Reconstruction from Averages

Dissertation
zur Erlangung des akademischen Grades eines Doktors
an der
Fakultät für Naturwissenschaften und Mathematik
der
Universität Wien

Eingereicht von
Mag. Harald Schwab

Ausgeführt am
Institut für Mathematik der Universität Wien

September, 2003
Meiner Mutter
und
meinem Vater.
Contents

1. German Summary 7

2. Acknowledgements 11

3. Introduction 13
   3.1. Motivation ........................................ 13
   3.2. Efficient Algorithms ................................ 15
       3.2.1. Sampling versus Averaging? ...................... 15
       3.2.2. Exploring the Structure ......................... 15

4. Notations and Definitions 17

5. Iterative Reconstruction Algorithms for Bandlimited Functions 23

6. Reconstruction from Averages for Bandlimited Functions 31
   6.1. An Overview ......................................... 31
   6.2. Fast Algorithm using CG-Method .................... 37
       6.2.1. Interpretation as Frame ......................... 38
       6.2.2. Fast Efficient Algorithm using CG-Method ...... 40
   6.3. Statistical Accuracy .................................. 49
       6.3.1. Signal Recovery from Noisy Data ................ 49
       6.3.2. Signal Recovery from Grouped Data .............. 51

7. Reconstruction Algorithms for Shift Invariant Spaces 55
   7.1. Introduction ........................................... 55
   7.2. Basic Facts about Shift Invariant Spaces ............ 56
   7.3. General Purpose Reconstructions for non-uniform sampling .......... 59
       7.3.1. Linear Algebra Solution ......................... 61
       7.3.2. The Normal Equations ........................... 61
       7.3.3. Frame Approach .................................. 61
       7.3.4. Iterative Frame Methods ......................... 62
   7.4. Exploiting the Structure of the Problem ............... 62
       7.4.1. A Localization Property ......................... 63
       7.4.2. A Local Reconstruction Algorithm ............... 65
7.5. Reconstruction on a Lattice ............................................. 75

8. **Fast Recovery of Shift Invariant Functions from Local Averages** 79
   8.1. Fast Iterative Algorithm ............................................. 79
   8.2. Fast Algorithm for Reconstruction of Shift-Invariant Functions from Averages 81
       8.2.1. Reconstruction of shift invariant functions ................. 81
       8.2.2. Density Estimation for Histograms in Shift Invariant Spaces . 83
       8.2.3. Reconstruction of the Density from Histograms with the proposed
               Algorithm ...................................................... 85

9. **Conclusion and Summary** 93
   9.1. Algorithms for the space of bandlimited functions ............... 93
   9.2. Algorithms for the space of shift-invariant functions ........... 96

A. **Program-Codes** 99
   A.1. Reconstruction Algorithm for Band-limited Function from Averages . . 100
   A.2. Local Reconstruction Algorithm for a Shift-invariant Function ........ 102
   A.3. Local Reconstruction Algorithm for a Shift-invariant Function on a Lattice
        using Segmentation ................................................. 104
   A.4. Reconstruction of the Density-Function from a given Uniform Histogram 107
   A.5. Reconstruction of the Density-Function from a given Nonuniform Histogram 109

B. **Bibliography** 117

**List of Figures** 123
1. German Summary


Hierbei liegt folgende Strategie zu Grunde: *Die Entwicklung eines effizienten Algorithmus ist nur dann möglich, sofern wir die Beschaffenheit des Problems berücksichtigen!* Darunter verstehen wir im Wesentlichen drei Punkte:

1. Gewinnung der Daten

2. Voraussetzung an die Rekonstruktion
Da in der Regel das Rekonstruieren einer Funktion ein schlecht gestelltes Problem ist müssen zusätzliche Voraussetzungen betreffend der Rekonstruktion gemacht werden. Diese sind meist Glattheitsvoraussetzungen oder allgemeiner der Raum in welchem die rekonstruierte Funktion enthalten sein soll. In dieser Arbeit werden diesbezüglich zwei unterschiedliche Voraussetzungen getroffen. In den Kapiteln 5 und 6 gehen wir von der
Annahme aus, dass die Signale band-begrenzte Funktionen \( f \in B_\Omega \) sind, wogegen in den Kapiteln 7 und 8 allgemeiner vorausgesetzt wird, dass es sich um translationsinvariante Funktionen \( f \in V(\varphi) \) handelt. Die zwei grundlegenden Voraussetzungen bedingen auch eine Zwei teilung dieser Arbeit. Sowohl für den Fall \( f \in B_\Omega \) als auch für \( f \in V(\varphi) \) wird zuerst die Theorie für das Rekonstruieren aus exakten Daten präsentiert und im Anschluss diese Resultate verallgemeinert für den Fall von Rekonstruktionen aus Mit telwerten. Dadurch soll auch die Tatsache bekräftigt werden, dass Resultate aus der "Sampling-Theory" verallgemeinert werden können zu Ergebnissen bezüglich "Reconstruction from Averages".

3. Struktur des Funktionenraumes
Für die Entwicklung von effizienten Algorithmen muss die jeweilige Struktur des Raumes, in welchem wir rekonstruieren, berücksichtigt werden. Obwohl der Raum der band begrenzten Funktionen eine Teilraum der translationsinvarianten Funktionen ist, lohnt es diesen gesondert zu betrachten um von den speziellen Eigenschaften zu profitieren. Diese Eigenschaften sind in dem Fall \( f \in B_\Omega \) die Tatsache, dass die Anzahl der relevanten Koeffizienten eines bandbegrenzten Signals in der Regel viel geringer ist als die Länge des Signals selbst. Wir zeigen, dass das Problem der Rekonstruktion von bandbegrenzten Funktionen aus gemittelten Werten zurückgeführt werden kann auf das Problem das lineare Gleichungssystem

\[
c_{\text{given}} = S \cdot c_{\text{new}}
\]


Für den Fall der translationsinvarianten Signale \( f \in V(\varphi) \) konzentrieren wir uns auf die Eigenschaft, dass der Generator eines solchen Raumes kompakten Träger hat. Die führt dazu, dass in Kapitel 7.4.2 vorgestellten Algorithmus ein lineares Gleichungssystem der Form

\[
b = T \cdot c
\]

gelöst werden muss. Wie wir zeigen ergibt sich die Matrix \( T \) als Bandmatrix, dessen Bandbreite von der Länge des Trägers vom Generator abhängt, welcher in der Regel um vieles kleiner ist als die Länge des zu rekonstruierenden Signals. Für die Lösung des Gleichungssystems verwenden wir das Band-Cholesky Verfahren, wodurch die Kosten des gesamten Algorithmus linear zu der Anzahl der vorgegebenen Daten sind, was in diesem Sinne einer optimalen Verarbeitung entspricht.

Im Kapitel 7.5 wird der Algorithmus, durch Verwendung der lokalen Rekonstruktionseigenschaft, optimiert für den Fall, dass die Rekonstruktion auf einem regelmässigen Gitter stattfindet.
In beiden Fällen, Kapiteln 6.2 und 7.4, enden wir mit neuen effizienten Algorithmen, für die im Anhang A auch der Programm-Code zur Verfügung gestellt wird.

Einige Beispiele werden angeführt darunter auch in Kapitel 8.2.3 das Problem der Approximation der Dichtefunktion aus einem gegebenen Histogramm. Normalerweise wird das Problem betrachtet wie die Dichtefunktion rekonstruiert wird unter Verwendung der Daten, welche dem Histogramm zugrunde liegen. Wir betrachten hier das Problem der Approximation der Dichtefunktion aus dem Histogramm selbst, das heißt ohne Kenntnis der Messdaten.


Wichtig für die theoretische Behandlung des Problems bezüglich Fehlerabschätzungen und Konvergenzbedingungen ist die verwendete Frame-Theorie sowie Eigenschaften des Wiener-Amalgam Raumes.
2. Acknowledgements

This thesis was written during my work at NuHAG (Numerical Harmonic Analysis Group at the Department of Mathematics, University of Vienna), headed by Prof. Hans G. Feichtinger. I am deeply grateful to Prof. Feichtinger not only for advising this thesis but also for introducing me to the world of scientific work. With his enthusiastic engagement on research within the mathematical international community he encouraged me to study one semester at Northwestern University (Illinois, USA) and also to attend conferences, which have been important experiences. One great benefit of this working group is the great number of international visitors, which not only allows for the exchange of scientific topics but for cultural enrichment as well.

Furthermore I want to thank Charly Gröchenig for the very instructive cooperation on our joint paper, him and Prof. Johann Cigler for accepting the task of reviewing this PhD-thesis and all the local colleagues at NuHAG, especially Tobias Werther, for his help in finishing this work, Norbert Kaiblinger and Monika Dörfler, for motivation and countless discussions.

The manner in which this working group is led and the way of cooperative development of new scientific knowledge have been the motivation for me to continue after a year of intermission my PhD-thesis.

Last but not least, I want to thank my family and my friends for their patience and understanding, especially Alice, who motivated and assisted me in a lot of ways.
3. Introduction

3.1. Motivation

Here we want to give a short motivation why in some cases it makes more sense to reconstruct a function from averages and not from given samples.

Let’s have a look at a very simple and trivial example to illustrate the problem: the velocity of a bicycle. Let us assume that the tachometer of a bicycle updates the velocity every 5 seconds. After this fixed time-interval the velocity is calculated and displayed. Figure 3.1 shows a short shot of 30 seconds. Obviously we obtain a step function from the measurements of the tachometer. The problem we want to solve is to reconstruct the true continuous velocity function, i.e., we want to know the velocity at any arbitrary time, which can be viewed as an arbitrary improvement of the resolution. The unknown velocity function is plotted as a dotted line.

As we can see from this trivial example we cannot use the midpoints of our 5 seconds measurements as samples for our reconstruction because the velocity function need not to pass these points. The solution of this problem is to view the data of the tachometer as averages over a known area. With this approach we are able to receive a good reconstruction, which is plotted in figure 3.1 as a dash-dotted line and coincides with the original velocity function.

Even this very simple example yields to an important conclusion, which should be taken as motivation for this thesis: To get a higher level of resolution of given measurements or observations there is often no need for better or more sensitive measuring instruments. The same or even better results can be received by a correct interpretation of the given data.

In most applications the parameters of the measuring instruments are known to a high degree of accuracy (like the 5 sec. measurement intervals in our example) which means that the right interpretation of the data is to look at them as averages and not as samples at certain points.

Of course one can argue that we can a good approximation of the velocity function if use a denser sampling rate, but this is much more demanding on the measuring instruments and also increases the amount of data. Therefore it is worth studying the mathematical theory and to develop efficient algorithms for reconstruction from averages.
Coming back to our simple example of the bicycle velocity. The reader should look at this example only as motivation and not as application of the theory, but we can use it to show another situation for averaging:

Having a closer look how a tachometer of a bicycle may work we can verify that not the covered distance is measured in a certain time interval, but also the other way around. The rotations of a wheel are counted and after a certain covered distance the time is used to compute the velocity. With this kind of velocity measurements the tachometer step function can be plotted as seen in figure 3.2. Since the averages are taken over a fixed distance (in our example over 8 meters) the time intervals measured vary in length. This example describes another type of the averaging problem but we will see in the following chapters that this problem can be handled in a satisfactory way as figure 3.2 shows.

Later on we will study situations where the intervals of averaging are overlapping or do not cover the whole area we want to recover. This will lead us to introduce so called averaging functions, which define in a more general way how the averages are taken from an unknown function.
3.2. Efficient Algorithms

3.2.1. Sampling versus Averaging?

As soon as the concept of reconstruction from averages is accepted the question arises whether the numerous results of sampling theory can be used for averaging sampling. With this work we want to emphasize that results from sampling theory can be generalized to the case of averages. Starting with a result of Feichtinger and Gröchenig in *Theory and Practice of Irregular Sampling* [28], who provide an average version for the *weighted frame algorithm* several results of sampling theory have been generalized e.g. by Sun and Zhou [70–72]. Aldroubi [2] and Aldroubi/Feichtinger [4] also extended their algorithms to average sampling.

