensional analysis of such cases.

2.2 Smoothed Particle Hydrodynamics

The SPH method is increasingly used to model physical flow of matters in astrophysics, medicine, animation, and other fields related to material flow. It represents an alternative to the grid based methods. SPH has proven to be superior to grid based methods for some applications in terms of being less rigid as well as eliminating the need for grid recalculation as it relies only on the sample of particles representing the matter. It further reduces the calculations from solving a set of PDE to solving a set of ODE. These computational benefits have resulted in real time SPH simulations of applications where the grid based method have not been able to do so. Real time applications are of high interest in medicine [Müller et al., 2004] and animation [Xu et al., 2014, Takamatsu and Kanai, 2011] as well as many other fields.

When SPH was first introduced in astrophysics, non-smooth flow was acknowledged as a challenge to the method. Both Lucy [1977] and Gingold and Monaghan [1977] as well as Monaghan [2005] dealt with this issue by introducing viscosity parameters into the physical equations. As the applications expanded to include additional areas of science such as medicine, animation, geophysics, metals and forging, etc., the characteristics of the materials to be simulated became

highly different from that of astrophysics and the SPH method needed to be able to handle more uneven flow, less homogeneous matters as well as different flow restriction conditions. A general SPH model consists of:

- A function $A(r)$ representing a physical property at location r
- Application of a smoothing function, $W(r, h)$, under integration: $\int A(r)W(r, h)dr$, where h is the smoothing length
- Numerical calculation of $\sum A(r)W(r, h)$

and the properties are calculated by:

$$A(r) = \int A(r')W(r - r', h)dr' \doteq \sum_b \frac{m_b A(r_b)}{\rho_b} A(r)W(r - r_b, h) \tag{2.7}$$

where b denotes the numbering of particles included by the smoothing length h, m_b is the mass of particle b, and ρ_b is the density of particle b.

Most research publications regarding SPH include a thorough discussion of the kernel function in terms of selection criteria, needed for case specific adjustments and methods for verification of results. Factors that seem to influence the selection are viscosity [Müller et al., 2004], shear factor, particle distribution [Monaghan, 2005],[Basafa et al., 2013], and boundary conditions [Sinnott et al., 2012]. A *kernel function* is a kind of averaging function [Wasserman, 2006]. Alternative names for a kernel function are window, weighting or smoothing function. The kernel function is commonly used in SPH as well as in statistical analysis of probability distribution functions.

The kernel function directly influences the results of the SPH analysis. Different kernel functions may lead to different results in terms of accuracy and stability despite the fact that they all satisfy the criteria established for SPH kernel functions [Liu and Liu, 2003].

These criteria are:

- $Unity$: When integrated over their supported domain, the integral should be one
- $Compact\ support$: The smoothing function takes the value zero for all input values (i.e. all particles) located further away than κh from the particle of evaluation (κ is a constant and h is the smoothing length)
- $Positivity$: The smoothing function does not take negative values for any input value in its supported domain
- $Monotone\ decreasing$: As the distance from the particle under evaluation increases within the supported domain, the value of the smoothing function should decrease.
- $Delta\ function\ property$: The limit of the smoothing function as the smoothing length h approaches zero should satisfy the Dirac delta function
- $Even$: The smoothing function should be an even function
- $Smoothness$: The smoothing function should be sufficiently smooth to avoid sensitivity to particle disorder. For more extreme particle disorder and/or boundary issues, additional actions are required.

The most commonly used kernel functions in SPH are [Liu and Liu, 2003]:

- $Splines$, in particular the spline of order 3 defined as

$$M_3 = \alpha_d \begin{cases} (\frac{2}{3} - r^2 + \frac{1}{2}r^3) & 0 \leq r < 1 \\ \frac{1}{6}(2-r)^3 & 1 \leq r < 2 \\ 0 & r \geq 2 \end{cases}$$

  where α_d is $\frac{1}{h}$, $\frac{15}{7\pi h^2}$ or $\frac{3}{2\pi h^3}$ for one, two or three dimensional case. The parameter h defines the smoothing length, the compact area of evaluation.

- $Gaussian$ based (Modified Gaussian to achieve compactness)

$$G_s = \alpha_d \begin{cases} (\frac{3}{2} - r^2)e^{-r^2} & 0 \leq r < 2 \\ 0 & \text{else} \end{cases}$$

  where α_d is $\frac{1}{\sqrt{\pi}}$ in the one dimensional case.

- $Wendland\ M_5$, defined as

$$W_5 = W_0(1+2s)(2-s^4)$$

where $s = \frac{|r|}{h}$, $W_0 = \frac{7}{2^6\pi h^2}$ and $\frac{7}{2^\pi h^3}$, (Rafiee et al. [2012]).

These are all smooth functions and work well in interpolation of smooth flow. However, in cases of less smooth flow, such as jumps and spikes, these kernel functions fail to accurately recreate the flow as jumps and spikes get smoothed out. Cabezón et al. [2008], as they used the kernel function to seek improve SPH for less smooth flows, used the family of functions expressed

by the $sinc$ function

$$W_n = \left\{ \frac{sin^n(\frac{\pi x}{2})}{(\frac{\pi x}{2})^n} \right\}_{n \in \mathbb{N}}, \tag{2.8}$$

where different powers of n makes up the family of functions. Figure 2.2 shows an example of this kernel function with $n = 5$. They found that such a family represented flexibility to use different kernel functions for one and the same simulation by varying n depending on variation in density and smoothness of the matter. Increasing the power of n would increase the peakiness of the kernel function. As can bee seen from their results (Figure 1.5), this seemed to increase accuracy in areas of low particle density, but actual jumps became over-smoothed. Further, they observed that noise increased prohibitively with increasing values of n. Some increased computational efficiency were obtained by pre-calculating values of the $sinc$ function for input at different n values.

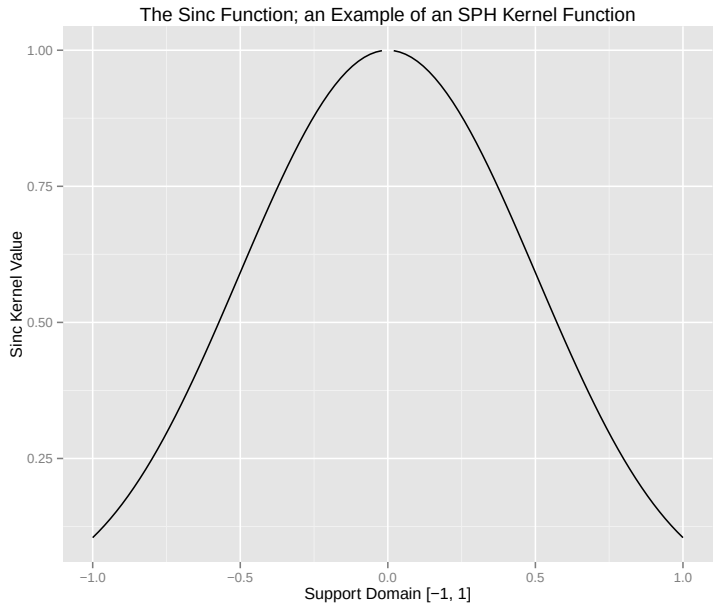

Figure 2.2: The sinc functions of power $n = 5$ and smoothing length $h = [-1, 1]$

The challenge in SPH of more accurate and efficient estimate flows of non-smooth nature, flows with jumps and spikes, triggered the idea of seeking to improve recreation by applying what is proven optimal in IFT. It has not been possible to find any research that has looked at SPH and IFT to compare interpolation operations. We will discuss IFT in more detail in section 2.4.

2.3 Numerical Realization, Accuracy and Stability

In numerical analysis any numerical realization must fulfill the requirements of being a well-posed problem. That is, the numerical problem must meet the following three criteria:

- A solution exists,
- The solution is unique,
- The problem is stable. That is: Errors in input lessen in significance as the algorithm executes.

The overall order of accuracy of the SPH method has typically been evaluated separately for each case as it is influenced by factors specific to the case in question. For evaluation purposes, these factors may be categorized as:

- The accuracy of the physical model prior to applying the smoothing function,
- The selection of smoothing function,
- The accuracy of the integration (continuous case),

- The accuracy of the numerical integration (discrete case),
- The particle distribution.

The order of accuracy of the physical model is ensured by using sound physical expressions based on the Navier-Stokes equations [Liu and Liu, 2003], expressed by

Continuity:

$$D\rho Dt = -\rho \frac{\partial \nu^{\beta}}{\partial x^{\beta}} \tag{2.9}$$

Momentum:

$$\frac{D\nu^{\alpha}}{Dt} = -\frac{1}{\rho}\frac{\partial \sigma^{\alpha\beta}}{\partial x^{\beta}} \tag{2.10}$$

Energy:

$$\frac{De}{Dt} = -\frac{\sigma^{\alpha\beta}}{\rho}\frac{\partial \nu^{\alpha}}{\partial x^{\beta}} \tag{2.11}$$

where notations are as in eqs. 2.4 - 2.6, and σ denotes the total stress tensor which is the combination of isotopic and viscous stress, [Liu and Liu, 2003].

The order of accuracy of the integration is an analytic evaluation derived by a Taylor expansion of the physical function in question. The order of accuracy of the numerical integration depends on the particle distribution [Monaghan, 1992] and is at least of order $O(h^2)$ for a symmetric smoothing function and may increase to $O(h^4)$ with a symmetric smoothing function which second derivative equals zero. This means that the accuracy of the reconstructed $\tilde{A}(r)$ of $A(r)$ depends on h by

$$\frac{|A(r) - \tilde{A}(r)|}{h^2} \leq M_1$$

and

$$\frac{|A(r) - \tilde{A}(r)|}{h^4} \leq M_2$$

respectively, where h is the kernel function's compactly supported domain, also referred to as *smoothing length*, M_1 and M_2 are constants of finite value.

According to Kincaid and Cheney [2002], stability in numerical integration is achieved by an implicit multi-step method of order two or less, and uniqueness is achieved if the function to be integrated and its first derivative satisfy the Lipschitz condition, which are certain continuity requirements of the function on the domain of reference.

In our work, the focus on accuracy was related to the segmentation, smoothing and, to some degree, sample size. We borrowed ideas from IFT to improve these accuracies.

2.4 Information Field Theory (IFT)

Information field theory refers to the theory of information analysis over a mathematical field. It is often combined with Bayesian analysis. In general mathematical terms a $field$ is a set with the two operations addition and multiplication defined and satisfying certain axioms, including commutative, associative and distributive laws, [Rudin, 1976]. Information field theory focuses on analysis of physical fields over continuous spaces [Ensslin, 1977]. In physics, the density of the ocean is a physical field over the continuous space ocean. Example of statistical field theory is the analysis of probability density functions based on finite set of observations, where the density functions represent the field [Lemm, 2003]. For real life applications, a finite set of obser-

vations is used to draw conclusions about the continuous space from where the observations are sampled. These observations may represent one dimensional signals, two-dimensional images, or other multi-dimensional objects. In this work, they represent particles of the matter to be analyzed.

In order to improve the results of IFT, Bayesian field theory is often used. According to Lemm [2003], Bayesian field theory is the information field theory where nonparametric Bayesian models are used to describe the field. The Bayesian element is commonly used to seek to improve the probability of a true representation by incorporating an already known or assumed prior probability distribution about the continuous space. This has proven valuable in both IFT as well as statistical density estimations, [Pensky, 1999], [Ensslin, 2014], [Lu and Tjöstheim, 2014]. The Bayesian element will be left for potential future PhD thesis research intended as a continuation of this work. Since this work does not address this directly, a detailed mathematical description is outside the scope of this thesis, but may be found in Lemm [2003] and Gelman et al. [2014]. Instead, we discuss wavelets in more detail since that topic is germane to this thesis.

2.5 Wavelets

A wavelet is a small wave described by a function that over a limited domain increases and decreases before returning to zero, as opposed to a wave such as $sin(x)$ that continues to oscillate on the entire real line. In wavelet theory wavelets are a class of functions used to localize a given function in both space and scaling. A family of wavelets can be constructed from a function

$$\psi(x)$$

sometimes known as a "mother wavelet", which is confined in a finite interval [Weisstein, 2015]. "Daughter wavelets", denoted

$$\psi_{a,b}(t)$$

are then formed by $translation$ (b) and $contraction$ (a), forming a wavelet family

$$\psi = \{\psi_{a,b}(t)\}_{a\in\mathbb{R},b\in\mathbb{R}^+},$$

where different values of a and b define the scaling and shifting resolution to represent the original function, i.e. a signal, picture or other object, to be analyzed.

The preferred wavelet functions used for interpolation in IFT are a family of functions that represent an orthogonal vector basis, or close to orthogonal, for the functions they are to interpolate. A $vector\ basis$, $\{v_k\}_{k=1,2,...m}$, for a space is a set of m independent vectors, where m is the maximum number of independent vectors possible for the space. All other vectors for this space will be linear combinations of its basis vectors $\{v_k\}$. When this set of vectors is orthogonal, the linear transformation represented by a matrix, V, satisfy $V^TV = VV^T = I$ and $V^T = V^{-1}$, I being the identity matrix, V^T the transpose of V and V^{-1} the inverse of V. This means the calculations for solving a set of linear equations can be done without matrix inversion, which eliminates the need for multiplications required for inverting a matrix.

The construction of wavelets ranges from wavelet families for non-smooth functions represented by the Haar functions to the most smooth functions such as Shannon wavelets [Mallat, 2009]. Figure (2.3) shows the Haar wavelet with alternative scaling and shifting values.

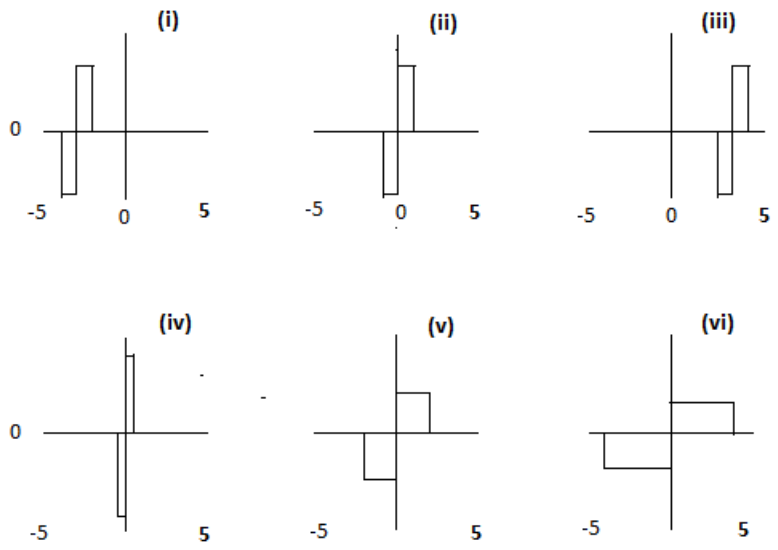

Figure 2.3: Haar wavelet with different scaling and shifting values, (i) $\psi_{1,-3}$, (ii) $\psi_{1,0}$, (iii) $\psi_{1,3}$, (iv) $\psi_{1/2,0}$, (v) $\psi_{2,0}$, (vi) $\psi_{4,0}$

Wavelet analysis is a key element for reconstruction of functions belonging to a continuous space based on a finite sample at hand. The properties of wavelets were used in this work to analyze functions and real life data sets for jumps and spikes.

The main criterion used for evaluating wavelets is the Riesz condition [Mallat, 2009], i.e. a measure for how close the family of wavelets is to an orthonormal basis in L^2, that is, for all the functions $f \in L^2$ such that

$$\int_{-\infty,+\infty} |f(x)|^2 dx < \infty$$

A family of vectors $\{\psi_n\}_{n\in\mathbb{N}}$, is a Riesz basis for Hilbert space H if it is linearly independent and

$$\exists B \geq A > 0$$

such that

$$\forall f \in H, \;\; A \parallel f \parallel^2 \leq \sum |\langle f, \psi_n \rangle|^2 \leq B \parallel f \parallel^2, \;\; n = 1, 2, 3, \ldots$$

If $A = B$, then the basis is orthogonal.