Consequently this work is structured in one part where sampling results are presented, followed by another part where these results are extended to the case of averaging sampling.

3.2.2. Exploring the Structure

This work pursues the following strategy: *We can only expect an efficient fast algorithm if we explore the structure of the problem!*

Exploring the structure has to be done in three steps:
Chapter 3

**Step 1:** First of all we have to know how the data we are using as input have been generated, i.e. we have to know some features of our averaging functions $u_k$ like the support, the position, etc.

**Step 2:** The space in which we want to reconstruct or approximate the unknown function has to be determined.

**Step 3:** Within this space the characteristic features have to be used to develop an efficient algorithm.

As an example for an impressing result of this strategy the reader is referred to the paper *Efficient numerical methods in non-uniform sampling theory* [29] by Feichtinger, Gröchenig and Strohmer. In this paper it is shown that the problem of recovering a band-limited function from its non-uniformly sampled values yields to a system matrix with Toeplitz-type structure. This special feature in combination with adaptive weights and conjugate gradient method produce the *ACT-Reconstruction Algorithm*, one of the most efficient ones for reconstruction bandlimited functions.

As a consequence this work is structured into two main parts, depending which space we are using for reconstruction:

- The space of band-limited functions;
- The space of shift-invariant functions;

Although the space of band-limited functions is a sub-space of the space of shift-invariant functions we handle this case separately to benefit from its characteristics. For the case of shift invariant functions we concentrate on those which are generated by a generator of compact support.

Following the approach described above, we end up in both cases with new efficient algorithms.

- In the case of band-limited functions we make use of the fact that the number of Fourier-coefficients is much smaller than the length of a discrete finite signal. Therefore, the costs of the new proposed algorithm primarily depend on the smoothness of the sought function and not on its length (see chapter 6).

- For the second part we show that the reconstruction of a shift invariant function with compact generator yields to a linear system with a symmetric band matrix. This includes a lot of desirable features, which are used to develop a new efficient algorithm (see chapter 7 and 8).
4. Notations and Definitions

In this chapter we want to present a collection of the most important definitions and notations which we are using in this thesis. Also a view characteristics of the presented objects are given.

1. For $f \in L^2(\mathbb{R})$ the **Fourier transformation** $\mathcal{F}f = \hat{f}$ is defined by

$$\mathcal{F}f(\omega) = \hat{f}(\omega) = \int_{\mathbb{R}} f(x)e^{-ix\omega}dx$$

and its inverse is

$$\mathcal{F}^{-1}\hat{f}(x) = f(x) = \frac{1}{2\pi} \int_{\mathbb{R}} \hat{f}(\omega)e^{ix\omega}d\omega$$

From this we have

$$\langle \hat{f}, \hat{g} \rangle = 2\pi \langle f, g \rangle$$

2. The **convolution** of two function $f, g \in L^1(\mathbb{R}^d)$ is given by

$$(f * g)(x) = \int_{\mathbb{R}^d} f(y)g(x - y)dy.$$ 

Following estimation is very useful for $g \in L^p(\mathbb{R}^d)$:

$$\|f * g\|_p \leq \|f\|_1\|g\|_p.$$ 

Using the Fourier transformation the convolution can be done by pointwise multiplication:

$$\hat{(f * g)} = \hat{f} \circ \hat{g}$$

3. A function $f$ is called a **band-limited functions with bandwidth** $2\Omega > 0$, if $f \in B_\Omega$ where

$$B_\Omega = \{f : \left(\int_{\mathbb{R}} |f(x)|^2dx\right)^{1/2} < \infty \text{ and } \text{supp} (\hat{f}) \subseteq [-\Omega, \Omega]\}.$$
Chapter 4

Therefore every \( f \in B_\Omega \) can be presented as
\[
f(x) = \frac{1}{2\pi} \frac{\sin \Omega(t-x)}{\pi(t-x)} = \frac{\Omega}{\pi} \overline{f}(x) \text{sinc}_\Omega(t) \text{ for } x \in \mathbb{R};
\]

We define \( \chi_A \) as the characteristic function of a set \( A \) and \( T_x f(t) = f(t-x) \) as the translation by \( x \). Then we have following properties for the space of band-limited functions:

- \( \mathcal{F}^{-1} \left( e^{-ix\omega} \chi_{[-\Omega,\Omega]}(\omega) \right)(t) = \frac{\sin \Omega(t-x)}{\pi(t-x)} = \frac{\Omega}{\pi} T_x \text{sinc}_\Omega(t) \text{ for } x \in \mathbb{R}; \)
- \( \left\{ \frac{1}{\sqrt{2\Omega}} e^{\frac{i\pi k\omega}{\Omega}}, k \in \mathbb{Z} \right\} \) is an orthonormal basis for \( L^2(-\Omega,\Omega) \) and therefore \( \left\{ \sqrt{\frac{\Omega}{2\pi}} T_{\frac{2\pi k}{\Omega}} \text{sinc}_\Omega, k \in \mathbb{Z} \right\} \) is an orthonormal basis of \( B_\Omega \);
- \( B_\Omega \) is a reproducing kernel Hilbert space, i.e. point evaluations \( f \to f(x) \) are continuous on \( B_\Omega \):
  \[
f(x) = \frac{\pi}{\Omega} f \ast \text{sinc}_\Omega(x) \text{ for } f \in B_\Omega.
\]

4. For an ordered sequence \( \cdots < x_{k-1} < x_k < x_{k+1} < \ldots \) the maximal gap size of sampling points is described in the form of \( \delta \), where
\[
\delta = \text{sup} \left( x_{k+1} - x_k \right).
\]

5. A sequence \( \{\varphi_k : k \in \mathbb{N}\} \) in a Hilbert space \( \mathcal{H} \) is said to be a frame if there exist positive constants \( C_L \) and \( C_U \) such that
\[
C_L \|f\|^2 \leq \sum_{k \in \mathbb{N}} \left| \langle f, \varphi_k \rangle \right|^2 \leq C_U \|f\|^2
\]
for every \( f \in \mathcal{H} \). The optimal numbers \( C_L \) and \( C_U \) are called the lower and upper frame bounds respectively.

6. \( \{\varphi_k : k \in \mathbb{Z}\} \) is called a Riesz basis if it is complete in \( \mathcal{H} \) and there exist positive constants \( C_L \) and \( C_U \) such that
\[
C_L \sum_{k \in \mathbb{Z}} |c_k|^2 \leq \sum_{k \in \mathbb{Z}} c_k \varphi_k \|2 \leq C_U \sum_{k \in \mathbb{Z}} |c_k|^2, \quad \forall \{c_k\} \in \ell^2.
\]

7. \( \mathcal{S} \) is called a frame operator with relaxation parameter \( \lambda > 0 \) if \( \{\varphi_k\} \) is a frame and
\[
S_\lambda f = \sum_{k=1}^{\infty} \lambda \langle f, \varphi_k \rangle \varphi_k.
\]
\{S^{-1}\varphi_k\} is called the dual frame of \{\varphi_k\} and then the frame expansion is of the form \( f = \sum_k \langle f, S^{-1}\varphi_k \rangle \varphi_k \).

8. \( \{u_k\} \) in \( L^2(\mathbb{R}^d) \) are called averaging functions centered at \( \{x_k\} \) of width \( \mu \) if they have following properties

   (1) \( \text{supp } u_k \subset [x_k - \frac{\mu}{2}, x_k + \frac{\mu}{2}] \), for \( \mu > 0 \)
   (2) \( u_k(x) \geq 0 \),
   (3) \( \int u_k(x) dx = 1 \).

Therefore \( \mu \) controls the length of the support of the averaging functions \( u_k \) and the local averages are given by

\[ \langle f, u_k \rangle = \int f(x) u_k(x) dx. \]

9. The shift invariant space generated by the generator \( \varphi \) and \( a > 0 \) is defined as

\[ V_a^p(\varphi) = \left\{ f(\cdot) = \sum_{k \in \mathbb{Z}^d} c_k T_{ak} \varphi(\cdot), \text{ for } (c_k) \in \ell^p(\mathbb{Z}^d) \right\} \subset L^p(\mathbb{R}^d). \]

To guarantee stability of these representations we have following assumption on the generator \( \varphi \): For \( V^p(\varphi) \subset L^p(\mathbb{Z}) \) we consider that there exist two constants \( m > 0, M > 0 \) for which

\[ m \|c\|_p \leq \| \sum_{k \in \mathbb{Z}} c_k T_k \varphi \|_p \leq M \|c\|_p \quad \forall c \in \ell^p. \]

10. For \( 1 \leq p < \infty \) the Wiener amalgam space \( W(L^\infty, \ell^p) \) consists of all measurable functions, for which the norm

\[ \|f\|_{W(L^\infty, \ell^p)} = \left( \sum_{k \in \mathbb{Z}} \sup_{x \in [0,1]} |f(x + k)|^p \right)^{1/p} < \infty \quad \text{(4.1)} \]

\( W(C, \ell^p) \) defines the subspace of continuous function in \( W(L^\infty, \ell^p) \).

Some very useful fact about Wiener amalgam spaces:

1. If \( \varphi \in W(C, \ell^1) \) then \( V^p(\varphi) \subset W(C, \ell^p) \quad 1 \leq p \leq \infty \)
2. \( W_0^p = W(C, L^p) \subset W(L^p) \) is also a Banach space [24].
3. Let \( f \in L^p(\mathbb{R}^d) \) and \( g \in W(L^1(\mathbb{R}^d)) \), then \( f * g \in W(L^p(\mathbb{R}^d)) \) and we have
\[
\|f * g\|_{W(L^p)} \leq C\|f\|_{L^p}\|g\|_{W(L^1)}.
\]

4. Let \( c \in \ell^p(\mathbb{Z}^d) \) and \( \varphi \in W(L^1(\mathbb{R}^d)) \), then
\[
\sum_{k \in \mathbb{Z}^d} c_k \varphi(\cdot - k) \leq \|c\|_{\ell^p}\|\varphi\|_{W(L^1)}.
\]

5. If \( f \in L^p(\mathbb{R}^d) \) and \( g \in W(L^1(\mathbb{R}^d)) \), then the sequence \( d \) defined by
\[
d_k = \int_{\mathbb{R}^d} f(x)g(x - k)\,dx, \quad k \in \mathbb{Z}^d,
\]
belongs to \( \ell^p(\mathbb{Z}^d) \) and we have
\[
\|d\|_{\ell^p} \leq \|f\|_{L^p}\|g\|_{W(L^1)} \quad 1 \leq p \leq \infty.
\]

11. A closed subspace \( \mathcal{H} \subset L^2(\mathbb{R}) \) of continuous functions on \( \mathbb{R} \) is a **reproducing kernel Hilbert space (RKHS)**, if for any \( x_0 \in \mathbb{R} \), the pointwise evaluation \( f \to f(x_0) \) is a bounded linear functional on \( \mathcal{H} \).

For the case \( p = 2 \) the additional structure of a RKHS. By the Riesz Representation Theorem, there exist \( K_{x_0} \in \mathcal{H} \), such that
\[
f(x_0) = \langle f, K_{x_0} \rangle.
\]
Let \( \overline{K_{x_j}} = S^{-1}K_{x_j}, j \in \mathbb{Z} \) defines the dual frame then following reconstruction is given:
\[
f(x) = \sum_{j \in J} \langle f, K_{x_j} \rangle \overline{K_{x_j}} = \sum_{j \in J} f(x_j) \overline{K_{x_j}}(x).
\]

12. The **Zak transform** is given by
\[
Zf(x, \omega) = \sum_{k \in \mathbb{Z}} f(x + k)e^{-ik\omega}.
\]

13. A **bounded partition of unity (BUPU)** associated with \( \{B_{\gamma}(x_j)\}_{j \in J} \) is a set of functions \( \{\beta_j\}_{j \in J} \) that satisfy
\[
1. 0 \leq \beta_j(x) \leq 1, \quad \forall j \in J;
2. \text{supp} (\beta_j) \subset B_{\gamma}(x_j);
3. \sum_{j \in J} \beta_j = 1.
\]
14. A set $X = \{x_j, j \in J\}$ is $\gamma_0$-dense in $\mathbb{R}^d$ if

$$\mathbb{R}^d = \bigcup_j B_\gamma(x_j)$$

for every $\gamma > \gamma_0$,

where $B_\gamma(x_j)$ are open balls centered on $x_j$ and with radius $\gamma$.