In addition, the function ψ has to satisfy the so-called admissibility condition

$$C_\psi = \int_{-\infty,+\infty} \frac{|\hat{\psi}(\omega)|^2}{|\omega|} d\omega < \infty$$

where $\hat{\psi}(\omega)$ is the Fourier transform of $\psi(t)$. It is further a necessary condition that

$$\hat{\psi}(0) = 0 \Leftrightarrow \int_{-\infty,+\infty} \psi(x) dx = 0$$

and sufficient condition that if

$$\int_{-\infty,+\infty} \psi(x) dx = 0$$

and

$$\int_{-\infty,+\infty} (1 + |x|) \psi(x) dx < \infty$$

then $\hat{\psi}(\omega)$ is continuously differentiable and $C_\psi < \infty$.

As with kernel functions, there are wavelets with different characteristics regarding smoothness, from the Haar wavelets (piecewise constant functions) to the Shannon wavelets (smooth) to wavelets in between such as the Daubechies wavelets and Battle-Lemarie wavelets, as well as many more. Daubechies and symmlet wavelets are shown in Figure (2.4).

The majority of work related to wavelets, both research and applications, has been for signal compression, de-noising and recreation. The primary use of wavelets in our work was for

jumps and spikes analysis. Earlier work in this area [Abramovich et al., 2007] used symmlets. As far as we were able to verify, it was not known if the criteria for wavelet selections for signal processing would translate directly to our application. Hence, we decided to evaluate several of the most common ones to determine which would provide the most accurate results for our purpose. In particular, we decided to include the Coiflet family of wavelets as these too represent wavelets with minimal phase distortion. As we would only used the wavelets for jump and spike identification, we don't know if phase distortion would be of importance, as opposite to if they were used for function recreation where it might lead to inaccuracy in terms of preservation of mass, momentum and/or energy.

Given the theory and applications of wavelets to IFT, it was natural to consider ideas from this field for use in SPH applications. The next two chapters will provide more details on how this was studied as well as some preliminary results. A more in-dept study of wavelets may be found in Mallat [2009] and Percival and Walden [2008].

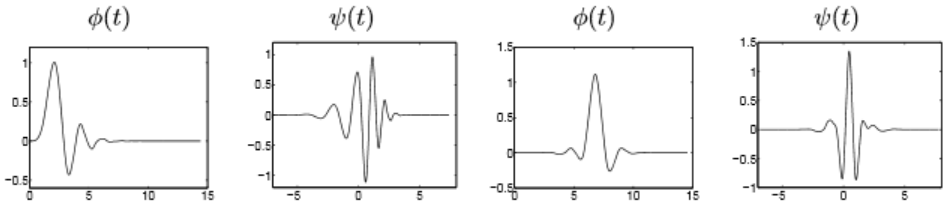

Figure 2.4: The Daubechies wavelet (graph one and two from left), and symmlet wavelets (graph three and four from left), (from Mallat [2009])

CHAPTER 3
METHOD

Previous work in SPH to improve accuracy when flows are less smooth has mainly focused on two approaches; modification of physical equations and fine tuning of kernel functions properties. The method in this work did introduce the use of wavelets and DWT (Discrete Wavelet Transformation) to SPH for the first time. As described by Abramovich et al. [2007], this method combines use of DWT and kernel functions by first identifying any jumps the function may have and a then interpolating the function segment-by-segment by use of an amalgamated bridge regression spline (ABS). A regression spline, or simply a spline in the context of this work, is a piece-wise polynomial function. An amalgamated spline is a spline where the piece-wise polynomial functions may differ from segment to segment. Among the splines, B-splines are splines generated from a basis of independent vectors. B-spline bases are preferred due to their rather stable numerical properties compared to other splines and was therefore selected for our purpose.

We structured our research into the following seven steps:

- Wavelet transform of the function/data
- Sorting of DWT coefficients according to absolute value in descending order
- Identification of DWT coefficients representing jumps and spikes by thresholding
- Locate jumps and spikes
- Segmenting function/data according to jumps and spikes

- Segment-wise interpolation in parallel

- Re-create function/data and compare accuracy to original function/data

The approach of our work was empirical testing by computer simulations in one dimension, supported by analytic evaluations where such analysis was expected to provide additional information to the simulation results and practical for the scope of this research. The method was preliminary tested on a standard function commonly used for wavelet evaluation and interpolation, as well as an applications in medicine (ECG), (Figure 3.1). The main research focused on additional two standard functions and two sets of real life data, water flow (USGS National Water Information System, Florida) and recordings of human pulmonary artery pressure (PAP). We also recreated and analyzed data representing a shock wave in homogeneous media [Cabezón et al., 2008], an example where the authors cited accuracy issues the way they used the SPH method, (Figure 1.5). Our intention to include at least one more examples directly from an SPH application on breaking water waves from [Khayyer et al., 2007] was excluded partly due to lack of access to the data and partly due to time constraints.

Some of the applications included functions with less smooth segments between jumps, in particular functions with spikes. While the method we chose is described and tested for jump identification, we also attempted to identify spikes by the same method. This did test the method beyond the work of Abramovich et al. [2007]. A list of the functions and data sets analyzed in our work is included, (Table 3.1).

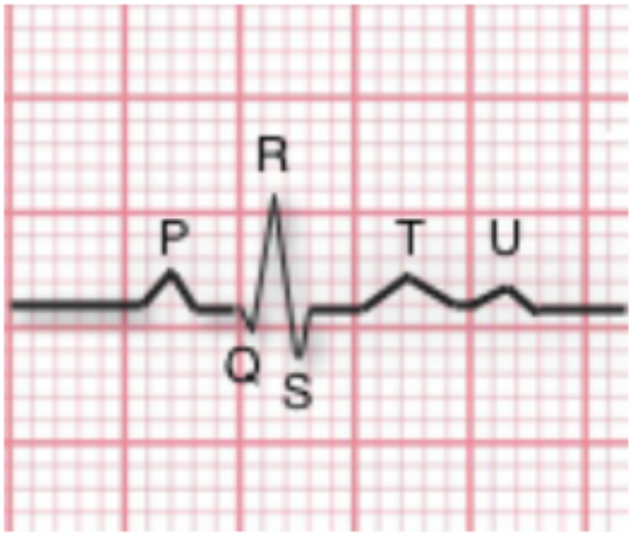

Figure 3.1: ECG recording of a normal heartbeat, often referred to as the 'PQRSTU' complex, (from www.Practicalclinicalskills.com)

A mathematical definition of the kind of jumps analyzed in this work may be expressed as

$$lim_{x \to x_0^-} f(x) \leq L_1 \tag{3.1}$$

$$lim_{x \to x_0^+} f(x) \leq L_2 \tag{3.2}$$

where $L_1 \neq L_2$ are finite limits as x approaches x_0 from left and right, respectively. These are all finite jumps as the limits are finite. Spikes also represent rather sudden changes in the functions values. While jumps are defined by use of limits, spikes are defined by the fact that the derivative of the function does not exist. For our work we included functions with spikes of different absolute and relative amplitude and analyzed their effect on the method's ability to identify the spikes and also to discriminate between jumps vs spikes.

The results were quantified in terms of accuracy, stability and computer efficiency.

Table 3.1: Functions for jump identification and reconstruction

Functions	Math Formula/Recordings	Jumps	Spikes	Comments
Segment-wise Cosine	$cosine(x) = cos(5.5\pi x) - 4sign(.23 - x) - 2sign(.3 - x) - 1.75sign(.55 - x) + 3sign(.7 - x)$	4	none	Figure 3.4
Piece-wise Regular	From Matlab	4	3	Figure 3.2
Piece-wise Polynomial	From Matlab	8	none	Figure 3.3
ECG-normal	Recordings	none	6	Figure 3.1
Water Levels	Recordings			Figure 4.3
Pulmonary Artery Pressure	Recordings			Figures 4.8, 4.9
Shock wave	$P(t) = P_2 + (P_1 - P_2)exp(-t^2/\sigma^2)$	1	none	Figure 1.5, P, P_1, P_2 denote pressure, σ constant

3.1 Identifying Jumps Using Standard Functions

Jump identification was tested on known functions by selecting functions that are already regarded as standard functions for testing related to mathematics, statistics, computer science and/or physics. The four standard functions named as *block, burt, heavisine* and *segment-wise cosine* (called *cosine with jumps* in their work) have already been used by Abramovich et al. [2007], (Figure 1.3). We selected two different functions for our analysis, the *piece-wise regular* and the *piece-wise polynomial* from wavelet analysis [Mallat, 2009], (Figures 3.2 and 3.3). Both these functions have multiple jumps. The piece-wise regular function also contains spikes. We also included the *segment-wise cosine* function for verification purposes. Data for these three standard functions were created in Matlab.

From Abramovich et al. [2007] we know that jumps are identified by coefficients of relatively large absolute values. This result was duplicated in our preliminary work as we analyzed the *segment-wise cosine* function. With a wavelet transform of this function using the coiflet 6 wavelet, the five coefficients with largest absolute value occurred located where the function has its five jumps. Further, we used thresholding and counted the number of coefficients of value different from zero, D, we got $D = 5$, which is the actual number of jumps for this function. We used a sample size of $n = 512$, which we kept through the entire work, supplemented with $n = 2048$ when a better understanding of how sample size did influence the results was needed.

Our approach was as follows:

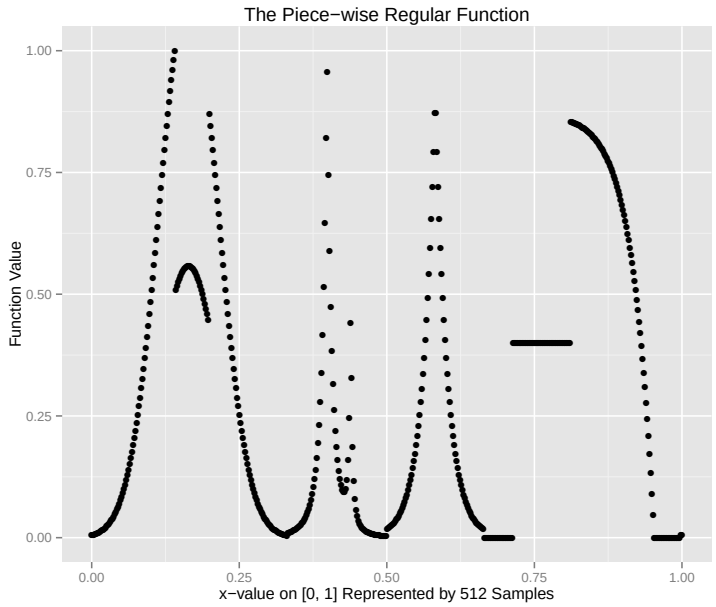

Figure 3.2: The piece-wise regular function, containing four jumps and three spikes

Figure 3.3: The piece-wise polynomial function, containing eight jumps

- A brief mathematical description of the standard functions selected in terms of jumps, spikes, compactness and segment-wise integrability.

- Analysis and selection of wavelets. The symmlet family of wavelets is often preferred in compression and reconstruction of functions due to their almost linear phase, close to symmetric energy and only real coefficients. To what degree this would be of importance to jump detection was tested by including five wavelets in addition to symmlets.

- Analysis and identification of jumps using the above selected wavelets. Results were compared in terms of the wavelet's ability to detection correct number of jumps and their location. Two thresholding methods were tested and the results discussed in terms of their ability to ensure that only coefficients not representing jumps were zeroed out.

Figure 3.4: The segment-wise cosine function, containing five jumps

- The jump detection techniques were evaluated with respect to:
  - Different wavelets' ability to most accurate detect location of jumps
  - The importance of jump magnitude for accurate detection
  - The method's sensitivity to threshold level and noise
  - Importance of absolute or relative difference in coefficients' value for jump identification
  - Trade-offs, if any, between accuracy and number of filter coefficients to keep number of multiplications as low as possible and still satisfy jump location accuracy

The results was measured by how accurate the jumps were located by comparing the simulated results to the actual standard functions. The results are presented in chapter four, first section.

3.2 Identifying Jumps in Applications from Medicine and Water Flow

The method and results from identification of jumps in standard functions was applied to real data recordings from medicine and water flow. The water flow data are from USGS National Water Information System, Florida. From medicine, two cases of human pulmonary artery pressure were selected. Such recordings are available from PhysioNet, (http://www.physionet.org). Data for water flow are available from USGS Water Resources for all of the fifty states.

The method followed the steps as described above for standard functions. The results are presented in chapter four, second section.

3.3 Jump Identification vs. Identification of Spikes

Preliminary results from this work indicated that the wavelet coefficients in case of jump are of larger absolute value than wavelet coefficients of spikes with same magnitude. An ECG signal from a normal heart function was used in the preliminary work. A jump was artificially created by replacing the signal's largest spikes, the R-S transition which is of rather short duration, by a jump of the same magnitude. This resulted in a wavelet coefficient of larger absolute value than when the normal ECG signal was analyzed by the same method. Our analysis of spikes were limited to the spikes of the piece-wise regular function and supported the preliminary findings from the ECG signal.

Focus was on:

- Discrimination between jumps and spikes based on wavelet coefficients' absolute value
- Difference in coefficients' absolute value for jumps vs spikes
- Absolute and relative magnitude of jumps and spikes and their influence on discrimination capability
- Other properties found to impact the results, such as function smoothness, sampling frequency and type of wavelet used.

The results were measured by number of spikes correctly detected, difference in wavelet coefficients values for jumps compared to spikes and related to magnitudes of spikes vs. jumps. The results are presented in chapter four, third and seventh section.

3.4 Interpolation of Standard Functions

From a mathematical point of view both wavelets, splines and kernel functions are used for interpolation. Based on earlier results [Mallat, 2009], [Abramovich et al., 2007], [Wasserman, 2006], the wavelet method tend to introduce so-called Gibbs oscillations when used to recreate jumps and spikes. Such oscillations are also known to occur in interpolation in general, in particular when less smooth segments of a function are attempted interpolated. We decided to use splines for function recreation both because of their stability properties and in order to compare our results to the results of Abramovich et al. [2007].

The same three standard functions used for jump identification were used for segment-wise and non-segmented recreation.

The results were measured by the difference of the recreated function compared to the known standard function using L_2 norm

$$\frac{1}{n}||f(x) - \tilde{f}(x)||_2 \tag{3.3}$$

for $x = (x_1, x_2, ..., x_n)$, n being the sample size, and f and $\tilde{f}$ the standard function and recreated function, respectively. This measure is referred to as accuracy or error in our work. The results are presented in chapter four, fourth section.

3.5 Kernel Selection Criteria for Interpolation

The importance of kernel function selection seem to vary by authors. Wasserman [2006] states in his book on non-parametric statistics that the estimates from different kernels are usually numerical very similar and hence not very sensitive to the kernel. Liu and Liu [2003] on the other hand stress the fact that a kernel like the M_3 spline with a piece-wise linear second derivative may cause stability issues. Such difference in view may be related to areas of application as the majority of SPH related research refers to M_3 as a relevant choice.

Our initial intention was to test the importance of kernel selection by applying different kernels to the same interpolation operation and compare results in terms of accuracy, stability and computational efficiency. The idea was to select a number of kernels that have been regarded as successful when applied to specific interpolation operations in previous SPH research. While this was not systematically explored due to time constraints, our research did include one specific example, the sinc kernel function used for interpolation of shock wave by Cabezón et al. [2008]. This is still a relevant problem and the idea would be to use different interpolation functions for segment-wise recreation of the functions and data sets. The results from using the M_3 spline and

additional splines frequently used in SPH, would be compared to the results of our work using B-splines.

However, we did evaluate the benefits of segment-wise function recreation compared to non-segmented recreation. Non-segmented recreation is the traditional method used in SPH, independent of kernel selection. The results are presented in terms of accuracy using the L_2 norm as defined above. The method's stability was evaluated based on the stability criteria from Kincaid and Cheney [2002]. Computational efficiency was measured in terms of multiplications and CPU time. Memory usage can also be an indication of efficiency, but will not be used for this work due to inaccuracy in measuring technique of our setup.

A summary of accuracy, stability and computational efficiency results is presented in the result chapter, eight section. Specific results are also included throughout the result chapter as relevant to the topics presented in the specific sections.

3.6 Application Of Jump Identification and Interpolation Results To SPH Related Functions

Of the numerous references to previous research attempting to improve accuracy, stability and efficiency through optimizing kernel function properties, we used the shock wave problem from Cabezón et al. [2008]. This was an example where attempts to improve the interpolation in SPH had already been tested and because the analytic function describing the wave propagation was accessible. A second problem related to breaking water waves as presented by Khayyer et al.