15. A kernel function $K$ satisfies the following properties

(i) $K(u) = K(-u) \geq 0$

(ii) $K(u) = 0$ for $|u| > 1$

(iii) $K(0) \geq K(u)$ for $u \neq 0$

(iv) $\int_{-1}^{1} K(u) = 1$

One well known approach (the kernel density estimation KDE) of finding an approximation $f_A$ for the unknown density function $f$ is obtained based on a kernel function $K(u)$ and a bandwidth $h$ as

$$f_A(x) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x - X_i}{h}\right).$$
Chapter 4
5. Iterative Reconstruction Algorithms for Bandlimited Functions

In this chapter we want to present some elementary facts about the reconstruction of bandlimited functions. In the case of equally spaced sampling we start with the famous sampling theorem by Shannon, Whittaker, Kotel’nikov and continue by introducing frame-theory to the case of irregular sampling. We only describe those theorems and algorithms, which are used later on in chapter 6 for reconstruction in the case of averaging.

Since a signal \( f(x) \) cannot be recorded in its entirety, it is sampled at a sequence \( \{x_k, k \in \mathbb{Z}\} \). The problem we want to handle is to reconstruct or at least to approximate the original function \( f \) from the given samples \( f(x_k) \). For theoretical study we assume that an infinite sequence of samples are given. Since for an arbitrary signal this reconstruction problem arises to be an ill-posed problem, we have to make some assumptions. One reasonable and widely used assumption is that the signal \( f \) does not contain high frequencies, i.e., mathematically spoken, \( f \) belongs to the space of bandlimited functions \( B_\Omega \). Recall the definition of the Fourier transformation:

\[
\mathcal{F}f(\omega) = \hat{f}(\omega) = \int_{\mathbb{R}} f(x) e^{-ix\omega} dx
\]

and its inverse is

\[
\mathcal{F}^{-1}\hat{f}(x) = f(x) = \frac{1}{2\pi} \int_{\mathbb{R}} \hat{f}(\omega) e^{ix\omega} d\omega
\]

Then we can define the space of band-limited functions:

**Definition 1 (Fourier Transformation).**

For \( f \in L^2(\mathbb{R}) \) the **Fourier transformation** \( \mathcal{F}f = \hat{f} \) is defined by

\[
\mathcal{F}f(\omega) = \hat{f}(\omega) = \int_{\mathbb{R}} f(x) e^{-ix\omega} dx
\]

and its inverse is

\[
\mathcal{F}^{-1}\hat{f}(x) = f(x) = \frac{1}{2\pi} \int_{\mathbb{R}} \hat{f}(\omega) e^{ix\omega} d\omega
\]

**Definition 2 (Band-limited Function).** A function \( f \) is called a **band-limited functions with bandwidth** \( 2\Omega > 0 \), if \( f \in B_\Omega \) where

\[
B_\Omega = \{f : \left(\int_{\mathbb{R}} |f(x)|^2 dx\right)^{1/2} < \infty \text{ and } \text{supp} (\hat{f}) \subseteq [-\Omega, \Omega]\},
\]

i.e.

\[
f(x) = \frac{1}{2\pi} \int_{-\Omega}^{+\Omega} \hat{f}(\omega) e^{ix\omega} d\omega.
\]
For the following considerations the constant $\delta$ should describe the gap size of the sampling points. For an ordered sequence $\ldots < x_{k-1} < x_k < x_{k+1} < \ldots$ we define

$$\delta = \sup_{k \in \mathbb{Z}}(x_{k+1} - x_k).$$  \hfill (5.4)

In the year 1949 Shannon presented in [66] a solution of $L^2$ signals of bounded bandwidth by using the cardinal series

$$\sum_{k=-\infty}^{+\infty} c_k \frac{\sin \pi(x - k)}{\pi(x - k)}$$

The result goes back to J.W. Whittaker [75] at the time, when Russian literature Kotel’nikov [51] introduced the same sampling theorem to communications theory.

For the situation that the samples are equally spaced we can formulate the famous sampling theorem by Shannon, Whittaker and Kotel’nikov for an explicit reconstruction:

**Theorem 1 (Shannon, Whittaker, Kotel’nikov).**

Let $f \in B_{\Omega}$ and $\delta$ defined by (5.4). Then for any

$$\delta \leq \frac{\pi}{\Omega},$$  \hfill (5.5)

and

$$x_k = k \cdot \delta$$  \hfill (5.6)

the function $f$ can be reconstructed from its samples $f(x_k)$ by

$$f(x) = \sum_{k=-\infty}^{+\infty} f(x_k) \text{sinc}_\Omega(x - x_k),$$  \hfill (5.7)

where $\text{sinc}_\Omega(x) = \frac{\sin(\Omega x)}{\Omega x}$ for $x \neq 0$ and $\text{sinc}_\Omega(0) = 1$.

The situation of regular sampling in the space of bandlimited functions is well studied and for more details the reader is referred to [12, 42, 46, 50, 78].

As described in [9] we can formulate a minimum rate at which each $f \in B_{\Omega}$ must be sampled for exact reconstruction:

**Theorem 2.** Let $f$ be in $B_{\Omega}$ for some $\Omega > 0$. Then for every $\alpha > 0$ with $0 < \alpha \leq \frac{1}{2\Omega}$, we have

$$f(x) = \alpha \sum_{k \in \mathbb{Z}} f(\alpha k) \text{sinc}_\Omega(x - \alpha k),$$  \hfill (5.8)

where the convergence is uniform and in $L^2(\mathbb{R})$.

This condition $0 < \alpha < \frac{1}{2\Omega}$ is necessary [9] and the rate of critical sampling is called Nyquist rate.

To study the case of irregular sampling, i.e., where the samples $\{x_k : k \in \mathbb{Z}\}$ are not equally spaced, we take a closer look on the space of bandlimited functions $B_{\Omega}$. The space of bandlimited function $B_{\Omega}$ is a closed subspace of $L^2(\mathbb{R})$ with the following properties:
Properties 1 (Space of Band-limited Functions). Let $\chi_A$ be the characteristic function of a set $A$ and $T_x f(t) = f(t - x)$ be the translation by $x$.

1. Then for $x \in \mathbb{R}$

$$
\mathcal{F}^{-1} \left( e^{-ix\omega} \chi_{[-\Omega, \Omega]}(\omega) \right)(t) = \frac{\sin \Omega(t - x)}{\pi(t - x)} = \frac{\Omega}{\pi} T_x \text{sinc}_\Omega(t)
$$

(5.9)

2. $\left\{ \frac{1}{\sqrt{2\Omega}} e^{i\pi k \omega/\Omega}, k \in \mathbb{Z} \right\}$ is an orthonormal basis for $L^2(-\Omega, \Omega)$ and therefore $\left\{ \sqrt{\frac{\Omega}{2\pi}} T_k \text{sinc}_\Omega, k \in \mathbb{Z} \right\}$ is an orthonormal basis of $B_\Omega$

3. $B_\Omega$ is a reproducing kernel Hilbert space, i.e. point evaluations $f \rightarrow f(x)$ are continuous on $B_\Omega$:

For $f \in B_\Omega$

$$
f(x) = \frac{1}{2\pi} \int_{\mathbb{R}} \hat{f}(\omega) e^{i\pi k \omega/\Omega} \chi_{[-\Omega, \Omega]}(\omega) d\omega
$$

$$
= \int_{\mathbb{R}} f(t) \frac{\sin \Omega(t - x)}{\pi(t - x)} dx
$$

$$
= \frac{\Omega}{\pi} \langle f, T_x \text{sinc}_\Omega \rangle
$$

$$
= \frac{\pi}{\Omega} f * \text{sinc}_\Omega(x).
$$

(5.14)

Now let us consider a few important consequences of these properties in the context of irregular sampling. If the orthogonal basis

$$
\left\{ e^{i\omega_k \pi}, k \in \mathbb{Z} \right\}
$$

is perturbed to

$$
\left\{ e^{i\omega x_k}, k \in \mathbb{Z} \right\},
$$

(5.11)

where

$$
\sup_{k \in \mathbb{Z}} \left| x_k - \frac{\pi k}{\Omega} \right| \leq \delta
$$

(5.12)

Paley and Wiener [57] present the following result:

**Theorem 3 (Paley-Wiener).**

Suppose that the irregular sampling set $\{x_k, k \in \mathbb{Z}\}$ satisfies (5.12), with

$$
\delta < \frac{\pi}{4\Omega}
$$

(5.13)

Then there exists a unique sequence $\{g_k\} \subset B_\Omega$, so that every $f \in B_\Omega$ has the representations

$$
f(x) = \sum_{k \in \mathbb{Z}} f(x_k) g_k(x) = \frac{\Omega}{\pi} \sum_{k \in \mathbb{Z}} \langle f, g_k \rangle T_{x_k} \text{sinc}_\Omega(x).
$$

(5.14)
Chapter 5

The series converges in $L^2(\mathbb{R})$ and uniformly on compact sets. The collections $\{g_k, k \in \mathbb{Z}\}$ and $\{T_{x_k} \text{sinc}_\Omega, k \in \mathbb{Z}\}$ are Riesz bases of $B_\Omega$ and $\frac{\Omega}{\pi}(T_{x_k} \text{sinc}_\Omega, g_n) = \delta_{kn}$ for $k, n \in \mathbb{Z}$.

Writing (5.12) as

$$\sup_{k \in \mathbb{Z}} \left| \frac{\Omega}{\pi} x_k - k \right| \leq \delta_0 \quad (5.15)$$

then the exact bound $\delta_0 = \frac{1}{4}$ is due to Kadec [49] and therefore Theorem 3 is sometimes referred to Kadec.

The first attempt to weaken the assumptions in Theorem 3 goes back to Levinson [53]. He proved that there exists $\{x_k\}_{k \in \mathbb{Z}} \subset \mathbb{R}$ for which equality is obtained in 5.13 and $\{e^{2\pi i x_k \Omega}\}_{k \in \mathbb{Z}}$ is complete but not a Riesz basis in $L^2$. Levinson’s statement implies every function $f \in B_\Omega$ can be written as $f = \sum_{k \in \mathbb{Z}} c_k \text{sinc}(\cdot - x_k)$, where the scalars $c_k$ are not necessarily unique [10].

Weakening the notation of basis the following theorem can be found in Duffin and Schaeffer 1952:

**Theorem 4 (Duffin-Schaeffer [21]).**

Suppose that $\{x_k\}$ is a real sequence and there are some constants $0 < \epsilon < 1$ and $0 < \alpha, \delta < \infty$ such that $|x_k - x_n| \geq \alpha$ for $n \neq k$ and

$$|x_k - \frac{\pi k}{\Omega}| \leq \delta. \quad (5.16)$$

then there exist two positive constants $C_L$ and $C_U$ depending only on $\epsilon, \alpha$ and $\delta$ such that for any $f \in B_\Omega$,

$$C_L \|f\|^2 \leq \sum_{k \in \mathbb{Z}} |f(x_k)|^2 \leq C_U \|f\|^2. \quad (5.17)$$

For understanding the consequences of Theorem 4 we have to introduce a very powerful tool: the frame theory. Frames in a Hilbert space were introduced by R. Duffin and A. Schaeffer as a generalization of Riesz bases in their fundamental work on irregular sampling of band-limited functions [21].

**Definition 3 (Frame).** A sequence $\{\varphi_k : k \in \mathbb{N}\}$ in a Hilbert space $\mathcal{H}$ is said to be a frame if there exist positive constants $C_L$ and $C_U$ such that

$$C_L \|f\|^2 \leq \sum_{k \in \mathbb{N}} |\langle f, \varphi_k \rangle|^2 \leq C_U \|f\|^2 \quad (5.18)$$

for every $f \in \mathcal{H}$. The optimal numbers $C_L$ and $C_U$ are called the lower and upper frame bounds respectively.

Since $f(x_k) = \frac{\Omega}{\pi}(f, T_{x_k} \text{sinc}_\Omega)$, the statement (5.17) in Theorem 4 is equivalent to $\{T_{x_k} \text{sinc}_\Omega, k \in \mathbb{Z}\}$ being a frame for $B_\Omega$. 

26
To receive a reconstruction algorithm based on frames we need the following reconstruction scheme presented by H.G. Feichtinger and K. Gröchenig 1994:

**Proposition 1 (Feichtinger and Gröchenig [28]).** Let $A$ be a bounded operator on a Banach space $(B, \| \cdot \|_B)$ that satisfies for some positive constant $\gamma < 1$

$$\|f - Af\|_B \leq \gamma \|f\|_B \quad \forall f \in B.$$  \hspace{1cm} (5.19)

Then $A$ is invertible on $B$ and $f$ can be recovered from $Af$ by the following iteration algorithm. Setting

$$f_0 = Af$$

$$f_{n+1} = f_n + A(f - f_n)$$ \hspace{1cm} (5.20)

for $n \geq 0$, we have

$$\lim_{n \to \infty} f_n = f$$ \hspace{1cm} (5.21)

with the error estimate after $n$ iterations

$$\|f - f_n\|_B \leq \gamma^{n+1} \|f\|_B.$$ \hspace{1cm} (5.22)

With Proposition 1 and (5.18) we can reconstruct $f$ from $\langle f, \varphi_k \rangle$ by introducing the so-called frame-operator:

**Definition 4 (Frame Operator).** $S$ is called a **frame operator** with relaxation parameter $\lambda$ if $\{\varphi_k\}$ is a frame and

$$S_{\lambda} f = \sum_{k=1}^{\infty} \lambda \langle f, \varphi_k \rangle \varphi_k,$$ \hspace{1cm} (5.23)

for $\lambda > 0$.

As we can find e.g. in [15] or [38] following statement holds:

**Theorem 5.** Given a sequence $\{\varphi_k\}$ in a Hilbert space $\mathcal{H}$, the following two statements are equivalent:

1. $\{\varphi_k\}$ is a frame with bounds $C_L, C_U$.

2. $Sf = \sum_k \langle f, \varphi_k \rangle \varphi_k$ is a bounded linear operator with $C_L I \leq S \leq C_U I$.

Moreover, the series $\sum_k \langle f, \varphi_k \rangle \varphi_k$ converges unconditionally.
The fact that $S$ is a positive operator, and the frame definition 3 implies
\[- \frac{C_U - C_L}{C_U + C_L} \|f\|^2 \leq \|f\|^2 - \frac{2}{C_L + C_U} \sum_k |\langle f, \varphi_k \rangle|^2 \]
\[= \langle (Id - S)f, f \rangle \]
\[\leq \frac{C_U - C_L}{C_U + C_L} \|f\|^2. \]

Since the operator norm of $Id - S$ on $H$ is at most $\frac{C_U - C_L}{C_U + C_L} < 1$ Proposition 1 can be applied, which leads us to an iterative reconstruction of $f$ from $Sf$.

Let define $g_k = S^{-1}\varphi_k$ and rewrite the reconstruction of $f$ from $\langle f, \varphi_k \rangle$ in the closed form
\[f = S^{-1}Sf = \sum_k \langle f, \varphi_k \rangle g_k. \tag{5.24} \]

The non-orthogonal expansion of $f$ with respect to the basis-functions $\varphi_k$ is given as
\[f = SS^{-1}f = \sum_k \langle S^{-1}f, \varphi_k \rangle \varphi_k = \sum_k \langle f, g_k \rangle \varphi_k. \tag{5.25} \]

For introducing the frame expansion and the dual frame expansion we present the following theorem [21, 34]:

**Theorem 6.** Following holds:

1. $S$ is invertible and $C_U^{-1}I \leq S^{-1} \leq C_L^{-1}$.
2. $\{S^{-1}\varphi_k\}$ is a frame with bounds $C_U^{-1}, C_L^{-1}$, called the dual frame of $\{\varphi_k\}$.
3. Every $f \in H$ can be written as
   \[f = \sum_k \langle f, S^{-1}\varphi_k \rangle \varphi_k \text{ (the frame expansion)} = \sum_k \langle f, \varphi_k \rangle S^{-1} \varphi_k \text{ (the dual frame expansion)}. \]

Theorem 6 and the representation $f = S^{-1}Sf$ allow to describe $f$ as limit of a sequence of partial sums, where $f^{(n)}$ is given by
\[f^{(n)} = \gamma \sum_{k=0}^{n} (I - \gamma S)^k S f. \tag{5.26} \]

An iterative description of this approximation algorithm is possible and useful for practical applications (cf. [23, 26]):

**Theorem 7.** The approximation of $f$ by the sequence of partial sums
\[f^{(n)} = \gamma \sum_{k=0}^{n} (I - \gamma S)^k S f \]
can be replaced by following iterative algorithm
\[ f^{(n)} = \gamma S f + (I - \gamma S)f^{(n-1)} = f^{(n-1)} + \gamma S(f - f^{(n-1)}), \]

where the iteration starts with
\[ f^{(0)} := \gamma S f. \]

A first approach to an approximation of a bandlimited function \( f \) through its non-uniform sampling values \( f(x_k) \) was introduced by Marvasti [55] and goes back to the natural sampling method by Wiley [76]. It consists in projecting the weighted sum of shifted Dirac measures \( \sum f(x_k)\delta_{x_k} \) into \( B_{\Omega} \) by means of convolution with the sinc-function. The corresponding linear approximation operator is equal to the frame operator defined in (5.23) with \( \varphi_k(x) = \text{sinc}_\Omega(x - x_k) \) and, hence, the reconstruction algorithm proposed by Marvasti is just a reformulation of the frame algorithm (see [67]).

The frame bounds \( C_L \) and \( C_U \) play an important role and determine the speed of convergence of the algorithm. Since the estimates of the frame bounds, specially the lower one, is usually difficult H.G. Feichtinger and K. Gröchenig provide in [28] the following way to handle this problem:

Consider the family of operators
\[ S_\lambda f = \lambda \sum_k \langle f, \varphi_k \rangle \varphi_k \]  
(5.27)

where \( \lambda \) is the so-called relaxation parameter.

For \( \gamma(\lambda) = \max \{ |1 - \lambda C_L|, |1 - \lambda C_U| \} \) and \( \gamma(\lambda) < 1 \) we get following estimate:
\[ \| f - S_\lambda f \| \leq \gamma(\lambda) \| f \| \]  
(5.28)

This provides a good estimation of the optimal value \( \gamma(\lambda_0) = (C_U - C_L)/(C_U + C_L) \) with the choice \( \lambda_0 = 2/(C_L + C_U) \).

Based on the theory of frames, Feichtinger and Gröchenig presented in [28] a new reconstruction method for the irregular sampling case of bandlimited functions: the adaptive weights method. This method produces very good numerical results and provides explicit frame bounds for the rate of convergence. First the weights are introduced by the distance between the midpoints of the samples, i.e.,
\[ w_k = \frac{x_{k+1} + x_k}{2} - \frac{x_k + x_{k-1}}{2} = \frac{x_{k+1} - x_{k-1}}{2} \]  
(5.29)

Again let be \( \delta \) defined as in (5.4), then following theorem can be formulated:
Chapter 5

Theorem 8 (Feichtinger and Gröchenig [28]).
Assume that
\[ \delta < \frac{\pi}{\Omega} \]
and \( w_k \) be defined as in (5.29).

(A) Weighted Frames
For any \( f \in B_\Omega \)
\[
\left( 1 - \frac{\delta \Omega}{\pi} \right)^2 \| f \|^2 \leq \sum_{k \in \mathbb{Z}} w_k |f(x_k)|^2 \leq \left( 1 + \frac{\delta \Omega}{\pi} \right)^2 \| f \|^2. \quad (5.30)
\]
Consequently, the collection \( \{ \sqrt{w_k \pi} T_{x_k} \sin \Omega, k \in \mathbb{Z} \} \) is a frame with frame bounds
\[
C_L = \left( 1 - \frac{\delta \Omega}{\pi} \right)^2 \text{ and } C_U = \left( 1 + \frac{\delta \Omega}{\pi} \right)^2. \quad (5.31)
\]

(B) The Adaptive Weights Method.
Any \( f \in B_\Omega \) can be reconstructed from its samples \( f(x_k) \) by the adaptive frame algorithm:
\[
f_0 = \frac{\pi^2}{\pi^2 + \delta^2 \Omega^2} \sum_{k \in \mathbb{Z}} f(x_k) \frac{\Omega}{\pi} T_{x_k} \sin \Omega := Sf
\]
\[
f_{n+1} = f_n + S(f - f_n) \quad n \geq 0
\]
with \( f = \lim_{n \to \infty} f_n \) in \( L^2(\mathbb{R}) \) and
\[
\| f - f_n \| \leq \gamma^{n+1} \| f \| \quad (5.32)
\]
where \( \gamma = \frac{2\pi \delta \Omega}{\pi^2 + \delta^2 \Omega^2} \).
6. Reconstruction from Averages for Bandlimited Functions

In this chapter we want to present a short overview of reconstruction from averages for bandlimited functions. Several theorems are presented, which generalize the results of chapter 5 to the case of averages. Based on a result of Gröchenig 1992 [32] and a modified version of the Adaptive Weights Method for the case of averaging (Feichtinger and Gröchenig [28]) Sun and Zhou [70, 71] extended these results and presented estimates of frame bounds for several scenarios. In chapter 6.2 we present a new effective algorithm, which makes use of the special structure of the space $B_\Omega$ and the numerical efficiency of the Conjugate-Gradient Method.

6.1. An Overview

In practice, the assumption that the samples $\{f(x_k) : k \in \mathbb{Z}\}$ can be measured exactly is not realistic and a better assumption is that an average value (or a weighted average value) in the neighborhood of $x_k$ is obtained.

Mathematically spoken we have to introduce the functions $u_k$, the so called averaging functions.

**Definition 5 (Averaging Functions).** The functions $\{u_k\}$ in $L^2(\mathbb{R}^d)$ are called **averaging functions** centered at $\{x_k\}$ of width $\mu$ if they have following properties

\begin{align*}
(1) & \quad \text{supp } u_k \subset [x_k - \frac{\mu}{2}, x_k + \frac{\mu}{2}], \text{ for } \mu > 0 \\
(2) & \quad u_k(x) \geq 0, \\
(3) & \quad \int u_k(x)dx = 1.
\end{align*}

Figure 6.1 shows four standard averaging functions with $\mu = 8$.

Figures 6.2 and 6.3 show the moving averages values of a given function by using the four standard averaging-functions (box, triangle, spline and cosine-flanks).
Figure 6.1.: Examples of standard averaging functions: box function, triangle function, spline function, box function with cosine flanks

Figure 6.2.: Original function (dotted line) and average values calculated by box function (upper figure) and triangle function (lower figure)

Figure 6.3.: Original function (dotted line) and average values calculated by spline function (upper figure) and box function with cosine flanks (lower figure)
Reconstruction from Averages for Bandlimited Functions

Then the local averages are given by
\[ \langle f, u_k \rangle = \int f(x) u_k(x) dx. \] (6.1)

Notice that now we have two constants to treat:
- \( \mu \), which controls the size of the support of the averaging functions;
- \( \delta \), which controls the distance of the sampling positions or, in the case of averaging, the distance of the centers of the averaging functions \( u_k \).