[2007], had to be excluded due to lack of data access. Both these problems represented the less smooth flow of high interest for our research.

The comparison focused on:

- Improvements achieved by using the two step method
- Potential areas of under-performance and their characteristics in terms of smoothness
- Recommendations for future SPH applications and ideas for further improvements

Also for this particular example, the results were measured by the methods described above. The results are presented in chapter four, fifth section.

3.7 Simulations

The simulations were carried out by using a combination of standard software packages and programming developed by the author to meet the specific needs of this work. We used the programming language R, the wavelet package "wavelets" version 0.3-0 and "smooth.spline" in the package "stats" version 3.2.2. The "wavelets" package contains all the wavelets we used except for symmlet 3 that was added by direct input of the wavelet coefficients as provided in the programming language Python as sym3.

Our program setup was first used to analyze the segment-wise cosine function for jump detection as well as function recreation, (Figure 1.3, cosine with jumps). The results we obtained

were in accordance with the original function, all five jumps were located within less than a sample size of its actual location and accuracy of the recreated function was of order 10^{-16} which is the order of machine epsilon for R.

CHAPTER 4
RESULTS

The method we used in this research is a two step method to: 1) identify any jumps a function or set of data may contain and use the jumps to segment the function or data set, and then 2) recreate the function or data sequence segment by segment. The results are presented accordingly.

4.1 Jumps Identification in Standard Functions

In summary, using a Discrete Wavelet Transforms (DWT) on three standard functions containing jumps, our method resulted in identification of the correct number of jumps as well as locating the jumps fairly accurately. Analyzing these benchmark functions, in particular the segment-wise cosine function, also served to verify our simulation model and to understand the significance of any parameters to be set before we used the method for analyzing real life data sets.

The ability to identify jumps was evaluated based on the correct number of jumps identified and the accuracy of the jump's location. Recall that in our method, a jump in a function is typically characterized by a DWT coefficient of larger absolute value than where there is no jump. A threshold value specific to the wavelet type was calculated and used to distinguish between DWT coefficients representing jumps vs. not representing jumps. Each jump's location was estimated and compared to the location of the actual jumps of the functions.

The three standard functions *piece-wise regular, piece-wise polynomial* and *segment-wise cosine* were used for jump identification (Figures 3.2, 3.3, 3.4, respectively). Six wavelets, *coiflet 6 (c6), Daubechies 6 (d6), coiflet 12 (c12), Daubechies 12 (d12), least asymmetric 12 (la12) and symmlet 3 (sym3)* were used for jump identification. The number behind a wavelet name states the number of coefficients, except for the symmlet 3 that has 6 coefficients. Symmlet 6 is commonly used in literature to refer to the la12 wavelet, so we retained the traditional naming convention for consistency within the literature.

4.1.1 Threshold Parameter Values

Two methods were used for calculating the threshold value, the False Discovery Rate (FDR) and the universal threshold level. The FDR method gave the most consistent results in terms of correct number of jumps. With this method, the threshold level, t_k^*, is decided using a confidence interval to distinguish between DWT coefficients located outside vs. inside the confidence interval, [Abramovich et al., 2006], given as:

$$t_k^* = \sigma * |z_{q/2*\frac{k}{n/2}}|, \tag{4.1}$$

where

$$|z_{q/2*\frac{k}{n/2}}|$$

is the right tail of the Gaussian quantiles of $z \sim N(0,1)$, q denotes the confidence level, $k \in \{1, 2, ...\frac{n}{2}\}$ and n is the number of observations. With this method, the DWT coefficients with absolute value located outside the specified confidence interval are highly likely to represent jumps, (Figure 4.1).

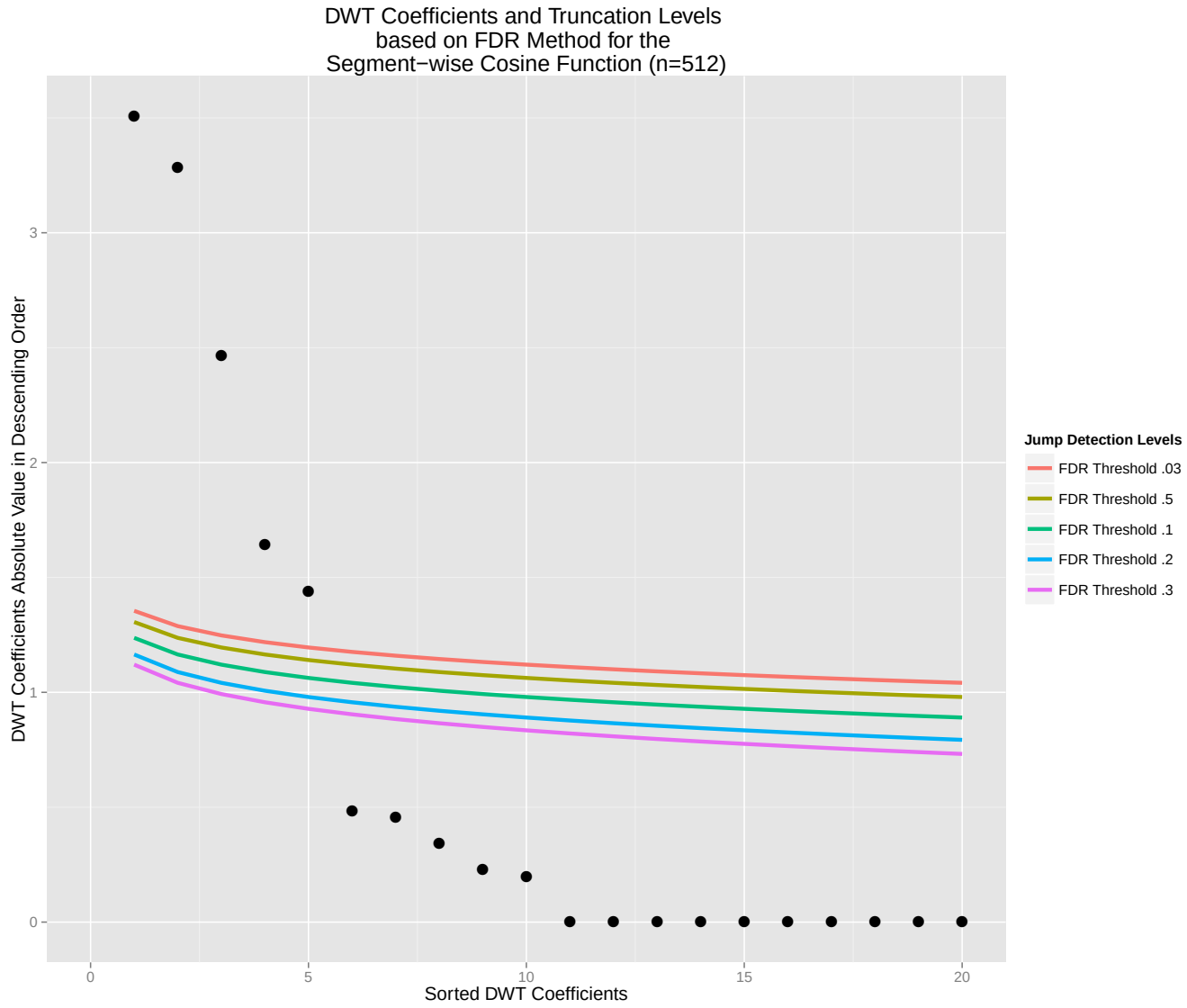

Figure 4.1: The five DWT coefficients representing the jumps of the segment-wise cosine function, using right tail threshold levels ($\frac{q}{2}$) of .15, .1, .05, .025 and .015, respectively

The universal threshold method referred to by Abramovich et al. [2007] is one of the more frequently used methods for thresholding of DWT coefficients. This method also worked for jump identification but did require some parameter adjustments. The method is given as

$$t_n^* = \sigma\sqrt{(\log n)^{1+\delta-2\alpha}}, \tag{4.2}$$

where σ is either the standard deviation of the noise, if known, or the median of the absolute deviation of the DWT coefficients at the highest resolution level divided by .6745. This is the median of the absolute value of standard normally distributed observations. The value n is the number of samples of the function, and α and δ are parameters of arbitrary small values recommended as $\delta > 0$ and $0 < \alpha < \frac{\delta}{2}$.

4.1.2 Jump Identification Results

Of the six different wavelet types, only c6 and c12 identified the correct number of jumps in all three standard functions when using the FDR threshold method. The c6 wavelet resulted in the most accurate jump location, $\leq .0029$ on the interval $[0, 1]$, with sample intervals of $\frac{1}{512} \approx .002$, (Figure 4.2). The c12 wavelet located jumps with an accuracy range of $[.0059 - .0068]$. The main difference between c6 and c12 is the number of coefficients. In general, a larger number of coefficients will smooth a function more than if fewer coefficients are used. This may be why a higher accuracy was achieved by the c6 wavelet than with the c12. When the universal threshold method was used, the same two wavelets, c6 and c12, also correctly identified the number of jumps. But this was only if the threshold value was properly adjusted by either varying the values of δ and

α or manually setting the threshold value greater than the first DWT coefficient not representing a jump as described in scenario 4 below.

By using the FDR threshold method, the sym3 wavelet correctly identified the number of jumps in the piece-wise regular and the segment-wise cosine function with a location accuracy as good as or better than the c6 wavelet, (Figure 4.2). It also identified two of the three spikes in the piece-wise regular function. It failed to identify any of the eight jumps in the piece-wise polynomial function within a location accuracy of .2 and hence excluded from further analysis of this function.

The d6, d12 and la12 wavelets did not identify the correct number of jumps, both because they failed to identify some jumps and also because they found jumps at locations where a function did not have any. A false jump was defined as a jump being located more than 10 sample steps (a deviation of .02 on a [0, 1] domain) of from its actual position. An increase in the location deviation to 20 sample steps (a deviation of .04 on a [0, 1] domain) allowed the d12 wavelet to correctly identify number of jumps in the Segment-wise Cosine and the Piece-wise Regular functions, but not the Piece-wise Polynomial function. In order to accept all jumps identified by d6, a jump location deviation in the order of 100 times the sample step size would have to be included, hence those jumps were regarded as fake jumps. The impact of jump location deviations was demonstrated as the functions were recreated during the second step of this work and are discussed in the Recreation of the Standard Functions section below.

The Haar wavelet and the Daubechies 4 (d4) were also tested, but with a less successful outcome and hence not investigated further.

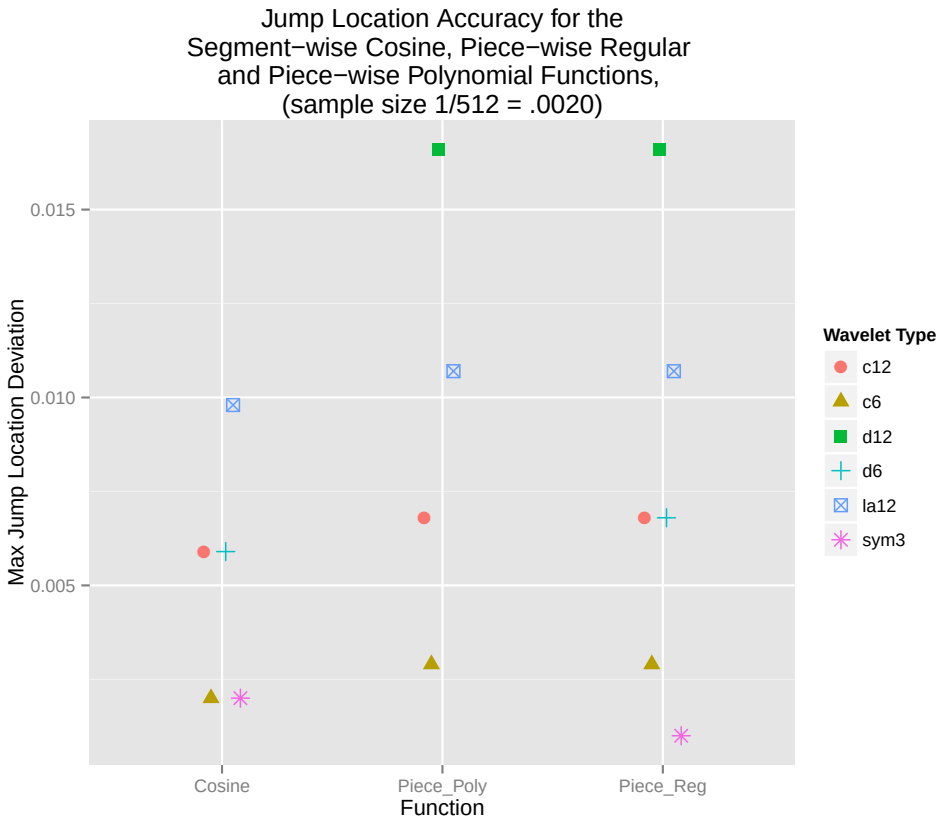

Figure 4.2: Jump location deviations for the three standard functions using six different wavelets

For the universal threshold level we tested four scenarios for t_n^* to seek a common threshold level that would serve to correctly identify number of jumps. In scenario 1, 2 and 3 the parameters δ and α were adjusted to achieve an increasing t_n^* value. In scenario 4 the threshold was adjusted manually. The particular scenarios values were:

1. $\delta = .5$, $\alpha = \frac{\delta}{2}$
2. $\delta = .5$, $\alpha = \frac{\delta}{4}$
3. $\delta = .8$, $\alpha = \frac{\delta}{4}$
4. selecting t_n^* as the largest DWT coefficient not representing a jump

Threshold scenario 1 provided the lowest threshold value. Scenario 4 was only selected if none of the other three provided correct result. As expected, the selection of parameter values for t_n^* directly influenced the method's ability to detect correct number of jumps in the functions. Table 4.1 shows what scenario(s) resulted in correct number of jumps for the standard functions.

Table 4.1: Correct jump identification for the standard functions by thresholding scenarios

Function	c6	d6	c12	d12	la12	sym3
Segment-wise Cosine	1	4	1	none	3	4
Piece-wise Reg	1	none	1	none	1, 2, 3	1, 2, 3
Piece-wise Poly	1	none	1	1	none	none

Based on these results the c6 and c12 wavelets were preferred, and the c6 was chosen for our analysis because of its superior jump location accuracy.

As the FDR threshold method did not require any prior information about the functions to be analyzed and performed superior to the universal threshold method, we decided to use the FDR method going forward with the real life data analysis.

4.2 Identifying Jumps in Real Life Data

The method used for identifying jumps in the standard functions was applied to sets of real life data using the c6 wavelet and FDR threshold method. With this method we were able to identify jumps fairly accurate also when applied to real life data. The results from these analyses provided additional information about the method's capability to locate jumps compared to the information gathered from analyzing the standard functions. In particular, the method did not identify all jumps and identified some jumps that were not real. Further, the method's sensitivity to both the DWT coefficients' standard deviation as well as the sample size was well demonstrated. When all other factors are kept unchanged, a lower standard deviation reduces the threshold level and leads to an increase in the number of DWT coefficients representing jumps. An increase in the number of samples has a lesser and opposite effect, as it leads to an increased confidence interval. The results from the real data analysis reflected these facts.

When analyzing digital data for jumps, it is worth keeping in mind that the mathematical definition of jumps by use of limits is not practical. A set of digital data is by nature discontinuous. So for our analysis of real life data we decided to compare any jumps identified to the corresponding difference in data values at the jump location. Such differences are referred to as

jump magnitudes. Even though this difference in jump definition is significant in theory, it is not a factor of importance in our work as the segmentation method uses both jumps and spikes. The results from recreating the piece-wise regular function, presented in section four of this chapter, demonstrates such an example.

4.2.1 USGS Data Jump Results

The first set of data was from USGS Water Resources Lochloosa, FL, of daily mean water discharge from May 10, 1947 to December 16, 1952 with values ranging from zero to 332 (Figure 4.3). The data set was analyzed by first using recordings from the 512 days from May 10, 1947 to October 2, 1948 and secondly by using all 2048 recordings from May 10, 1947 to December 16, 1952. For the first 512 days there were four jumps identified representing change in daily mean water discharge ranging from 23 to 41 cubic feet per second, with a location accuracy of $\leq .002$ using an one-sided confidence level of $.015$, ($q = .03$), (Figure 4.4). By increasing the confidence level to $q = .2$, additional 2 jumps of magnitude 23 and 27 cubic feet per second, respectively, were identified. One of these jumps, located at September 25, 1947, ($x = .2715$ on a domain $[0, 1]$ representing the 512 days range), represented the second of two consecutive jumps and located at the end of a two jump sequence, while the second one at January 25, 1948 ($x = .5098$) represented a jump in between two others, hence not representing a beginning or an end of a jump sequence, (Figure 4.5). A function's minimum and maximum value at a jump, or sequence of consecutive

jumps, is used for segmentation. This was further evaluated in the second part of our work as it was expected to influence the accuracy of function recreation.

For all 2048 days there were 13 jumps located using a one-sided confidence level of $.015$, (Figure 4.7). This included the six jumps from the first 512 days. These 13 jumps were of magnitude ranging from 14 to 65 cubic feet per second. In addition, an intermediate jump of magnitude 59, that is, a jump located between two other jumps, was identified on March 10, 1948 ($x = .1489$). Intermediate jumps are not expected to be of high importance for function recreation, as it is located directly between the two values noting the beginning and end of the jump. The estimated location deviation of these 13 jumps ranged from $.0002$ to $.0009$, which was less than two sampling intervals off of the actual jump location, (sampling interval of $\frac{1}{2048} \sim .0005$). Except for the intermediate jump described above, most consecutive jumps seemed to be identified by the first and last values and hence the sum magnitude was correctly identified. Such an event occurred and was identified by the DWT coefficients on the dates 3/10/1948 to 3/12/1948, with magnitudes of 59, 65 and 27 cubic feet per second, respectively. A similar event occurred on 9/5 to 9/7/1950, with magnitudes of 18, 55 and 28 cubic feet per second, respectively.

This data set represented rather rapid fluctuations within segments compared to the standard functions where the segments were of much smoother nature. Real life data most likely included some noise that the standard functions did not have. The reason why two of the jumps in the first 512 days were not identified without using a more narrow confidence level, was most likely caused by a higher standard deviation for this subset of the data, compared to the whole 2048 days, with standard deviation of 3.5 vs 1.6, for the 512 days vs. the 2048 days, respectively.

The results from analyzing the USGS data set for jumps served as a good example of the method's sensitivity to the DWT coefficients' standard deviation and the number of samples used.

4.2.2 PAP Data Jump Results

The second set of real life data involved recordings of human pulmonary artery pressure (PAP) taken from PhysioNet [Welch et al., 1991, Goldberger et al., 2000]. Among the database's 250 recordings, two were selected for our analysis based upon visual inspection indicating potential jumps, (Figure 4.8 and Figure 4.9). Both recordings had a sample frequency of 360 samples per second and contain 3600 samples, with values ranging from 5.26 to 38.09 and 10.85 to 44.53 mmHg for case 012 and 022, respectively.

In order to be consistent with the analysis of standard functions and the USGS data, we used 512 and 2048 samples for our analysis. For case 012 the segment containing the last 512 samples (the last 1.422 seconds of the recording) was used. The only jump detected was at the start of the segment reflecting a PAP start value of 12 mmHg, which was correctly located at $x < .002$, on $[0, 1]$, using an one-sided confidence level of $.015$, ($q = .03$). For case 022 and the last 512 samples, two jumps were detected. First jump was at the start of the recording with a value of 26, second jump was between $x_{103} = .1977$ and $x_{104} = .2000$, which was the last of seven consecutive jumps at magnitude 2, 2, 3, 3, 3, 3 and 2, respectively, (Figure 4.11). This second jump was also identified using $n = 2048$.

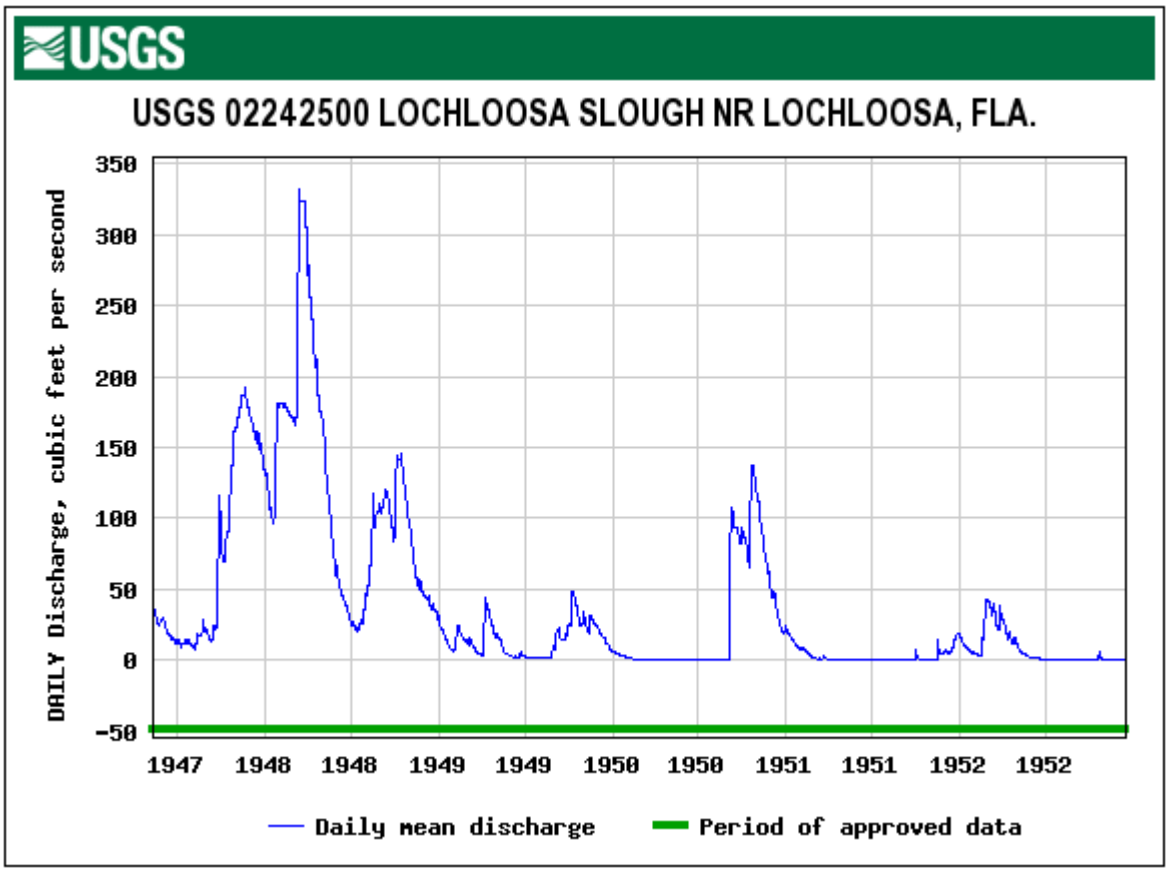

Figure 4.3: Mean daily water discharge at Lochloosa, FL, for the 2048 days from May 10, 1947 to December 16, 1952 (from USGS, U.S. Geological Survey, 2012, National Water Information System data available on the World Wide Web (USGS Water Data for the Nation), accessed July 16, 2016, at URL http://waterdata.usgs.gov/nwis/

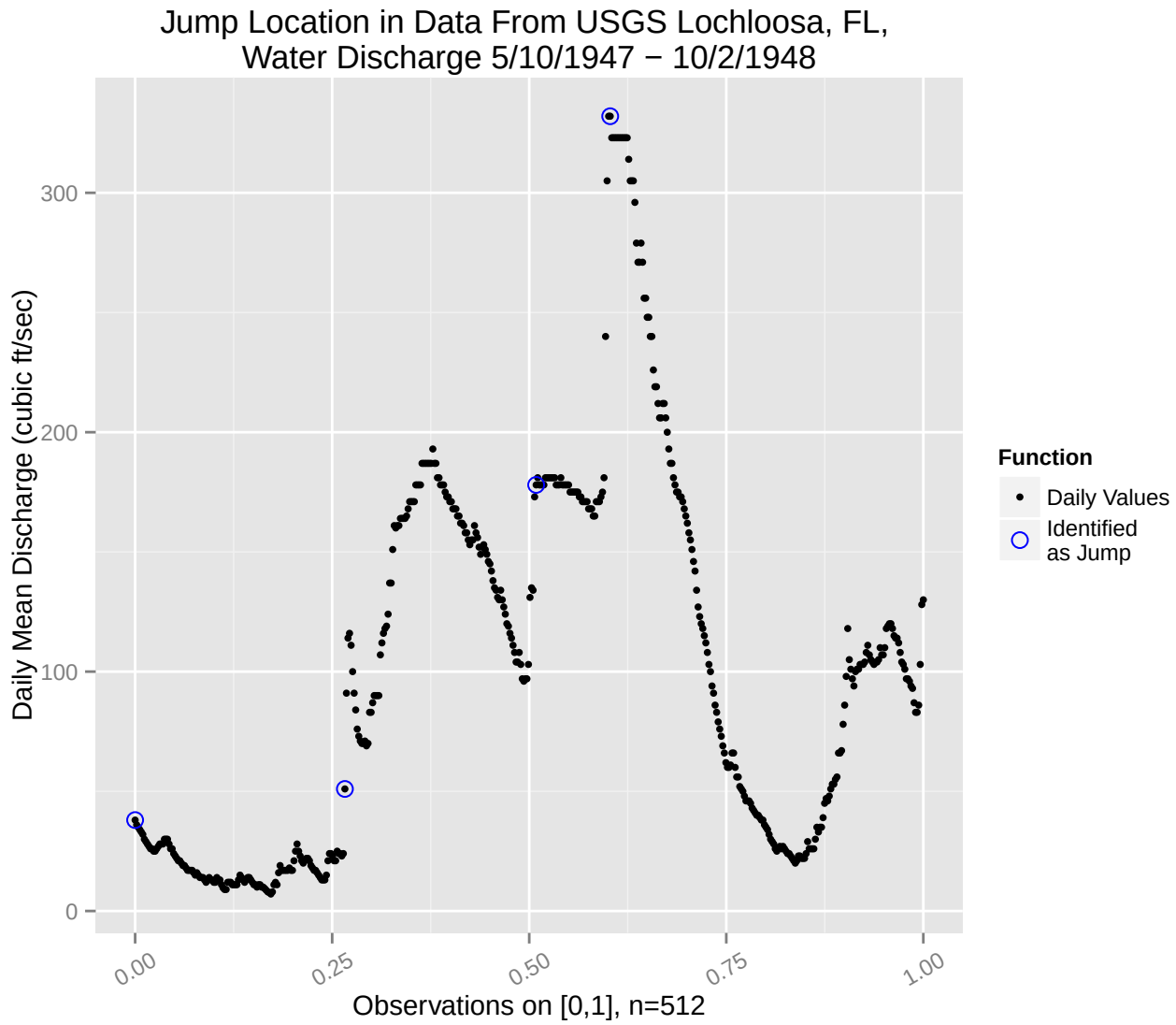

Figure 4.4: Location of the four jumps (in blue) identified in mean daily water discharge at Lochloosa, FL, for the 512 days from May 10, 1947 to Oct 2, 1948, normalized on [0, 1], using confidence level of $q/2 = .015$

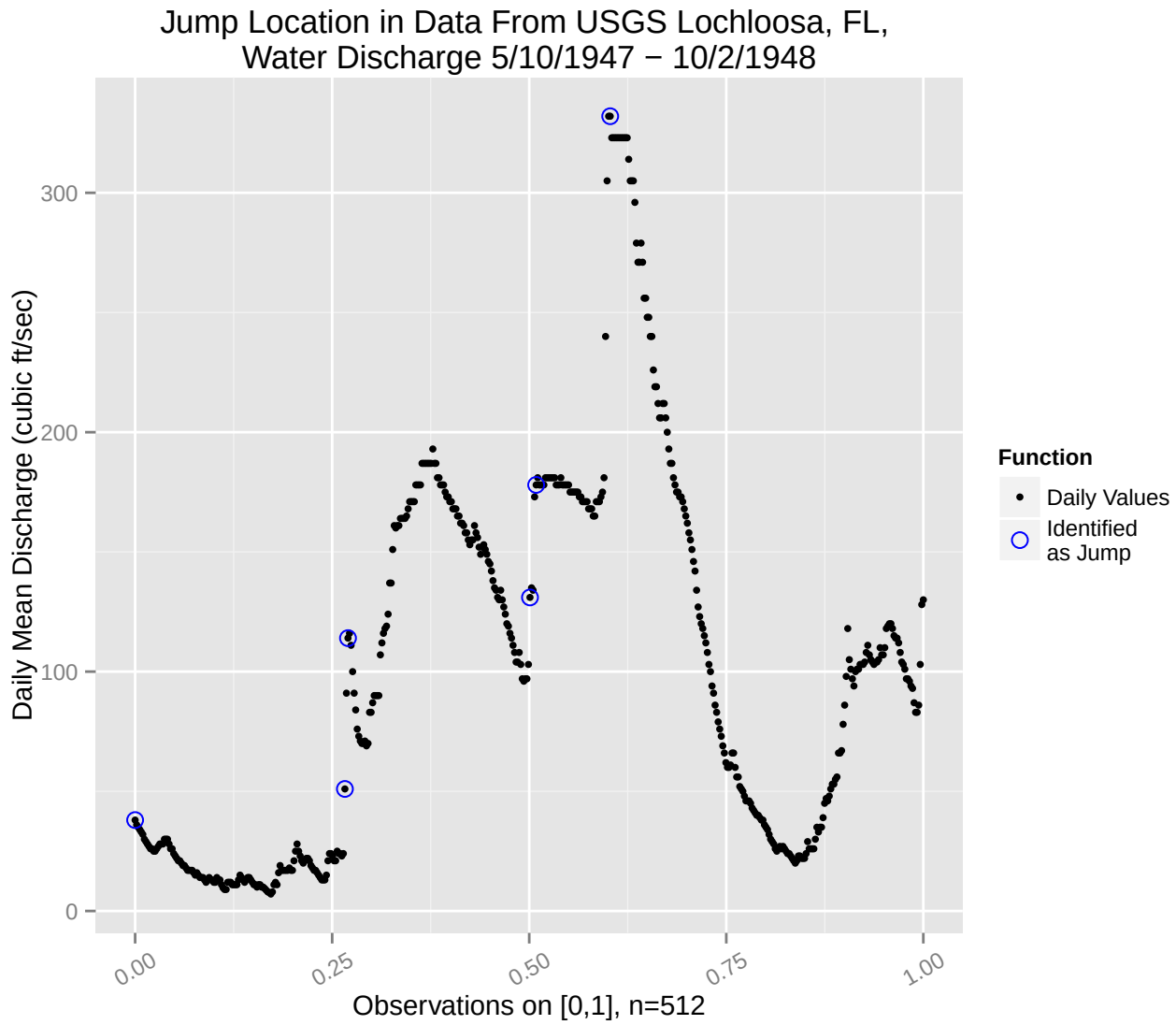

Figure 4.5: Six jumps (in blue) identified in mean daily water discharge at Lochloosa, FL, for the 512 days from May 10, 1947 to Oct 2, 1948, normalized on [0, 1], using confidence level of $q/2 = .1$