First let us assume that \( \mu = \delta \). If \( \delta \) is “small”, then we can expect that we should obtain at least a good approximation of the original function \( f \).

In the year 1992 Gröchenig proved that band-limited functions are completely determined by their local averages:

**Theorem 9 (Gröchenig [32]).**

Let be \( \delta \) defined as in (5.4) and \( u_k, \mu \) given as in definition 5.

If \( \mu \leq \delta < \frac{1}{\sqrt{2\Omega}} \) then every \( f \in B_\Omega \) is uniquely determined by the local averages \( \langle f, u_k \rangle \) around \( x_k \) and \( f \) can be reconstructed by an iterative algorithm.

The iterative algorithm is similar to that of Proposition 1:

The approximation operator is given as
\[ A_u = P \left( \sum_k \langle f, u_k \rangle \chi_k(x) \right), \] (6.2)

where \( P \) denotes the orthogonal projection from \( L^2(\mathbb{R}) \) onto \( B_\Omega \) and \( \chi_k \) the characteristic function of the set \( [x_k, x_{k+1}) \).

Then by applying the iterative algorithm (5.20) the error
\[ \| f - A_u f \|_2 \] (6.3)
has to be estimated. By defining the \( \delta \)-oscillation of \( f \)
\[ \text{osc}_\delta f(x) = \sup_{y:|y-x|\leq\delta} |f(x) - f(y)| \] (6.4)
one obtains
\[ \| f - A_u f \|_2 \leq \| \text{osc}_\delta f \|_2. \] (6.5)

In [32] it is shown that
\[ \| \text{osc}_\delta f \|_2 \leq \sqrt{2\delta\Omega} \| f \|_2 \] (6.6)
thus if \( \sqrt{2\delta\Omega} < 1 \) \( A_u \) is invertible on \( B_\Omega \) and Proposition 1 can be applied.

Later Theorem 8 was extended to the case of reconstruction by averages:
Theorem 10 (Feichtinger and Gröchenig [28]).

Let \( y_k = \frac{x_k + x_{k+1}}{2} \) be the midpoints of the sampling positions, and \( w_k = y_k - y_{k-1} \) be the weights. Then for

\[ \mu = \delta < \frac{\pi}{\Omega} \]

and the averaging functions

\[ u_k = \frac{1}{w_k} \chi_{(y_{k-1}, y_k)} \]

every \( f \in B_{\Omega} \) is uniquely determined by its averages \( \langle f, u_k \rangle \) and can be reconstructed by the following algorithm:

\[ f_0 = \sum_{k \in \mathbb{Z}} \langle f, u_k \rangle \frac{\pi}{\Omega} T_{x_k} \text{sinc}_\Omega \]

\[ f_{n+1} = f_n + f_0 - \sum_{k \in \mathbb{Z}} \left( \int_{y_{k-1}}^{y_k} f_n(x) dx \right) \frac{\pi}{\Omega} T_{x_k} \text{sinc}_\Omega \quad n \geq 0. \]

Then \( f = \lim_{n \to \infty} f_n \) in \( L^2(\mathbb{R}) \) and

\[ \| f - f_n \| \leq \left( \frac{\delta \Omega}{\pi} \right)^{n+1} \| f \|. \] (6.8)

In the last years Wechang Sun and Xingwei Zhou [69–71], presented several theorems for the case of average sampling, which are based on Theorem 8 (Weighted Frames), and Theorem 9 (Gröchenig [32]).

First we need the following lemma:

**Lemma 1.** Suppose that \( \{ u_k(x) : k \in \mathbb{Z} \} \subset L^2(\mathbb{R}) \). If there exist two positive constants \( C_L \) and \( C_U \) such that

\[ C_L \| f \|^2 \leq \sum_{k \in \mathbb{Z}} |\langle f, u_k \rangle|^2 \leq C_U \| f \|^2, \quad \forall f \in B_{\Omega}, \] (6.9)

then there is a frame \( \{ \varphi_k(x) : k \in \mathbb{Z} \} \) for \( B_{\Omega} \) with bounds \( \frac{1}{C_U} \) and \( \frac{1}{C_L} \) such that for any \( f \in B_{\Omega}, \)

\[ f(x) = \sum_{k \in \mathbb{Z}} \langle f, u_k \rangle \varphi_k(x), \] (6.10)

where the convergence is both in \( L^2(\mathbb{R}) \) and uniform on \( \mathbb{R} \).

This lemma is needed to proof one of the main statements of [71]:
Reconstruction from Averages for Bandlimited Functions

Theorem 11 (Sun and Zhou [71]).
Let be \( \delta \) defined as in (5.4) and \( \mu \) as in definition 5, with
\[
\delta < \frac{\pi}{\Omega},
0 < \mu < \frac{\pi}{\Omega} - \delta,
\]
then there is a frame \( \{ \varphi_k(x) : k \in \mathbb{Z} \} \) for \( f \in B_\Omega \) with estimates
\[
\left(1 + \frac{\mu \Omega}{\sqrt{2}}\right)^{-2} \left(1 + \frac{\delta \Omega}{\pi}\right)^{-2} \leq C_L \quad \text{and} \quad C_U \leq \frac{81}{2} \left(1 - \frac{\delta - \mu \Omega}{\pi}\right)^{-3} \tag{6.11}
\]
such that for any \( f \in B_\Omega \),
\[
f(x) = \sum_{k \in \mathbb{Z}} \left(\frac{x_k+1 - x_k-1}{2}\right)^{1/2} \langle f, u_k \rangle \varphi_k(x), \tag{6.12}
\]
where the convergence is both in \( L^2(\mathbb{R}) \) and uniform on \( \mathbb{R} \).

Remark: This result of Sun and Zhou is an improvement of Theorem 9 by Gr"ochenig because of following argument: Let be \( \delta = \frac{\pi}{2\Omega} \). Then Theorem 11 says that every \( f \in B_\Omega \) is uniquely determined by \( \langle f, u_k \rangle \) for any sequence of averaging function \( \{u_k\} \) with
\[
\text{supp } u_k \subset [x_k - \frac{\mu}{2}, x_k + \frac{\mu}{2}], \quad 0 < \mu < \frac{\pi}{2\Omega} \tag{6.13}
\]
Since \( \frac{\pi}{2\Omega} > \frac{1}{\sqrt{2\Omega}} \) the result of Sun and Zhou contains the one of Gr"ochenig [32].

Notice that in this theorem \( \delta \) need not to be equal to \( \mu \). The connection between \( \delta \) and \( \mu \) can be viewed in the sense that the distance from \( \delta \) to its upper bound \( \frac{\pi}{2\Omega} \) is used to define the maximal length of support \( \mu \).

The next step is to focus on the problem that the sequence \( \{x_k\} \) is a perturbation of some regular sampling points.

Also the Theorem 4 of Duffin and Schaeffer is generalized by Sun and Zhou for the case of averaging sampling:

Theorem 12 (Sun and Zhou [71]).
Let \( \{x_k\}, \varepsilon, \alpha \) and \( \delta \) be defined as in Theorem 4, i.e.
\[
|x_k - \frac{\pi k}{\Omega}| \leq \delta
\]
and
\[
|x_k - x_n| > \alpha
\]
for \( 0 < \varepsilon < 1 \) and \( 0 < \alpha, \delta < \infty \), and \( \mu \) defined as in definition 5.
Chapter 6

Then for any $0 < \mu < \alpha$, there is a frame $\{\varphi_k(x) : k \in \mathbb{Z}\}$ for $B_\Omega$ such that for any $f \in B_\Omega$,

$$f(x) = \sum_{k \in \mathbb{Z}} \langle f, u_k \rangle \varphi_k(x),$$

where the convergence is both in $L^2(\mathbb{R})$ and uniform on $\mathbb{R}$.

Furthermore Sun and Zhou present in [71] frame bounds for the case that the $\{x_k\}$ are equidistantly spaced, i.e. $\varepsilon = 1$.

**Theorem 13 (Sun and Zhou [71]).**

Suppose that

$$x_k = \frac{k\pi}{\Omega} \quad (6.14)$$

and $\mu$ defined as in definition 5 with

$$\mu < \frac{\pi}{2\Omega}$$

Then there is a frame $\{\varphi_k(x) : k \in \mathbb{Z}\}$ for $B_\Omega$ with bounds

$$\frac{\pi}{\Omega} \left( 2 - \cos \frac{\mu\Omega}{2} + \sin \frac{\mu\Omega}{2} \right)^{-2} \text{ and } \frac{\pi}{\Omega} \left( \cos \frac{\mu\Omega}{2} - \sin \frac{\mu\Omega}{2} \right)^{-2} \quad (6.15)$$

such that for any $f \in B_\Omega$,

$$f(x) = \sum_{k \in \mathbb{Z}} \langle f, u_k \rangle \varphi_k(x), \quad (6.16)$$

where the convergence is both in $L^2(\mathbb{R})$ and uniform on $\mathbb{R}$. Furthermore the conclusion falls if $\mu > \frac{\pi}{2\Omega}$.

Until this point we have made no restriction on what the averaging functions look like. For the case that the averaging functions are box functions Sun and Zhou give following estimates of the frame bounds:

**Theorem 14 (Sun and Zhou [71]).**

For $0 < \varepsilon \leq 1$ let be

$$x_k = \frac{k\epsilon\pi}{\Omega} \quad (6.17)$$

and the averaging functions $u_k(x)$ given as

$$u_k(x) = \frac{1}{\mu_k} \chi_{[x_k-\frac{\mu_k}{2}, x_k+\frac{\mu_k}{2}]}(x). \quad (6.18)$$

If

$$0 < \mu_k \leq \mu < \frac{\pi}{\Omega}, \quad (6.19)$$
then there is a frame \( \{ \varphi_k(x) : k \in \mathbb{Z} \} \) for \( B_\Omega \) with estimates

\[
\frac{\epsilon \pi}{\Omega} \left( 1 + \frac{\mu \Omega}{\pi} \right)^{-2} \text{ and } \frac{\epsilon \pi}{\Omega} \left( 1 - \frac{\mu \Omega}{\pi} \right)^{-2}
\] (6.20)

such that for any \( f \in B_\Omega \),

\[
f(x) = \sum_{k \in \mathbb{Z}} (f, u_k) \varphi_k(x),
\]

where the convergence is both in \( L^2(\mathbb{R}) \) and uniform on \( \mathbb{R} \).

### 6.2. Fast Algorithm using CG-Method

In the chapters before we described reconstruction methods based on frame theory. Now we want to present a fast algorithm, which takes into account the special structure of problem, i.e. the structure of the frame-operator \( S \). This approach yields to an algorithm which combines frame theory and the technique of Conjugate Gradients (CG-method).

First we examine the situation, that the averages are always taken with respect to the same window \( u \). In this situation of moving averages a reduction to existing reconstruction algorithm is possible.

Figure 6.4 shows an example of moving averages (solid line). The average values of the original function (dotted line) are calculated by using a box function.
Lemma 2. If \( \hat{u} \) is a continuous function with \( \hat{u} \neq 0 \) on the compact set \([-\Omega, \Omega]\), then the reconstruction of \( f \) from the averages \((f * u(x_k))_{k \in \mathbb{Z}}\) is possible for all \( f \in B_\Omega \) if and only if reconstruction of \( g \) from \((g(x_k))_{k \in \mathbb{Z}}\) is possible for all \( g \in B_\Omega \).