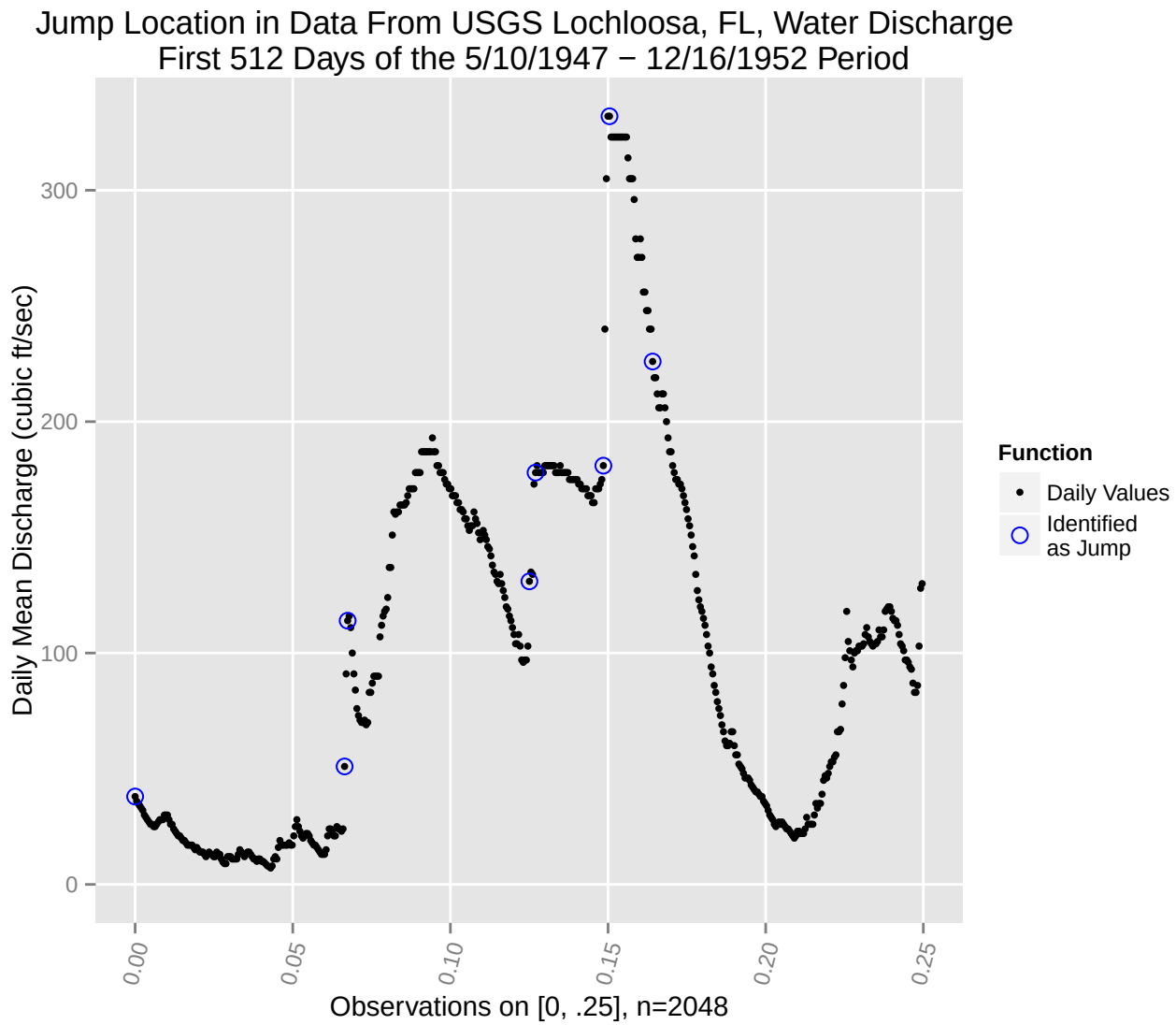

Figure 4.6: Location of the four jumps (in blue) identified in mean daily water discharge at Lochloosa, FL, for the 512 days from May 10, 1947 to Oct 2, 1948, normalized on [0, 1], using confidence level of .015

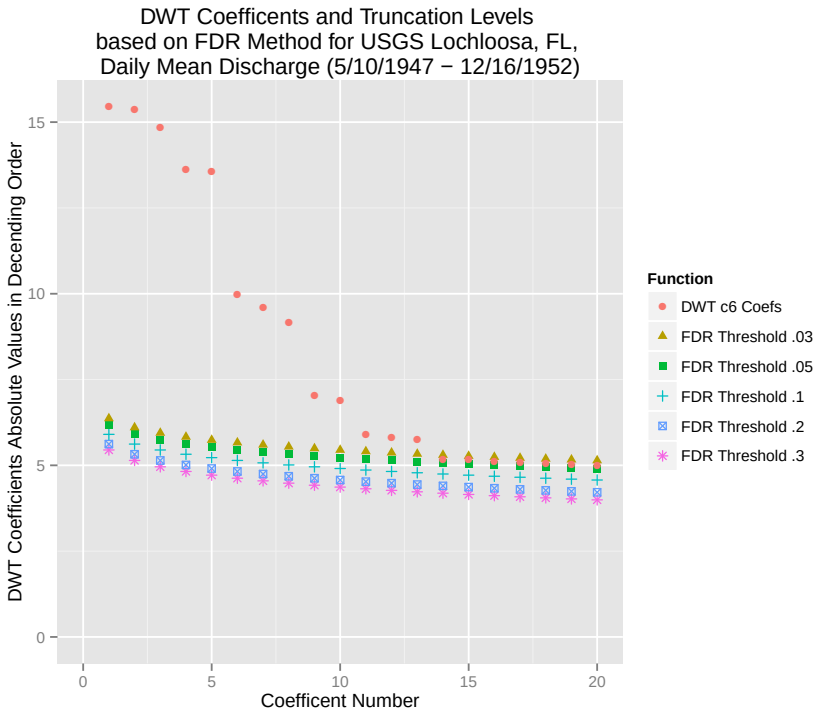

Figure 4.7: Thirteen jumps identified in mean daily water discharge at Lochloosa, FL, for the 2048 days from May 10, 1947 to Dec 16, 1952

For recording 012, the number of jumps identified in the last 1.422 seconds of recording increased from one to 10 when the sample size was increased from 512 to 2048. That is nine additional jumps over the same 512 samples of the set, (Figure 4.10). For recording 022, only one additional jump were found when the last 2048 samples were analyzed. In case 012, the standard deviation for the last 512 samples were twice as high as for the last 2048 samples (.1475 vs. .0789), whereas the difference in standard deviation for recording 022 was only .1401 vs. .1415. This result aligned with the results from the USGS analysis, pointing to the method's sensitivity to the DWT coefficients' standard deviation. This sensitivity appeared even more evident when the universal threshold method was used as it in general resulted in a lower threshold level than the FDR method.

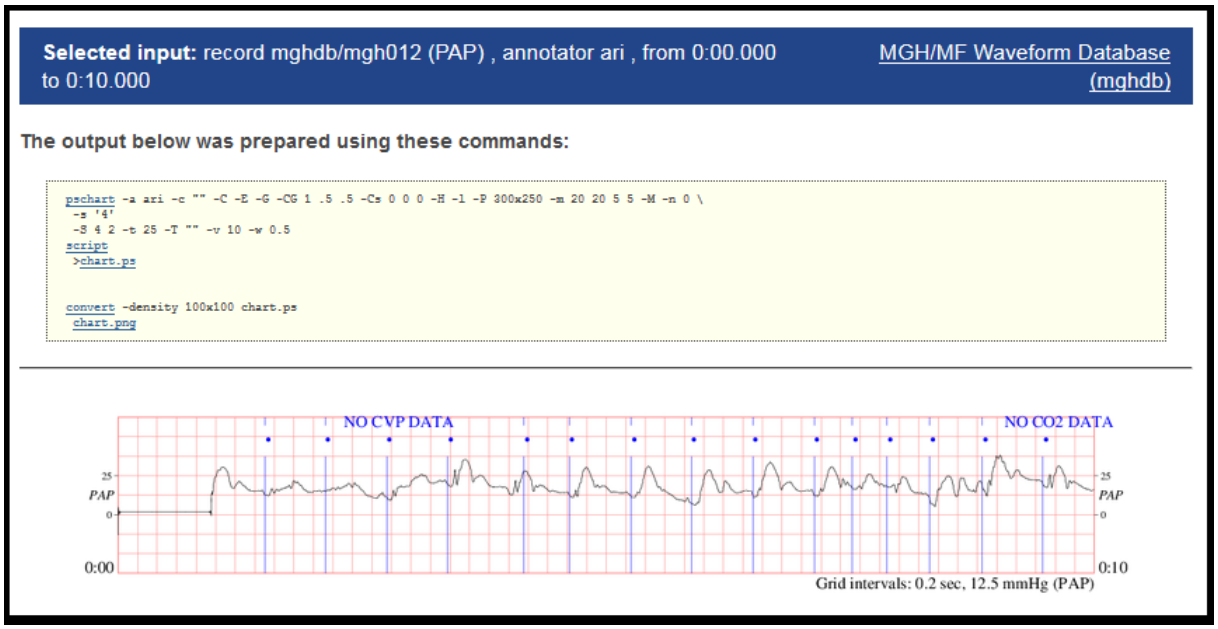

Figure 4.8: Recordings of Pulmonary Artery Pressure (mm Hg) case 012 (from Welch et al. [1991])

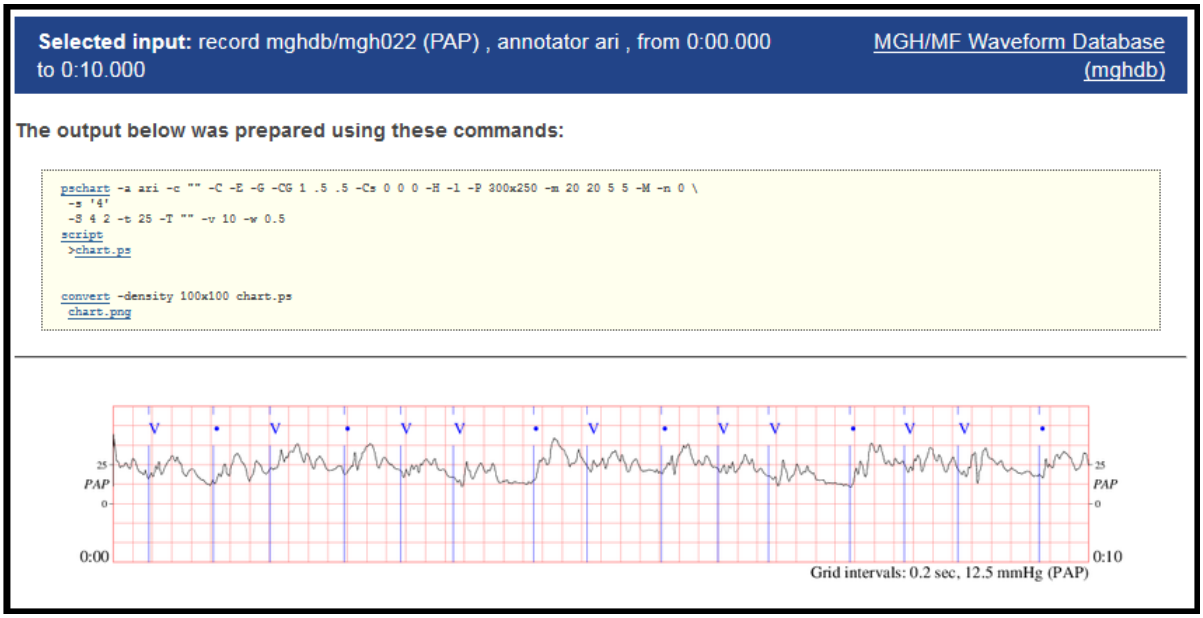

Figure 4.9: Recordings of Pulmonary Artery Pressure (mm Hg) case 022 (from Welch et al. [1991])

4.3 Jumps vs. Spikes

The differentiation of jumps vs. spikes in digital data seemed to be less clear than in continuous functions like the benchmark function we used. Even though the mathematical definition of jumps and spikes are clear in stating the difference of the two, the differentiation did not seem as clear when analyzing real life data. For both sets of real life data, some of the jumps identified were referred to as intermediate jumps. As these recordings are of digital nature vs. continuous, the limit evaluations involved to differentiate between jumps vs. spikes are not always practical. Hence, from a strict mathematical point of views, some of the jumps referred to above, may be spikes. From a pure theoretical point of view, the differentiation between the two may be of interest, but for our application it did not make any difference because jumps and spikes were used

for the same purpose, to segment the function or data set and then interpolate segment by segment. The recreation of the piece-wise regular function represent such an example and the results are presented in this chapter, section three.

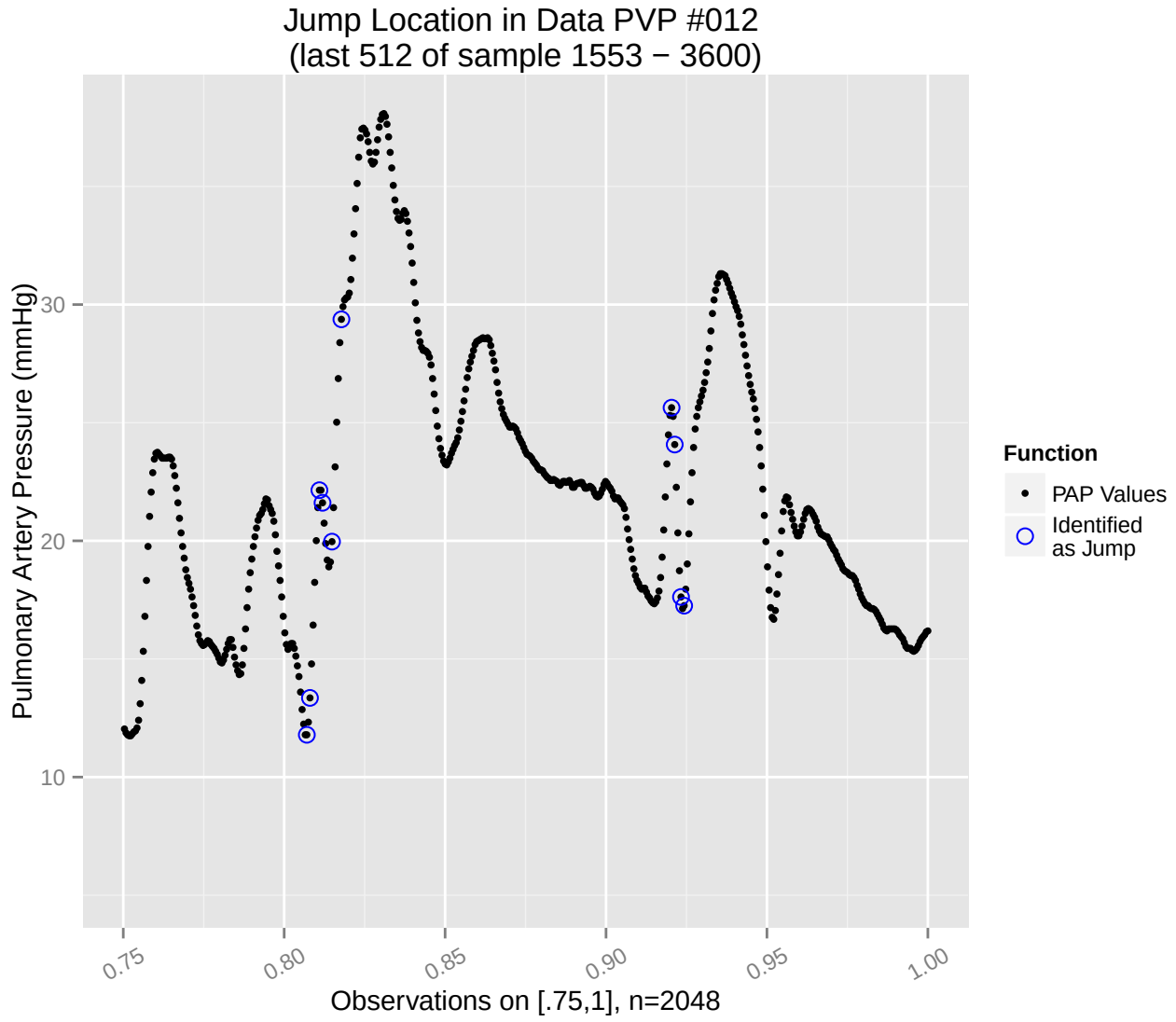

Figure 4.10: Case 012: Jumps and their locations (in blue) identified in the last 512 samples, normalized on [0, 1], confidence level $q/2 = .015$