Proof. Indeed, under the given assumptions the mapping \( f \mapsto f * u \) is just an invertible isomorphism from \( B_\Omega \) into itself. In particular there is some other bounded, continuous function \( \check{u} \) which is non-zero on \([-\Omega, \Omega]\) such that \( \check{u}(x) = 1/\hat{u}(x) \) on \([-\Omega, \Omega]\). Since \( g = f * u \) is band-limited (with \( \text{spec}(f) \subseteq [-\Omega, \Omega] \)) if and only if \( f \in B_\Omega \), the key argument for the lemma is the following one: Given the samples \((f * u(x_k))_{k \in \mathbb{Z}}\) we just use any existing method (iterative or not) to recover \( f * u \) first. Once this signal is known we obtain \( f \) by multiplying the Fourier transform of \( f * u \) with \( \check{u} \) and then go back to the time domain to obtain \( f \). \( \square \)

For the case that the averages are taken in the sense of a moving averages function \( \hat{u} \) it can be shown easily that any width is allowable, which is smaller than twice the Nyquist rate.

### 6.2.1. Interpretation as Frame

In order to describe new frame approach which will allow to recover a band-limited function \( f \) from local averages we have to set up some definitions. We assume for a sequence of well concentrates, real-valued functions \( u_k \) the scalar products \( \langle f, u_k \rangle \) to be given, for some \( f \in B_\Omega \), and we want to recover \( f \) from these data.
As various iterative methods a two-step method is used: First using the sampling information a function of the form $\sum_k c_k u_k$ is generated, which approximates $f$ as well as possible in the $L^2$-sense (see figure 6.5).

![Figure 6.5.: Original $f$ and first approximation $S_1(f)$, without mass compensation.](image)

Under the assumption made it is natural to choose the coefficients in such a way that the individual terms $c_k u_k$ are just the projections of $f$ in the direction of $u_k$. Adding those term the first overall approximation operator is then of the form

$$S_1(f) = \sum_k \langle f, u_k \rangle \frac{u_k}{\|u_k\|^2}. \quad (6.21)$$

Then to make sure that we have a symmetric operator with range within $B_\Omega$ we modify $S_1$ to the operator $S_\Omega$ in the following way:

$$S_\Omega(f) = S_1(f) \otimes \text{sinc}_\Omega = \sum_k \langle f, u_k \rangle \frac{1}{\|u_k\|^2} (u_k \otimes \text{sinc}_\Omega), \quad (6.22)$$

where $\otimes$ denotes convolution.

**Lemma 3.** Let $[-\Omega, \Omega]$ be a subset of the frequency domain of signals of length $N$. A collection $(u_k)$ of non-zero vectors allows to recover every $f \in B_\Omega$ from the scalar products $\langle f, u_k \rangle$ if and only if

$$(P_\Omega u_k) = (u_k \otimes \text{sinc}_\Omega), \quad (6.23)$$

the projections of $(u_k)$ form a frame for $B_\Omega$, respectively if and only if $S_\Omega$ is an invertible operator on $B_\Omega$.

In the positive case the family $(g_k)$, given as

$$g_k = \left( \frac{u_k}{\|u_k\|^2} \otimes \text{sinc}_\Omega \right) \quad (6.24)$$

is also a frame for $B_\Omega$, and the corresponding frame operator coincides with $S_\Omega$. 

39
Lemma 3 can be turned into several practical ways of recovering \( f \) from \( \langle f, u_k \rangle \). Since it is possible to generate \( S_\Omega(f) \), given only \( \Omega, (u_k) \) and \( \langle f, u_k \rangle \), one would only have to invert \( S_\Omega \) in order to obtain \( f = S_\Omega^{-1}(S_\Omega f) \). In practice this is only feasible if \( \Omega \) is relatively small and the pure frequencies from \( \Omega \) are chosen as orthonormal basis for \( B_\Omega \).

In most other cases iterative strategies have to be applied, such as the standard frame iteration or the CG-strategy.

### 6.2.2. Fast Efficient Algorithm using CG-Method

For describing the algorithm we have to formulate the reconstruction problem for the discrete case:

Let be \( f := (f_1, f_2, \ldots, f_N)^t \) a finite sequence of complex numbers in \( B_\Omega \), i.e., only DFT-coefficients inside \( \Omega \) may be non-zero.

Furthermore let \( \{v_k(\cdot)\}, k = 1, \ldots, K \) be a collection of indicator functions with disjoint supports, which are stored as columns of a matrix of size \( N \times K \)

\[
B = (v_1 | v_2 | \ldots | v_K).
\]  

(6.25)

![Figure 6.6: The indicator functions where the averages are taken are stored in the columns of the matrix \( B \). In this example we have seven averages with different length of support. Notice there are some gaps between the regions where the averages are taken.](image)

We also need two different normalized versions, namely

\[
B_1 = \left( \frac{v_1}{\|v_1\|_1} | \frac{v_2}{\|v_2\|_1} | \ldots | \frac{v_K}{\|v_K\|_1} \right)
\]  

(6.26)

and

\[
B_2 = \left( \frac{v_1}{\|v_1\|_2} | \frac{v_2}{\|v_2\|_2} | \ldots | \frac{v_K}{\|v_K\|_2} \right).
\]  

(6.27)
Clearly the following equation hold:
\[ B \ast B_1^t = B_2 \ast B_2^t. \] (6.28)

Furthermore let be the average functions \( u_k \) defined as
\[ u_k = \frac{v_k}{\|v_k\|_1}, \quad k = 1, \ldots, K. \] (6.29)

Thus, by knowing the scalar products \( \langle f, u_k \rangle \), we also know the “average” values given by
\[ m = (m_k) = B_1^t \ast f \] (6.30)
(see figure 6.7).

Figure 6.7.: This figure shows the unknown function and the triangles indicate over which regions the averages are taken. The values of the triangle corresponds to the values \( m_k, k = 1 \ldots 7. \)

Therefore it is our aim to reconstruct the original signal \( f \) from the sequence \( m := (m_1, m_2, \ldots, m_K)^t. \)
As mentioned above the original signal \( f \) is band-limited with maximal frequency \( \Omega \), i.e., the original signal has at most \( R = 2\Omega + 1 \) relevant coefficients. Notice that
\[ R \leq K \leq N. \] (6.31)

Using this information we can define the matrix \( F_\Omega \) of size \( R \times N \) representing the Fourier transformation including a projection from \( B_\infty = B_N \rightarrow B_\Omega \). Analogously the corresponding Inverse Fourier transform matrix \( F_\Omega^{-1} \) is of size \( N \times R \), with \( F_\Omega^{-1} = (F_\Omega)^t. \)

Since the function \( f \) is completely terminated by its coefficients and equation (6.31) implies that the number of relevant coefficients is much smaller than the length of the unknown signal \( f \) we concentrate on reconstructing these unknown coefficients
\[ c = (c_1, c_2, \ldots, c_R)^t. \]
Chapter 6

First we describe the averages of (6.30) in the form
\[ \mathbf{m} = B_1^t * F_{\Omega}^{-1} * \mathbf{c}. \] (6.32)

Starting with the averages we multiply \((m_k)\) with \(B\), i.e., we calculate a linear combination \(S_1(x)\) of the vectors \((u_k)\) from the given mean values
\[ B * \mathbf{m} = B * B_1^t * F_{\Omega}^{-1} * \mathbf{c} \] (6.33)
or equivalently by using the projection onto \(B_\Omega\)
\[ F_{\Omega}(S_1(f)) = F_{\Omega} * B * \mathbf{m} = F_{\Omega} * B * B_1^t * F_{\Omega}^{-1} * \mathbf{c}. \] (6.34)
By applying (6.28) we have following system matrix \(S = S_\Omega\)
\[ S = F_{\Omega} * B * B_1^t * F_{\Omega}^{-1} = F_{\Omega} * B_2 * B_2^t * F_{\Omega}^t \] (6.35)
Then defining \(T = B_2^t * F_{\Omega}^t\) shows that \(S = T^t * T\) is a positive definite matrix and writing \(\mathbf{c}_{\text{given}}\) be the left side of (6.34) we have to solve the linear system
\[ \mathbf{c}_{\text{given}} = S * \mathbf{c}. \] (6.36)
Since \(S\) is a positive definite matrix we are using the fast Conjugate Gradient method described like in [68] for calculating \(\mathbf{c}\) from this equation.
After calculating the Fourier coefficients \(\mathbf{c} = (c_1, c_2, ..., c_R)^t\) we obtain the first approximation of the original by zero-padding, followed by an inverse FFT:
\[ \tilde{\mathbf{c}} = (c_1 | c_2 | ... | c_R | 0 | ... | 0)^t, \quad \tilde{f} = F^{-1} \tilde{\mathbf{c}} \] (6.37)
To iterate this reconstruction the difference in the averages are calculated
\[ \mathbf{m}_{\text{new}} = \mathbf{m}_{\text{given}} - B_0^t * \tilde{f} \] (6.38)
and the procedure starts with these new averages.

Summarizing we can describe the algorithm with three steps:

- **Step 0**: Building up the matrix \(S\):
  \[ S = (B_2^t * F_{\Omega}^t) * (B_2^t * F_{\Omega}^t); \]

- **Step 1**: Calculating \(\mathbf{c}_{\text{given}}\) and solving the linear system to receive \(\mathbf{c}_{\text{new}}\)
  \[ \mathbf{c}_{\text{given}} = F_{\Omega} * B * \mathbf{m} \]
  \[ \mathbf{c}_{\text{given}} = S * \mathbf{c}_{\text{new}} \]

- **Step 2**: Updating averages
  \[ \mathbf{m}_{\text{new}} = \mathbf{m}_{\text{given}} - B_0^t * \tilde{f} \]

- Repeat from Step1
Reconstruction from Averages for Bandlimited Functions

Analyzing the system-matrix $S$:

The following points are important to notice and make the proposed algorithm very effective:

1. Using the Cauchy-Schwarz inequality one shows that $S_\Omega$ is non-expansive i.e. for all $f \in B_\Omega$ one has

$$\|S_\Omega(f)\| \leq \lambda_1 \|f\|,$$

where $\lambda_1 \in (0,1]$ is the maximal eigenvalue of $S_\Omega$. Thus $S_\Omega$ is invertible if and only if

$$\|\nu S_\Omega - Id_{B_\Omega}\| < 1$$

for each $\nu < 2/\lambda_1$, so certainly for any $\nu \in (0,1]$.

If the minimal respectively maximal eigenvalue $\lambda_1, \lambda_2$ of $S_\Omega$ was known we could choose $\nu_0 = 2/(\lambda_1 + \lambda_2)$ as optimal relaxation parameter, for which $\|\nu_0 S_\Omega - Id_{B_\Omega}\| \|_{B_\Omega}$ is minimal, but in practice $\nu_0$ is not known. Figure 6.8 shows the original signal and the approximation using the relaxation parameter $\nu = 2.44$ which has been chosen as compensation for the “loss of mass” (due to the gaps between the indicator functions $(v_i)$).

![Figure 6.8: $f$ and $\nu S_\Omega(f)$, for some $\nu > 2$](image)

2. Furthermore it is important to notice, that the matrix $S$ operates on the relevant coefficients. Therefore $S$ is of size $R \times R$, with $R \leq K \leq N$ ($R$ being in the most cases much smaller than $N$), i.e. that the size of the linear system (6.36) we have to solve depends primarily on the smoothness of the function $f$ than on the length of the signal. The fact that we operate on the coefficients and not on the signal itself makes it possible to reconstruct even large signals.

3. The algorithm uses the technique of the Conjugate Gradient method described like in [68]. The CG-method is one of the most effective technique for solving linear
systems and can be applied within our algorithm because of the fact that the system matrix $S$ is a positive definite matrix. Iterative reconstruction based on the CG strategy is preferable as it leads to fast convergence (indeed, one would have perfect reconstruction after a finite number of steps in the case of exact arithmetic).
Numerical Example:

Figure 6.9.: Original function and the averages.

Figure 6.9 shows the unknown function and the average values. In our example we use for better illustration a short signal with length $N = 128$ and with $K = 32$ indicator functions $(v_i), i = 1, \ldots, 32$. The unknown signal $f$ is a band-limited function with $R = 16$ coefficients non-equal to zero. As mentioned above we know exactly over which regions the averages are taken (dots). Our aim is to reconstruct the unknown function from the average values.

Figure 6.10 shows the first four iterations using the proposed algorithm. As we can see even after four iterations we barely can see any difference between original function and reconstruction.

Figure 6.11 shows the error of the reconstruction after each iteration. We compare three methods:
1. the proposed CG-method
2. using best $L^2$ approximation (projection onto $B_\Omega$)
3. using best $L^2$ approximation with mass loss compensation for the "loss of mass" due the gaps between the indicator functions

Obviously the CG-strategy provides the fastest convergency.

Figure 6.12 shows the system matrix $S$ of the linear equation system $c_{\text{given}} = S \ast c_{\text{new}}$, which has to be solved. As we can see this matrix is diagonal dominated and also positive definite. Furthermore the size of matrix $S$ doesn’t depend on the length of the signal ($N = 128$) but on the number of relevant coefficients ($R = 16$).

The next example demonstrates the performance of the algorithm for the case that there are large gaps between the indicator functions $(v_i)$. 
Chapter 6

Figure 6.10.: Original function (broken line) and the first four iterations steps (solid line).

Figure 6.11.: Logarithmic reconstruction error
Reconstruction from Averages for Bandlimited Functions

Figure 6.12.: positive definite diagonal-dominated system matrix $S$

Figure 6.13.: The original function (dotted line) and the averages

In figure 6.13 we see the original bandlimited function (dotted line) and the averages (dots). To point out the intervals, where the averages are taken, we plotted these in form of the vertical lines. It is worth to notice that there exist large gaps between these intervals, i.e., there exist gaps where no information is available. Even in this situation we obtain a good reconstruction. In the upper plot of figure 6.14 we cannot notice any difference between the original and the reconstructed function. In the lower plot of figure 6.14 we have plotted the absolute difference of the reconstruction versus the original (notice the scaling). Obviously the error in the reconstruction is higher at the positions where no information is available.
Figure 6.14.: Upper plot: The reconstructed function, which looks identical to the original function, and the averages - Lower plot: The error of the reconstruction $|f - f_{rec}|$
6.3. Statistical Accuracy

In [62] Pawlak and Stadtmüller give three motivations for using averages (they call it grouped data) instead of exact samples:

1. In many signal processing problems we are faced with large data sets containing thousands samples per second. Therefore standard reconstruction techniques become very time consuming and elaboration of reconstruction methods with a reduced computational complexity is desirable.

2. The data must be rounded to grid points defined on an equally spaced mesh due to finite precision width which one can observe and record continuous variables.

3. Signals are often sampled imprecisely yielding unwanted jitter error. The data grouping approach can transform this nonuniform sampling problem into a standard regular sampling situation.

In literature relatively little attention has been given to the problem of signal sampling in the presence of random noise. The first rigorous statistical treatment of this problem has been given 1994 by Pawlak/Rafajlowicz [58] and Pawlak/Stadtmüller [59]. In this section we want to present statistical aspects of reconstruction algorithms in the situation of noisy grouped sampling. For the most part we follow the description of Pawlak/Stadtmüller in [62].

First we study the situation of reconstruction from noisy samples and present a reconstruction algorithm with its statistical aspect. Later we generalize this algorithm for reconstruction from grouped data.

For a general statistical methodology on data grouping the reader is referred to [39] and to [22], [61], [37] for some specific applications in statistic.

6.3.1. Signal Recovery from Noisy Data

We assume the observation of equally spaced discrete samples of a signal \( f(t) \) in the presence of random noise, i.e., let be

\[ x_k = k \cdot \delta \]

and the samples \( f(x_k) \) given as

\[ y_k = f(x_k) + Z_k, \quad k = 0, \pm 1, \ldots, \pm n, \quad (6.41) \]

where \( \delta > 0 \) is the sampling period and \( Z_k \) is a noise process assumed to be zero mean and finite variance denoted by \( \sigma^2 \).

The present of random noise forces to consider the recovery of \( f \) in a statistical setting. The measurement of the reconstruction quality is done by the pointwise mean squared error (MSE) or by the global mean integrated squared error (MISE).
For \( f \in B_\Omega \) a straightforward estimate based on Theorem 1 is

\[
f_n(t) = \sum_{|k| \leq n} y_k \text{sinc}\left(\frac{\pi}{\delta}(t - x_k)\right),
\]

where \( \delta \leq \pi/\Omega \).

I.e., we simply substitute the exact data \( f(x_k) \) by the noisy data \( y_k \). It has been shown in [58–60] that one must not do this replacement in order to obtain converging reconstruction algorithms from noisy data. Here we want to give a short statistical explanation for this fact:

Let be

\[
Ef_n = \sum_{|k| \leq n} f(x_k) \text{sinc}\left(\frac{\pi}{\delta}(t - x_k)\right)
\]

then \( f_n(t) \) is an asymptotically unbiased estimate of \( f(t) \) because \( Ef_n(t) \rightarrow f(t) \) as \( n \rightarrow \infty \). Nevertheless it is shown in [62] that

\[
\text{VAR } f_n(t) = \sigma^2 \sum_{|k| \leq n} \text{sinc}^2\left(\frac{\pi}{\delta}(t - x_k)\right) \rightarrow \sigma^2 \text{ as } n \rightarrow \infty.
\]

Having a closer look on the mean squared error we have

\[
MSE(f_n) = E \left( f_n(t) - f(t) \right)^2 \rightarrow \sigma^2 \text{ as } n \rightarrow \infty,
\]

for every \( \delta \leq \pi/\Omega \).

And for the global error we obtain

\[
MISE(f_n) = E \left( \int_{-\infty}^{\infty} \left( f_n(t) - f(t) \right)^2 dt \right) \rightarrow \infty \text{ as } n \rightarrow \infty.
\]

Thus, \( f_n(t) \) cannot be a consistent estimate of \( f(t) \) for any period \( \delta \).

Pawlak and Stadtmüller present in [62] a solution how to handle this problem. They introduce a certain data smoothing:

Let a certain data smoothing: Let

\[
\tilde{f}_n(t) = \delta \sum_{|k| \leq n} y_k \varphi(t - x_k),
\]

with

\[
\varphi(t) = \frac{\sin(\Omega't)}{\pi t},
\]

where (6.47) is a smoothed version of the first estimation (6.42), and \( \Omega' \geq \Omega \).

Furthermore we need some assumptions on the tails of \( f(t) \). To quantify the tail behavior let us use the following assumption:
Assumption 1: Let for \( f \in B_{\Omega} \) we have
\[
|f(t)| \leq c|t|^{-(r+1)}, \text{ for some constant } c > 0 \text{ and } |t| > 0.
\]

Then following theorem describes the statistical behavior of the reconstruction technique (6.47):

**Theorem 15 (Pawlak and Stadtmüller [62]).**

Suppose that Assumption 1 is satisfied with \( 1 \leq r < \infty \). If \( \delta \leq \pi/\Omega \) then for \( n \geq 1 \) and \( \Omega' \geq \Omega \) we have
\[
MISE(\hat{f}_n) \leq \frac{\pi^{2\Omega'/\pi}}{\pi} \delta^2(2n+1) + d_1\delta^5(2n+1)
+ d_2\delta^3(2n+1) + d_3((n+1/2)\delta)^{-2r},
\]

where
\[
d_1 = \|f\|^2\Omega'^5/720, d_2 = 4\|f\|^2\Omega'^3/\pi^2, d_3 = 32\Omega'/r^2.
\]

It is important to notice, that the estimation does not require the knowledge of the bandwidth \( \Omega \). We need only to know that \( \Omega \) is not greater than \( \Omega' \).

The right side of inequality in Theorem 15 indicates, that there is a sampling rate \( \delta \) that is minimizing the error. Hence selecting
\[
\delta^* = n^{-(2r+1)/2(r+1)} \quad (6.49)
\]
we get for the global error
\[
MISE(\hat{f}_n) = O\left(n^{-r/(r+1)}\right). \quad (6.50)
\]

With Theorem 15, the reconstruction scheme (6.47) and the sampling rate (6.49) we have verified the statistical aspect of signal recovering from noisy data. For more details on the case of band-limited noisy signals the reader is referred to [58–60, 62]. Now we want to concentrate on the reconstruction from averages:

### 6.3.2. Signal Recovery from Grouped Data

Let
\[
\{\overline{y}_k, |k| \leq N\} \quad (6.51)
\]
be a sequence of \( N \) group points such that \( \overline{y}_k \) is representing the data falling into the \( k \)-th block. The data are grouped into time intervals of size \( h \). Each such an interval is of the form
\[
(kh - h/2, kh + h/2), \quad k = -N, \ldots, N.
\]


Then replacing \( y_k, \delta, n \) by \( y_k, h, N \) the equivalent formula to the reconstruction algorithm (6.47) becomes

\[
\bar{F}_n(t) = h \sum_{|k| \leq N} y_k \varphi(t - kh) \quad (6.52)
\]

There are various ways of summarizing data falling into a given block. Generally a grouping function can be defined in the following way:

**Definition 6 (Grouping Function).** A **grouping function** \( w \) of order \( p \geq 2 \) is defined if its first \( p - 1 \) moments vanish, i.e.

\[
\int_{-\infty}^{\infty} t^s w(t) dt = \begin{cases} 
1 & s = 0 \\
0 & 0 < s < p \\
\neq 0 & s = p 
\end{cases} \quad (6.53)
\]

Let us assume that the grouping function \( w(t) \) is symmetrical and defined on the interval \((-1/2, 1/2]\) such that \( \int_{-1/2}^{1/2} w(t) dt = 1 \). Some simple standard grouping functions are, e.g.:

- \( w(t) = 1_{(-1/2, 1/2]}(t) \)
- \( w(t) = 2(1 - 2|t|)1_{(-1/2, 1/2]}(t) \)
- \( w(t) = 6(1/4 - t^2)1_{(-1/2, 1/2]}(t) \)
- \( w(t) = 2(3 - 8|t|)1_{(-1/2, 1/2]}(t) \) (negative weights)

With these grouping functions a grouping strategy can be used in the form of weighted average of the data points \( \{y_j, |j| \leq n\} \) falling into the \( k \)-th block

\[
\bar{y}_k = \alpha_k^{-1} \sum_{j \in B_k} w \left( \frac{kh - x_j}{h} \right) y_j, \quad k = -N, \ldots, N \quad (6.54)
\]

where \( B_k = \{j : kn - h/2 < x_j \leq kh + h/2\} \) and \( \alpha_k \) is the number of elements in \( B_k \). Hence, the sample set \( \{y_j, |j| \leq n\} \) is divided into \( 2N + 1 \) blocks with the block size equal \( h \).

As shown in [62] one can obtain better accuracy if instead of (6.54) the following grouping scheme is used

\[
\tilde{y}_k = h^{-1} \sum_{j \in B_k} y_j \int_{(j-1/2)\delta}^{(j+1/2)\delta} w \left( \frac{kh - u}{h} \right) du. \quad (6.55)
\]

These two grouping rules defined in (6.54) and (6.55) combined with the basic reconstruction scheme (6.52) yield the following signal recovery techniques

\[
\bar{F}_n(t) = h \sum_{|k| \leq N} \bar{y}_k \varphi(t - kh) \quad (6.56)
\]
Reconstruction from Averages for Bandlimited Functions

\[ \tilde{f}_n(t) = h \sum_{|k| \leq N} \bar{y}_k \varphi(t - kh) \]  
\hspace{1cm} (6.57)

The statistical accuracy of these two reconstruction techniques is assessed by Pawlak and Stadtmüller in [62]. The reader is referred to this article for the details. Here we want to present following results:

\[ MISE(\hat{f}_n) = O(n^{-(r-1)/4(r+1)}) \]  
\hspace{1cm} (6.58)

\[ MISE(\tilde{f}_n) = O(n^{-r/(r+1)}) \]  
\hspace{1cm} (6.59)

where 6.58 is a slower rate than 6.59.