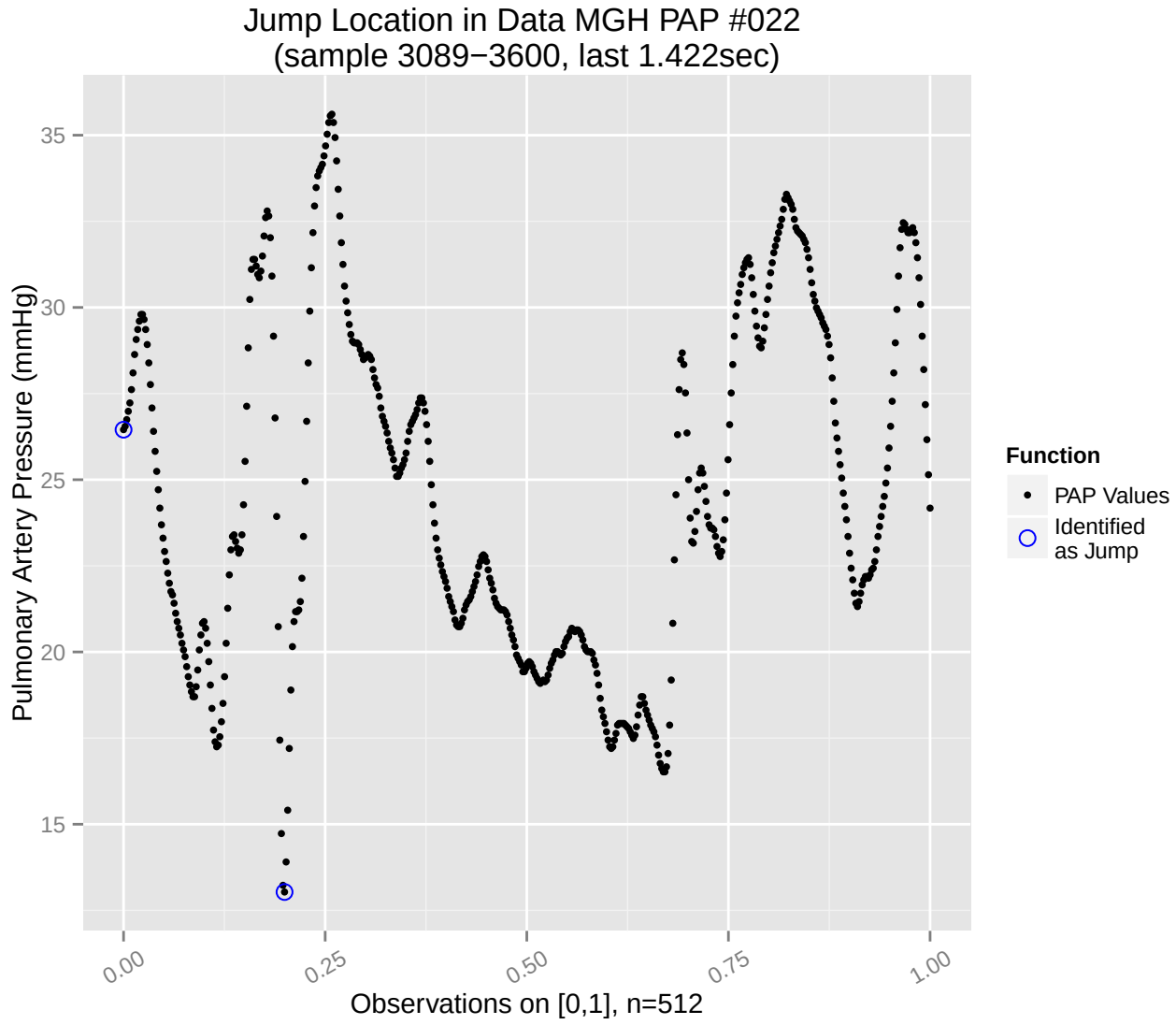

Figure 4.11: Case 022: Jumps and their locations (in blue) identified in the last 512 samples, normalized on [0, 1], confidence level $q/2 = .015$

The results from jump detection for both sets of real life data, USGS and Physiobank, served as input to the function recreation part of this thesis, where the accuracy of the jump de-

tection method influenced the accuracy of the recreation. A more comprehensive discussion is included in the this chapter, sixth section.

4.4 Recreation of the Standard Functions

The results from jump identification were used to segment and then recreate the functions segment by segment. The interpolation method we used was B-splines of degree three with penalties, ("smooth.spline" in the package "stats" in R). The B-splines are based on so-called equivalent bases, which makes these splines superior to splines from truncated power bases in terms of stability, [Abramovich et al., 2007]. The number of multiplications for the B-spline method is of $O(n)$, "big O of order n", (Ramsay et al. [1997]), where n is the sample size or number of knots used. The loss of accuracy was defined as $1/n$ of the L_2 norm for the original function vs. recreated function as stated in eq. 3.3. The results obtained by using the c6 wavelet were first used as it was the only wavelet to identify correct number of jumps in all three functions.

For all three standard functions, the segment-wise recreation was superior to non-segmented recreation in terms of:

- Improved computational efficiency in terms of reduced number of multiplications
- Improved computational speed by use of parallel execution of segments
- Less loss of accuracy when number of multiplications were reduced.

The only exception was for the Piece-wise Polynomial function that had the estimated location of jump one and jump seven one sample off from actual location, which resulted in higher error for the segment-wise recreation than the non-segmented recreation when number of knots and degrees of freedom (df) were equal to the sample size. The segment-wise recreation was superior to the non-segmented recreation as the number of knots and degrees of freedom were reduced to less than the full sample size.

As expected, using all n samples as knots gave the best accuracy, with errors in the order of 10^{-9}, 10^{-13} and 10^{-16} for the piece-wise polynomial, piece-wise regular and segment-wise cosine functions, respectively, for segment-wise recreation. For the context, these are bounded by the numeric precision limits of the tools we were using that is machine epsilon of order 10^{-16} for double precision. By reducing the number of knots to $\frac{2}{3}$ of the sample size, the error increased to 10^{-5}, 10^{-4}, and 10^{-9} respectively, for the three functions. In addition to reducing number of knots, the degree of freedom was also reduced. For the recreation method used here, a reduction in degrees of freedom reduces the smoothness between knots. We found that degrees of freedom had to be higher than number of knots in order to not cause increased error in addition to the error from knot reduction. This is illustrated by the results from recreating the segment-wise cosine function, where as the degrees of freedom was reduced to be less than the number of knots, the error increased, (Figure 4.12). This was the case for all three standard functions.

As a consequence, if the accuracy achieved by a reduction in number of knots by $r_1 * n$, additional computational efficiency, without further loss of accuracy, may be achieved by reducing degrees of freedom by a factor r_2 as long as $0 < r_1 < r_2 < 1$.

For all three standard functions, the increase in error was higher for non-segmented recreation than for segment-wise recreation when the number of knots were reduced. The error for knot reduction to $2n/3$ are listed in Table 4.2. The difference in error between segment-wise recreation vs. non-segmented was largest for the Segment-wise Cosine function. This was the only of the three standard functions where all jump locations were correctly estimated. The Piece-wise Regular function was the one with least benefit from segmentation as three of the four jump location estimates were one sample off from actual location and caused these jumps to appear as part of a segment in stead of between segments. Still with the misses in jump location, the benefit from segment-wise recreation lead to three times lower error than non-segmented recreation. From a graph of the function the non-segmented recreation seemed to display more profound fluctuations around the jumps, the so-called Gibbs phenomenon, than the segment-wise recreation, (Figure 4.13). In order to evaluate the potential impact on the recreation accuracy from inaccurate jump locations, the sym3 wavelet, which gave correct location of all four jumps, was used . This wavelet also identified the two first of the three spikes this function contains. The function was first segmented by the four jumps and then recreated. This resulted in loss of accuracy similar to when three of the four jumps were one sample off form actual location. Secondly, the function was segmented using both the four jumps and the two first spikes. This improved accuracy by reducing the error by almost a factor of about three or better, ref. Table 4.3.

The piece-wise regular function was most accurately recreated when the sym3 wavelet was used, which lead to seven segments that were initiated by the four jumps and two of the three spikes. It was further worth noticing that when segmenting without using the spikes, the

sym3 wavelet resulted in only slightly higher accuracy than the c6 wavelet. For the segment-wise cosine function, the two wavelets resulted in equal segmentation and hence equal accuracy when recreated. The sym3 wavelet failed to identify any of the jumps the piece-wise polynomial function, while the c6 wavelet identified all eight jumps.

Based on these results we concluded that the c6 wavelet was to be preferred for the remaining work. We further concluded that additional research is needed to understand the characteristics of the wavelets and identify criteria for optimal use.

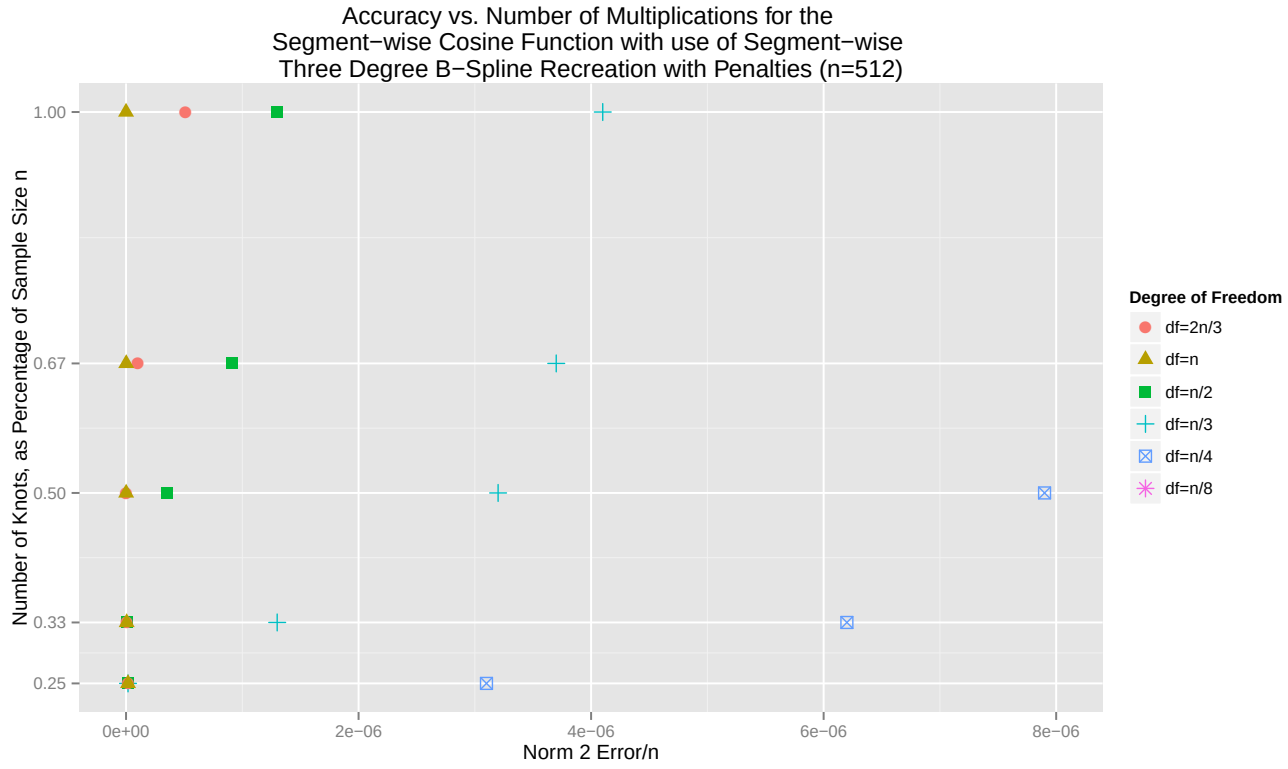

Figure 4.12: Multiplications and reduction of number of multiplications with effect on error in recreation

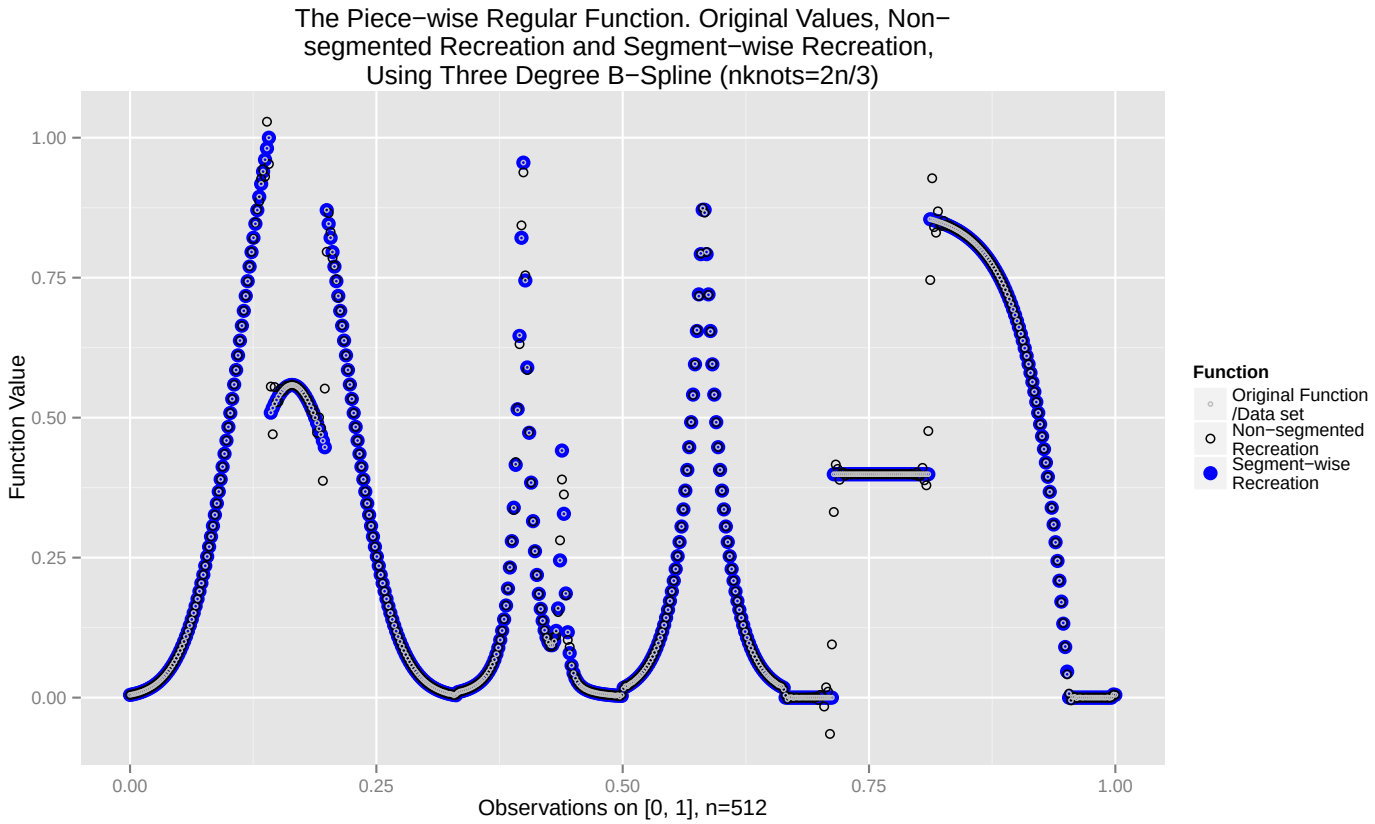

Figure 4.13: Recreation of the Piece-wise Regular function, displaying the Gibbs Phenomenon at the jumps

Table 4.2: Error for segment-wise vs. non-segmented recreation of the three standard functions. Knots=2n/3, df=n. Wavelet c6.

Function	Segment-wise	Non-segmented
Piece-wise Polynomial	1.9e-5	5.9e-4
Piece-wise Regular	1.7e-4	5.7e-4
Segment-wise Cosine	1.5e-9	5.6e-3

Table 4.3: Error for segment-wise vs. non-segmented recreation of the Piece-wise Regular function, segmented by jumps vs. both jumps and spikes.

No. Knots	Jump segmented, c6	Jump segmented, sym3	Jump and spike segmented, sym3
$2n/3$	1.7e-4	1.6e-4	2.9e-5
$n/2$	2.7e-4	1.9e-4	5.9e-5
$n/3$	5.3e-4	5.2e-4	1.5e-4

4.5 Recreation of a Shock Wave, A Direct Example from SPH Modeling

A function representing the density profile of a shock wave front in homogeneous media of similar nature as described as the Sedov test, [Cabezón et al., 2008], was analyzed using the c6 wavelet, (Figure 1.5). The function values represent density ($\frac{gram}{cm^2}$) as a function of time. Two jumps were detected, an initial jump at $x = 0$ and a second jump at $x = .6797$. The latter jump represents where the shock wave reaches its maximum density before stabilizing at a lower level.

The function was segmented and recreated according to the two jumps identified. The results showed increase in accuracy compared to non-segmented recreation, both when using all 512 data points as knots and when number of knots were reduced. The accuracy increased from the order of 10^{-12} to 10^{-16} for 512 knots and from 10^{-3} to 10^{-11} for $2n/3$ knots used. The difference in fit is displayed and the Gibbs phenomenon is visible in the vicinity of the second jump for the non-segmented recreation, (Figure 4.14). The segment-wise recreated function seemed to correspond well to the original function on both segments. This was an improvement compared to the SPH example, where the authors among others comment on the delay of the kernel function to conform to the analytic function's constant segment, (second segment). The ability of our chosen method to rather accurately recreate this function demonstrates the strength of this method to handle jumps in otherwise smooth functions.

4.6 Recreation of Real Life Data

The real life data sets were analyzed using the same method as for the standard functions. Each set of data was segmented according to the jumps identified and the original data sequence recreated by the smoothing routine. The results were measured in terms of accuracy using the same error measure.

The real life data sets had common factors that most likely influenced the results of our analysis. In particular, the data did most likely contain noise and also the segments between jumps were less smooth compared to the standard functions and the shock wave data.

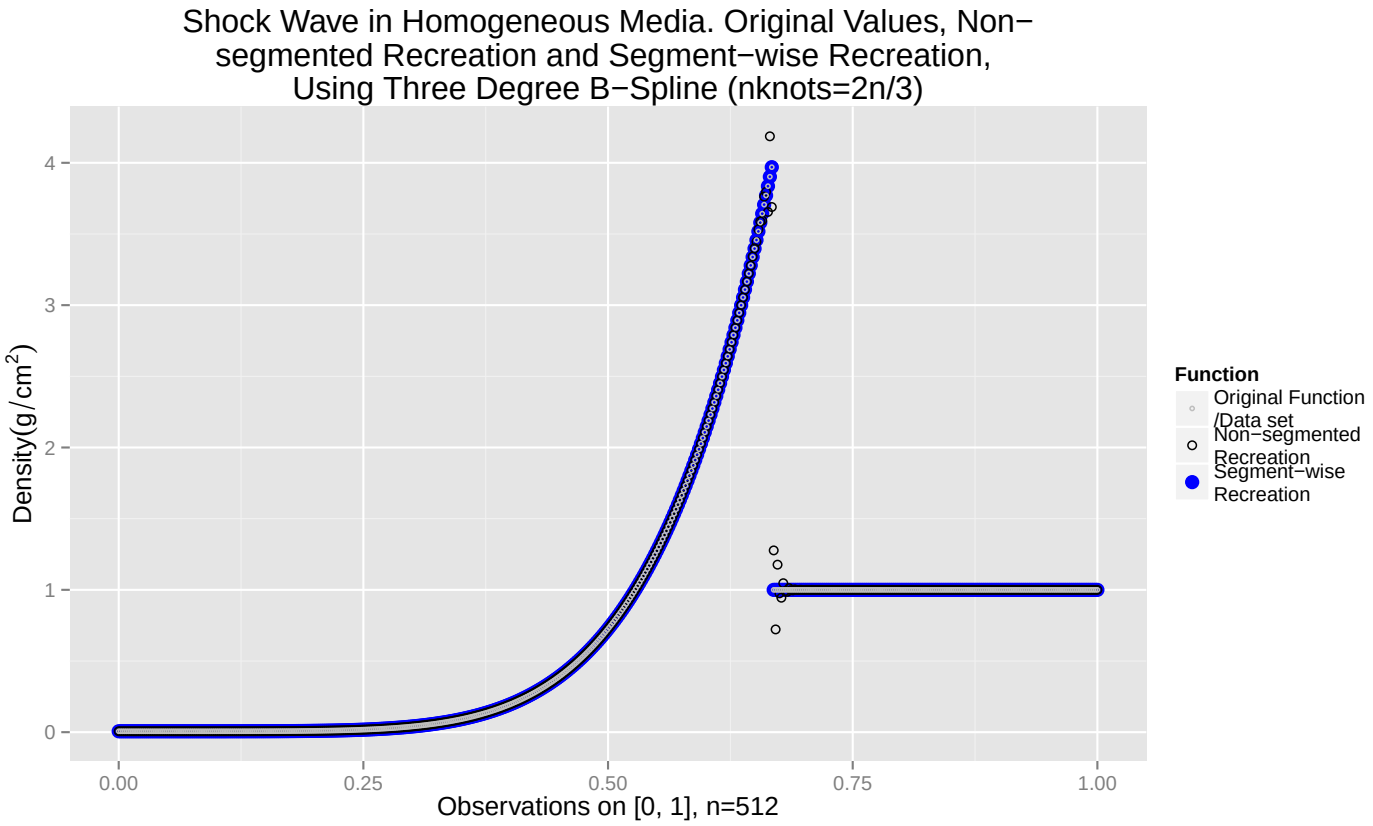

Figure 4.14: Shock wave homogeneous media, comparison of segment-wise recreation vs. non-segmented

The results supported our conclusions from segment-wise recreation of the standard functions and the shock wave by allowing for parallel processing of segments without loss in accuracy. The accuracy was in all cases consistently equal to or less with parallel processing vs. non-segmented recreation.

The accuracy of segment-wise recreated data sets were far less than for the standard functions and the shock wave, in particular when number of knots were reduced. This was most likely due to the two above mentioned factors, noisy data and less smooth segments.

The first 512 days of recordings in the USGS Water Resources Lochloosa, FL, data set was segmented into four based on the four jumps detected by the c6 wavelet. This data set demonstrated very minor differences in accuracy weather it was segmented or not before recreation. With all 512 data points used as knots, the segmented recreation resulted in error of $3.1 * 10^{-11}$, vs. $5.1 * 10^{-11}$. With two-thirds of the data points as knots, the error increased to $.0410$ vs. $.0500$, respectively, for the segment-wise and non-segmented recreation. By reducing number of knots to half of the number of data points, the error increased slightly to $.0543$ and $.0610$, respectively. Degree of freedom was kept at 512 in each of these cases. The minor increase in error when reducing the number of knots from $2n/3$ to $n/2$ may be accepted as trade-of when computational efficiency and real time simulations are of priority. Data for the first 512 days, as well as some additional segments of length 512 were analyzed both by c6 and sym3, with the results supporting the above findings both in terms of accuracy and computational benefits. Lastly, the entire data set of 2048 days were analyzed. The results also supported the above findings, showing minor loss of accuracy by non-segmented recreation vs. segment-wise recreation. Common for both the data representing 512 days and data from 2048 days were the identification of some intermediate jumps as reported under jump identification (Figures 4.4, 4.6). The smoothing method used in this work does not support segments of less than four data points. This was overcome by identifying the jump with the largest magnitude and segment the data sequence accordingly.

The first case, case 012, of human pulmonary artery pressure recordings did not contain any jump except for at the very start of the recordings, and hence, was excluded from segment-wise recreation. For the second case, case 022, the data sequence was segmented into two according to

the jump located between sample 103 and 104. The results in terms of accuracy did not show any difference between segment-wise vs. non-segmented recreation. Both when all n points were used as knots and where number of knots were reduced, the accuracy degraded by the same amount independent of segmentation, from 10^{-13} to 10^{-4} for n and $n/2$, respectively. The results of processing these two segments in parallel reduced the length of the processing sequence by about 20 percent, and hence increased the capacity for real time results accordingly.

4.7 Spikes

The initial intention was to include analysis of spikes in this research to evaluate if and how spikes may be identified and used to improve function recreation. The results from analyzing the piece-wise regular function indicated that spikes appear to also be detectable during jump identification as they seemed to cause increase in the absolute values of DWT coefficients. The absolute values of these coefficients were found to be less than for DWT coefficients associated with jumps, but higher than where no jumps or spikes were present. When the c6 wavelet was used, these coefficients were not identified as significant, that is, they were inside the confidence interval at $\frac{q}{2} > .15$, (Figure 4.15 coefficients five and six). By using the sym3 wavelet, the first two spike of the function were identified as significant, as represented by coefficients five and six in figure 4.16. As noted in the results from recreation of the standard functions, the piece-wise regular function was more accurately recreated when the spikes were used for segmenting vs. not, (Table 4.3). The third spike was not detected by either wavelet. Further evaluation of the function at this

location showed a finite value of the derivative of the function as this local maximum consisted of two function values of equal value. Secondly, this third spike had less steep inclination and declination than the other two spikes.

The results from analyzing the piece-wise regular function and the ECG recordings that we in our preliminary work modified to contain a jump, consistently identified jumps with DWT coefficients of larger absolute value than the presence of spikes. Both these factors, combined with wavelet type and confidence level selected, were expected to be of relevance for detecting spikes by the DWT method, something that will be of interest for the intended continuation of this research.

4.8 Accuracy, Stability and Computational Efficiency Results

The results of our work document improvements in modeling and simulation of less smooth flow of matters in terms of accuracy, stability and computational efficiency. The results further support the fact that our chosen method also performs better in the case of smooth segments than for less smooth segments, including segments with noise. This was clearly demonstrated as accuracy improvements were much higher for the shock wave case and the four standard functions than for the real life data sets with unknown signal to noise ratio.

The criteria we used to segment the functions and data sets resulted in segments with superior continuity properties compared to the non-segmented functions and data sets. As such continuity properties are required conditions to ensure that a numerical problem has a unique solution, our method may also be used to meet the uniqueness criteria when otherwise not met. Our simulations

processes performed according to the criteria for a well-posed numerical problem. Further, the B-splines' stability properties ensured simulations that resulted in convergence by a finite number of steps for all of our simulations.

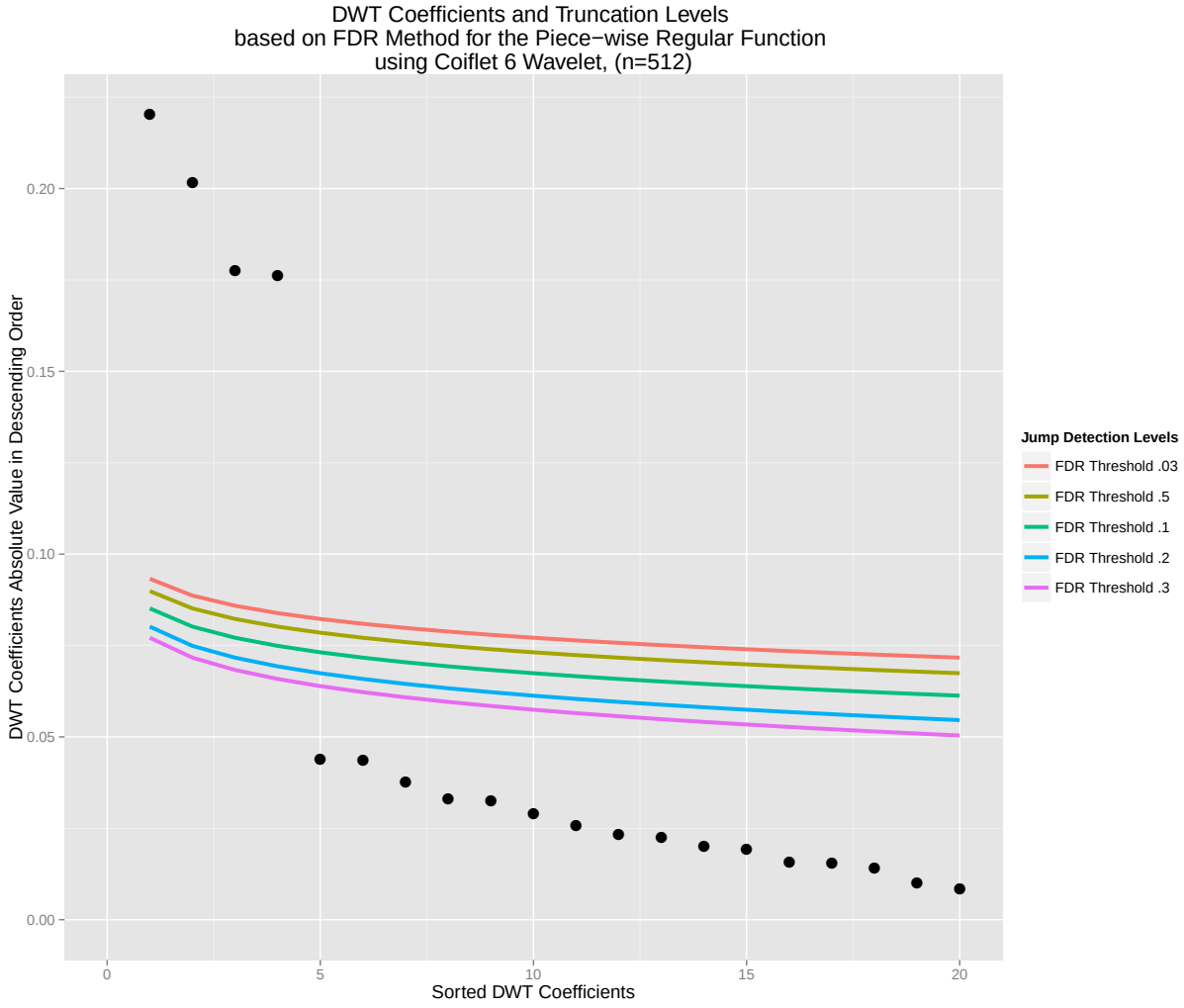

Figure 4.15: Four significant c6 DWT coefficients identifying four jumps of the piece-wise regular function

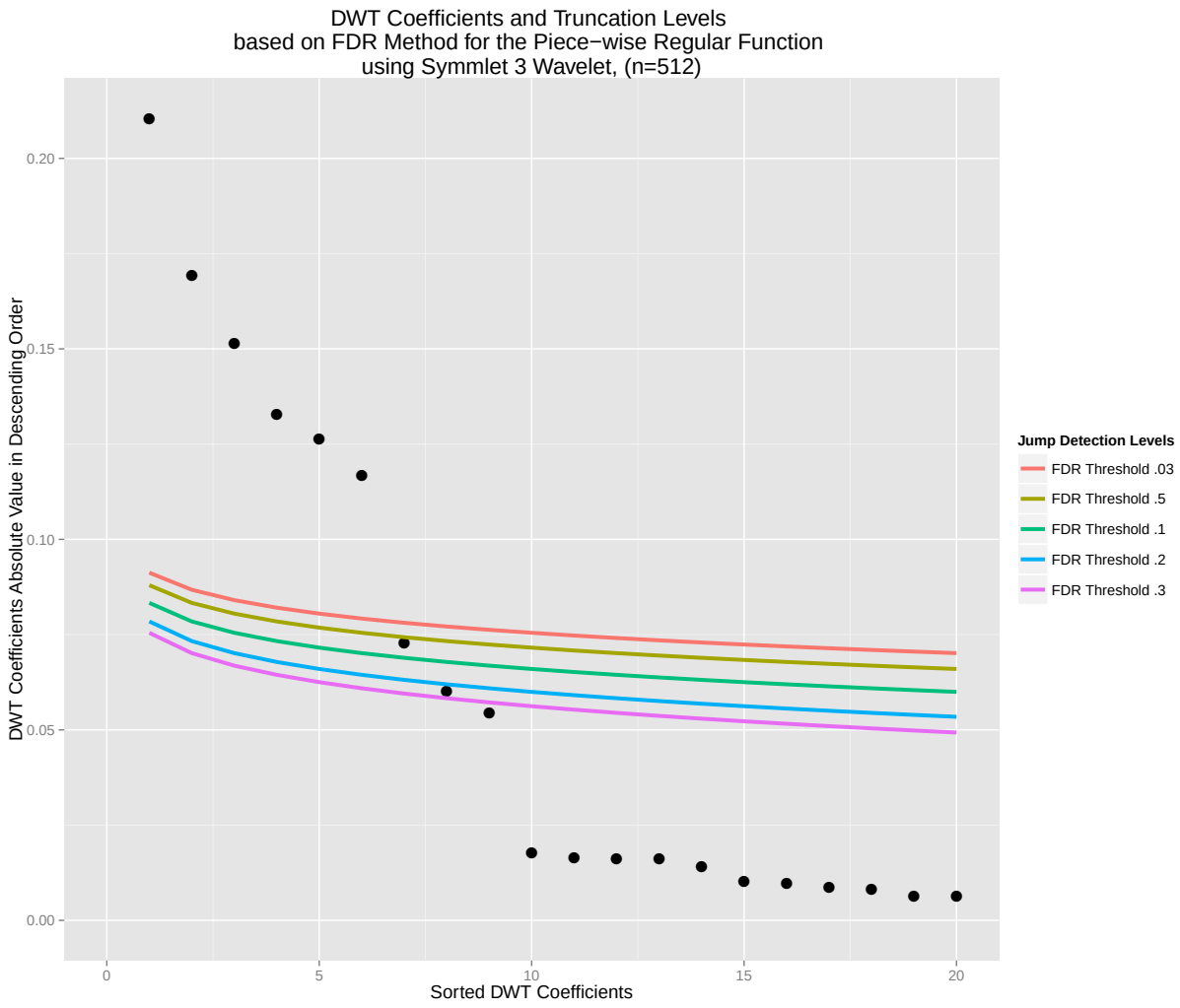

Figure 4.16: Six significant sym3 DWT coefficients identifying four jumps and two spikes of the piece-wise regular function

The computational efficiency was increased. Both the absolute number of computational operations and the actual execution time were reduced. The number of multiplications was reduced by about 30 and 50 percent by reducing number of knots and degrees of freedom for interpolation. The increase in error by such reduction was less for the segment-wise smooth functions than the

less smooth ones. In both cases the method represents a tool to evaluate the trade off between computational efficiency vs. accuracy. The relevance of these two factors are expected to differ from application to application and having this tool provides the opportunity to optimize accordingly.

Actual processing time was reduced. By segmenting the functions and data sets, the interpolation of the segments could be executed in parallel and actual processing time was bounded by the time it took for the longest segment to be recreated. For the shock wave example, this resulted in a reduction from 512 sequential operations to two parallel operations of max 342 sequential steps, a reduction of about 30%. As this research only looked at one-dimensional functions of rather low number of samples, the actual CPU time was negligible both for non-segmented and segment-wise interpolation. For multi-dimensional functions the number of computational operations and use of CPU time increase significantly. In such cases the parallel structure we used has the potential of significantly reducing processing time and increase real-time capacity.

The need for real time results has been one of the key reasons cited by authors to use SPH. Our demonstrated ability to both use parallel processing and reduce number of multiplications creates an opportunity for real-time simulations results accordingly.

Table 4.4: Computational Efficiency and Accuracy. Computational Efficiency ($\%$ reduction in number of multiplications). Accuracy Loss is Due to Knot Reduction. No Accuracy Loss Caused by Parallel Processing.

Functions (Sample Size $n = 512$)	Parallel Processing (Max. Segment Length)	No. Knots $2n/3$	Accuracy Loss ($2n/3$)	No. Knots $n/2$	Accuracy Loss ($n/2$)
Segment-wise Cosine	70% (154)	81%	$2.9 * 10^{-16}$ – $1.5 * 10^{-9}$	85%	$2.9 * 10^{-16}$ – $8.8 * 10^{-10}$
Piece-wise Regular	73% (141)	82%	$2.0 * 10^{-5}$ – $2.9 * 10^{-5}$	86%	$2.0 * 10^{-5}$ – $5.9 * 10^{-5}$
Piece-wise Polynomial	80% (102)	87%	$8.9 * 10^{-9}$ – $1.9 * 10^{-5}$	90%	$8.9 * 10^{-9}$ – $1.1 * 10^{-4}$
Shock wave	32% (347)	55%	$1.0 * 10^{-16}$ – $4.3 * 10^{-11}$	66%	$1.0 * 10^{-16}$ – $2.3 * 10^{-11}$
USGS Data	60% (204)	73%	$3.1 * 10^{-11}$ – .041	80%	$3.1 * 10^{-11}$ – .054
PAP Data 012	0% (512)	30%	$4.0 * 10^{-13}$ – $3.6 * 10^{-4}$	50%	$4.0 * 10^{-13}$ – $6.8 * 10^{-4}$
PAP Data 022	20% (409)	47%	$4.2 * 10^{-13}$ – $4.0 * 10^{-4}$	60%	$4.2 * 10^{-13}$ – $6.0 * 10^{-4}$

CHAPTER 5
CONCLUSIONS

We have demonstrated with the results of this research that functions and data sets containing jumps and spikes are interpolated more accurately and computationally efficient with a two-step method using wavelet analysis and spline interpolation [Abramovich et al., 2006] compared to SPH kernel interpolations. This means that flow of less smooth nature, in particular flow containing jumps and spikes, may be simulated with higher accuracy and improved real time capability, without compromised stability, a need that has been stressed numerous times by researchers in this field, as shown in many of the references included in our work.

In addition to the documented improvements, we have also introduced wavelet theory and information field theory to the SPH area of science for the first time, as far as we have been able to verify. By continuing to build and strengthen connections between the different fields of science, interdisciplinary problem solving will continue to find solutions to the challenging and interesting tasks presented by the future.

5.1 Contributions

SPH has proven to be a valuable method for flow simulations. Our results has demonstrated improved accuracy, stability and efficiency under more challenging conditions such as flow with jumps and spikes. This will benefit applications in a wide area of science. In medicine, improvements will lead to more realistic and real time models to increase understanding of human

physiology as well as pathological conditions and how to deal with such conditions. The many research references from medicine included in this work is a strong indication of this need. Examples are virtual surgical training, real time diagnostics and treatment of pathological conditions. Such improvements will represent significant cost savings in addition to health benefits due to improved understanding, diagnostics and treatments.

The results of this work is expected to benefit additional disciplines in science. The method's ability to demonstrate direct improvements when compared SPH interpolation is a strong indication of this. The analysis of three sets of real life data with unknown signal to noise ratio resulted in improved computational efficiency without loss of accuracy. These results further strengthen the method's ability to improve results in SPH modeling and simulations.

5.2 Future Research

It is our belief that the results from this research will serve as a solid foundation for future work in this field to improve modeling and simulations of less smooth flow and enhance real time simulations. It is also our belief and intention that the results of this work will serve as foundation for further development of SPH as a method and expand its areas of applications.

Particular areas for future research:

- Selection of wavelet type. Both coiflet 6 and symmlet 3 did prove relevant for jump and spike detection. The coiflet 6 was superior to symmlet 3 for detection of jumps. The symmlet 3 was

superior to coiflet 6 for spikes. Further research is recommended to identify their strengths and weaknesses and under what conditions one might be superior to the other.

- Spike detection. One of the standard functions, the piece-wise regular, represents both jumps and spikes. Our research results did indicate that some spikes are detectable by DWT. Further research is needed to determine the factors of importance for the detection. We would expect such factors to include wavelet characteristics, spike characteristics, signal-to-noise ratio, among others.

- Ensuring the three fundamental laws of physics, conservation of mass, momentum and energy, are met if using B-splines for interpolation in SPH. The B-splines used here do fulfill several of the seven criteria listed for SPH interpolation functions, such as compact support, positivity and smoothness, but additional conditions may be necessary in order to ensure that the other four criteria are met, unity, monotone decreasing, delta function property, smoothness and that the interpolation function is an even function.

- Further improve SPH results by use of Bayesian field theory. Bayesian methods in field theory, when used successfully, will lead to increased probability of a true representation by using an already known or assumed prior probability distribution about the underlying continuous space. We would expect that for several of the problems modeled by SPH , such apriori probability distributions may be achieved, completely or in part, as these problems are rather known problems from medicine, physics, and other fields of science.

The contributions from our research are important steps in order to improve interpolation in SPH in cases of less smooth flow. We would be honored to see this work carried further by members of the science research community. Our intention is for the author of this thesis to take a direct and active role in this work.

APPENDIX: COPYRIGHT PERMISSIONS

License no: 3971430776797 Sankhya: The Indian Journal of Statistics

License no: 3932591362704 Journal of Computational Physics Title: A non-parametric family of interpolating kernels for smoothed particle hydrodynamics studies

License no: 3906160112811 Coastal Engineering Title: Corrected Incompressible SPH method for accurate water-surface tracking in breaking waves

E-mail confirmation from Gabriel Peyre (gabriel.peyre@ens.fr) 10/19/2016 for permission to use figures from www.numericaltours.com for non-commercial use.

E-mail confirmation from Bill Kania (bill.kania@easyauskultation.com) 10/17/2016 for permission to use image of normal ECG from http://www.practicalclinicalskills.com/ecg-interpretation for non-commercial use.

LIST OF REFERENCES

Felix Abramovich, Joav Benjamini, David L. Donoho, and Ian M. Johnstone. Adapting to unknown sparcity by controlling the false discovery rate. *The Annals of Statistics*, 34:584–653, 2006.

Felix Abramovich, Anestis Antoniadis, and Marianna Pensky. Estimation of piecewise-smooth functions by amalgamated bridge regression splines. *Sankhya: The Indian Journal of statistics*, 69:1–27, 2007.

Ehsan Basafa, Ryan J. Murphy, Michael D. Kutzer, Yoshito Otake, and Mehran Armand. A particle model for prediction of cement infiltration of cancellous bone in osteoporotic bone augmentation. *PLoS ONE*, 8(6), 2013.

Francesca Boso, Alberto Bellin, and Michael Dumbser. Numerical simulations of solute transport in highly heterogeneous formations: A comparison of alternative numerical schemes. *Advances in water resources*, 52:178–189, 2013.

Ruben M. Cabezón, Domingo Garcia-Senz, and Antonio Relano. A one-parameter family of interpolating kernels for smoothed particle hydrodynamics studies. *Journal of Computational Physics*, 227:8523–8540, 2008.

Yim Pan Chui and Pheng Ann Heng. A meshless rheological model for blood vessel interaction in endovascular simulation. *Progress in Biophysics and Molecular Biology*, 103:252–261, 2010.

Paul W. Cleary, Raymond C.Z. Cohen, Simon M. Harrison, Matthew D. Sinnott, Mahesh Prakash, and Stuart Mead. Numerical modelling of water waves with the sph method. *International Journal for Computer-Aided Engineering and Software*, 30:157–196, 2013.

R.A. Dalrymple and B.D. Rogers. Numerical modelling of water waves with the sph method. *Costal Engineering*, 53:141–147, 2006.

Torstein Ensslin. Information field theory. *American Institute of Physics Conference Series*, 1553: 184–191, 1977.

Torstein Ensslin. Astrophysical data analysis with information field theory. *Astro-ph.IM*, May 2014. arXiv: 1405.7701v1.

Ali Fakhimi and Mark Lanari. Dem-sph simulation of rock blasting. *Computers and Geotechniques*, 55:158–164, 2014.

Michael Fowler. Frames of reference and newton's laws. March 2008. http://galileo.phys.virginia.edu/classes/252/lecture1.html.

Andrew Gelman, John B. Carlin, Hal S. Stern, David B. Dunson, Aki Vehtari, and Donald B. Rubin. *Bayesian Data Analysis*. CRC Press, 2014.

R.A. Gingold and J.J. Monaghan. Smoothed particle hydrodynamics: theory and application to non-spherical stars. *Mon. Not. R. astr. Soc.*, 181:375–389, 1977.

AL Goldberger, LAN Amaral, L Glass, JM Hausdorff, PCh Ivanov, RG Mark, JE Mietus, GB Moody, C-K Peng, and HE Stanley. Physiobank, physiotoolkit, and physionet: Components

of a new research resource for complex physiologic signals. *Circulation 101(23):e215-e220*, 101:215–220, 2000.

Andrew Kenneth Ho, Ling Tsou, Sheldon Green, and Sidney Fels. A 3d swallowing simulation using smoothed particle hydrodynamics. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging and Visualization*, 2:237–244, 2014.

S. Majid Hosseini and James J. Feng. A particle based model for the transportation of erythrocytes in capillaries. *Chemical Engineering Science*, 64:4488–4497, 2009.

Yu Huang and Zili Dai. Large deformations and failure simulations for geo disasters using smoothed particle hydrodynamics method. *Engineering Geology*, 168:86–97, 2014.

A. Khayyer, H. Gotoh, and S.D. Shao. Corrected incompressible sph method for accurate water surface tracking in breaking waves. *Costal Engineering*, 55:236–250, 2007.

D.R. Kincaid and E.W. Cheney. *Numerical analysis: mathematics of science computing*. American Mathematical Society, 2002.

Jorg C. Lemm. *Bayesian Field Theory*. John Hopkins University Press, 2003.

G.R. Liu and M.B. Liu. *Smoothed Particle Hydrodynamics*. World Scientific Publishing Co. Pte. Ltd., 2003.

Zudi Lu and Dag Tjöstheim. Nonparametric estimation of probability density functions for irregularly observed spatial data. *Journal of the American Statistical Association*, 109:1546–1564, December 2014.

Leon B Lucy. A numerical approach to the testing of the fission theory. *The Astronomical Journal*, 82:1013–1024, 1977.

Stephane Mallat. *A wavelet tour of signal processing*. Academic Press, 2009.

J.J. Monaghan. Smoothed particle hydrodynamics. *Ann. Rev. Astron. Astrophys*, 30:543–574, 1992.

J.J. Monaghan. Smoothed particle hydrodynamics. *Report on Progress in Physics*, 68:1703–1759, 2005.

Matthias Müller, Simon Schirm, and Matthias Teschner. Interactive blood simulation for virtual surgery based on smoothed particle hydrodynamics. *Technology and Health Care*, 12:25–31, 2004.

Marianna Pensky. Estimation of a smooth density function using meyer type wavelets. *Statistics & Decisions*, 17:111–123, 1999.

Donald B. Percival and Andrew T. Walden. *Wavelet Methods for Time Series Analysis*. Cambridge University Press, 2008.

Mahesh Prakash, Kai Rothauge, and Paul W. Cleary. Modelling the impact of dam failure scenarios on flood inudation using sph. *Applied Mathematical Modelling*, 38:5515–5534, 2014.

Soon Hyoung Pyo, Jeongjin Lee, Seongjin Park, Kyoung Won Kim, Yeong-Gil Shin, and Bohyung Kim. Physically based nonrigid registration using smoothed particle hydrodynamics: Applica-

tion to hepatic metastasis volume preserving registration. *IEEE Transactions on Biomedical Engineering*, 60:2530–2540, 2013.

Ashcan Rafiee, Sharen Cummins, Murray Rudman, and Krish Thiagarajan. Comparative study on the accuracy and stability of sph schemes in simulating energetic free-surface flows. *European Journal of Mechanics, B/Fluids*, 36:1–16, 2012.

J.O. Ramsay, N. Heckman, and B.W. Silverman. Spline smoothing with model-based penalties. *Behavior Research Methods, Instruments and Computers*, 29:99–106, 1997.

Walter Rudin. *Principles of Mathematical Analysis*. McGraw-Hill, 1976.

S. Shahriari, L. Kadem, B.D. Rogers, and I. Hassan. Smoothed particle hydrodynamics method applied to pulsatile flow inside a rigid two dimensional model of left heart cavity. *International Journal for Numerical Methods in Biomedical Engineering*, 28:1121–1143, 2012.

M.D. Sinnott, P.W. Cleary, J.W. Arkwright, and P.G. Dinning. Investigating the relationship between peristaltic contraction and fluid transportation in human colon using smoothed particle hydrodynamics. *Computers in Biology and Medicine*, 42:492–503, 2012.

Kenji Takamatsu and Takashi Kanai. A fast practical method for animating particle-based viscoelastic fluids. *The International Journal of Virtual Reality*, 10(1):25–31, 2011.

Larry Wasserman. *All of nonparametric statistics*. Springer, 2006.

Eric W. Weisstein. Wavelet, from mathworld, a wolfram web resource. March 2015. http://mathworld.wolfram.com/Wavelet.html.

JP Welch, PJ Ford, RS Teplick, and RM Rubsamen. The massachusetts general hospital-marquette foundation hemodynamic and electrocardiographic database–comprehensive collection of critical waveforms. *J Clinical Monitoring*, 7:69–97, 1991.

Peter Widas. Introduction to finite element analysis. April 1997. http://www.sv.vt.edu/classes/MSE2094_NoteBook/97ClassProj/num/widas/history.html.

Tianchen Xu, Wen Wu, and Enhua Wu. Real time generation of smoothed particle hydrodynamics based special effectsin character animation. *Computer Animation and Virtual Worlds*, 25:185–198, 2014.

G.M. Zhang and R.C. Batra. Modified smoothed particle hydrodynamics method and its applications to transient problems. *Computational Mechanics*, 34:137–146, 2004.