**Notice:** Comparing the two convergences of (6.50) and (6.59) we see that the convergence of \(MISE(\tilde{f}_n)\) is the same as the one of \(MISE(\hat{f}_n)\), where \(\hat{f}_n\) was the reconstruction from the data without grouping.

This observations plays an important role in the aspect of data compression, because it means that we can choose the compression size yielding a given level of accuracy. The accuracy of the estimates from grouped data is equivalent to that obtained for the reconstruction techniques calculated from the original non grouped data.

Recall the notations:

- \(\hat{f}_n\) is the reconstruction scheme of non-grouped data;
- \(\hat{f}_n\) is the reconstruction scheme of grouped data using grouping strategy (6.54);
- \(\tilde{f}_n\) using the improved grouping strategy (6.55);

Following results are proofed in [62]:

\[ \frac{MISE(\tilde{f}_n)}{MISE(\hat{f}_n)} \to c_1 \text{ as } n \to \infty, \text{ some } c_1 \geq 1 \]  
\hspace{1cm} (6.60)

and

\[ \frac{MISE(\tilde{f}_n)}{MISE(\hat{f}_n)} \to c_2 \text{ as } n \to \infty, \text{ some } c_2 \geq 1 \]  
\hspace{1cm} (6.61)

For each global mean integrated squared error \((MISE)\) the optimal number of blocks \((2N + 1)\) can be calculated, which defines the compression rate. For the technical details of determining the compression rate the reader is referred to [62], where also the explicit compression size for each reconstruction scheme is given.
7. Reconstruction Algorithms for Shift Invariant Spaces

In this chapter we want to give a short introduction for treating non-uniform sampling in shift invariant spaces. Some basic facts about shift invariant spaces are given and after an overview about general techniques of reconstruction we present a new effective algorithm, which takes into consideration the special structure of the reconstruction problem in shift invariant spaces.

7.1. Introduction

Shift-invariant spaces serve as a universal model for uniform and nonuniform sampling of functions. The objective of the so-called sampling problem is either to recover a signal (function) \( f \) from its samples \( \{ f(x_k) : k \in \mathbb{Z} \} \) or to approximate a data set \((x_k, y_k)\) by a suitable function \( f \) satisfying \( f(x_k) \approx y_k \). Obviously this problem is ill-posed, and so a successful reconstruction requires some a priori information about the signal. Usually it is assumed that \( f \) is contained in the span of integer translates of a given generator \( \varphi \).

In technical terms, the original function \( f \) has the form

\[
    f(x) = \sum_{k \in \mathbb{Z}} c_k \varphi(x - k) \tag{7.1}
\]

and belongs to the shift-invariant space \( V(\varphi) \). Introducing \( T_x \) the translation by \( x \) we can write (7.1) as

\[
    f(x) = \sum_{k \in \mathbb{Z}} c_k T_{k \varphi}(x)
\]

Until recently the only choice for \( \varphi \) was the cardinal sinc function \( \varphi(x) = \frac{\sin \pi \alpha x}{\pi \alpha x} \), since in this case \( V(\varphi) \) coincides with the band-limited functions of band-width \( 2\alpha \). Despite the existence of fast numerical methods [29], this model has some drawbacks because it is non-local, and the behavior of \( f \) at a point \( x \) also depends on samples far away from the position \( x \). For this reason, one works with truncated versions of the cardinal sine. This idea leads naturally to work in shift-invariant spaces with a generator \( \varphi \) of compact support.
Chapter 7

The concept of shift-invariant spaces first arose in approximation theory and wavelet theory [16, 18, 47]. Its potential for the systematic treatment of sampling problems was recognized much later. We refer to [6] for a detailed survey of the state-of-art and an extensive list of references.

7.2. Basic Facts about Shift Invariant Spaces

Let us formulate a very general definition for the shift invariant space:

**Definition 7 (Shift Invariant Space).** The shift invariant space generated by the generator \( \varphi \) and \( a > 0 \) is defined as

\[
V^p_a(\varphi) = \left\{ f(\cdot) = \sum_{k \in \mathbb{Z}^d} c_k T_{ak} \varphi(\cdot), \text{ for } (c_k) \in \ell^p(\mathbb{Z}^d) \right\} \subset L^p(\mathbb{R}^d). \tag{7.2}
\]

The important role of the size of the sampling lattice, which is defined by the parameter \( a \) is well described e.g. by Feichtinger and Werther in [30]. In this chapter we will concentrate only on the case \( a = 1 \) and therefore we use \( V^p(\varphi) \) for \( V^p_1(\varphi) \).

If \( \varphi(x) = \frac{\sin \pi x}{\pi x} \), and \( d = 1 \), then \( V^2(\varphi) \) consists exactly of the band-limited functions with spectrum in \([-1/2, 1/2]\), which is also called the Paley-Wiener space \( B_{1/2} \).

To guarantee the stability of these representations, we assume that the generator \( \varphi \) is stable, which means that there exists a constant \( C > 0 \) that

\[
C^{-1} \|c\|_2 \leq \left\| \sum_{k \in \mathbb{Z}} c_k T_{k} \varphi \right\|_2 \leq C \|c\|_2 \tag{7.3}
\]

for all finite sequences \( c = (c_k)_{k \in \mathbb{Z}} \), or equivalently, the translates \( T_{k} \varphi, k \in \mathbb{Z} \), form a Riesz basis for \( V^2(\varphi) \).

As we can see in [54] or [56] (7.3) is equivalent to the following condition on \( \varphi \)

\[
0 < m \leq \hat{a}_\varphi(\xi) = \sum_{j \in \mathbb{Z}} |\hat{\varphi}(\xi + j)|^2 \leq M < \infty, \tag{7.4}
\]

for \( m, M > 0 \).

**Assumption [generator \( \varphi \):** For \( V^p(\varphi) \subseteq L^p(\mathbb{Z}) \) we consider that there exist two constants \( m > 0, M > 0 \) for which

\[
m \|c\|_p \leq \left\| \sum_{k \in \mathbb{Z}} c_k T_{k} \varphi \right\|_p \leq M \|c\|_p \quad \forall c \in \ell^p. \tag{7.5}
\]

Following the description in [5] we see that we need a more stringent condition on \( \varphi \) than (7.3), because we need pointwise evaluation. For this we introduce the definition of Wiener amalgam space:
Definition 8 (Wiener Amalgam Space): For $1 \leq p < \infty$ the amalgam space $W(L^\infty, \ell^p)$ consists of all measurable functions, for which the norm
\[
\|f\|_{W(L^\infty, \ell^p)} = \left( \sum_{k \in \mathbb{Z}} \sup_{x \in [0,1]} |f(x+k)|^p \right)^{1/p} < \infty
\] (7.6)

$W(C, \ell^p)$ defines the subspace of continuous functions in $W(L^\infty, \ell^p)$.

It can be shown that if the generator $\varphi$ belongs to $W(C, \ell^1)$, and if the condition (7.4) is satisfied for $p = 2$, then (7.5) is satisfied for all $p, 1 \leq p \leq \infty$ (for more details see [48]).

Aldroubi and Gröchenig present in [5] a convenient and sufficient condition for the problem of non-uniform sampling in shift invariant spaces to be a well-defined problem:

Theorem 16 (Aldroubi and Gröchenig).
Assume that $\varphi \in W(C, l^1)$.

1. Then $V^p(\varphi) \subset W(C, \ell^p)$ $\forall p, 1 \leq p \leq \infty$.

2. If $X = \{x_j, j \in \mathbb{Z}\}$ is separated, i.e., $\inf_{j,k} |x_j - x_k| > 0$, then
\[
\left( \sum_{j \in \mathbb{Z}} |f(x_j)|^p \right)^{1/p} \leq C\|f\|_p \quad \forall f \in V^p(\varphi).
\] (7.7)

In particular, if $\varphi$ is continuous and has compact support, then the conclusions hold.

Let us recall the definition of a reproducing kernel Hilbert space [77]:

Definition 9 (Reproducing Kernel Hilbert Space). A closed subspace $\mathcal{H} \subset L^2(\mathbb{R})$ of continuous functions on $\mathbb{R}$ is a reproducing kernel Hilbert space (RKHS), if for any $x_0 \in \mathbb{R}$, the pointwise evaluation $f \to f(x_0)$ is a bounded linear functional on $\mathcal{H}$.

Theorem 16 implies for the case $p = 2$ the additional structure of a RKHS. By the Riesz Representation Theorem, there exist $K_{x_0} \in \mathcal{H}$, such that
\[
f(x_0) = \langle f, K_{x_0} \rangle.
\] (7.8)

Corollary 1. If $\varphi \in W(C, l^1)$ and the assumption (7.5) holds then $V^2(\varphi)$ is a reproducing kernel Hilbert space.

Writing the kernel for $V^2(\varphi)$ more explicitly we express it by the means of a dual basis $\{T_k \tilde{\varphi}, k \in \mathbb{Z}\}$ of the Riesz basis $\{\varphi(\cdot - k) : k \in \mathbb{Z}\}$ of $V^2(\varphi)$, where the dual generator $\tilde{\varphi} \in V^2(\varphi)$ must satisfy
\[
\langle \tilde{\varphi}(\cdot), \varphi(\cdot - k) \rangle = \delta_{k0} \quad k \in \mathbb{Z}.
\] (7.9)
Chapter 7

Following the argumentation in [5] \( \tilde{\varphi} \in V^2(\varphi) \) has the Fourier transform

\[
\tilde{\varphi}(\xi) = \hat{\alpha}_\varphi(\xi) \tilde{\varphi}
\]

for some \( 2\pi \)-periodic function \( \hat{\alpha}_\varphi \in L^2(0, 1) \). (7.9) implies that

\[
\langle \tilde{\varphi}(.), \varphi(. - k) \rangle = \int_\mathbb{R} \tilde{\varphi}(\xi) \bar{\varphi}(\xi) e^{2\pi ik\xi} d\xi = \int_0^1 \sum_{k \in \mathbb{Z}} \hat{\alpha}_\varphi(\xi) |\tilde{\varphi}(\xi + k)|^2 e^{2\pi ik\xi} d\xi = \int_0^1 \hat{\alpha}_\varphi(\xi) \bar{\alpha}_\varphi(\xi) e^{2\pi ik\xi} d\xi = \delta_{k0}
\]

Therefore it follows that

\[
\hat{\alpha}_\varphi = 1/\hat{\alpha}_\varphi \in L^2(0, 1).
\] (7.11)

If \( \alpha_k \) are the Fourier coefficients of \( \hat{\alpha}_\varphi \), then the generator \( \tilde{\varphi} \) of the dual basis is given by

\[
\tilde{\varphi}(\cdot) = \sum_{k \in \mathbb{Z}} \alpha_k T_k \varphi(\cdot).
\] (7.12)

Now we can express the reproducing kernel explicitly as

\[
K_{\varphi_0}(y) = \sum_{k \in \mathbb{Z}} \overline{\varphi(\varphi_0 - k)} \tilde{\varphi}(y - k).
\] (7.13)

Since \( \varphi \in W(C, l^1) \), the sum over \( k \) converges absolutely and

\[
K_{\varphi_0}(\cdot) = \sum_{k \in \mathbb{Z}} \overline{\varphi(\varphi_0 - k)} \tilde{\varphi}(\cdot - k) \in V^2(\varphi).
\] (7.14)

How we can use this for reconstruction will be shown in the next section.

Due to symmetric role of the generator \( \varphi \) and the dual generator \( \tilde{\varphi} \) it is of great interest to investigate the question to which extent good properties of \( \varphi \) are automatically shared by \( \tilde{\varphi} \). Therefore we present here some statements from [30] and [25]:

**Theorem 17.** Let \( (B, \| \cdot \|_B) \) be some isometrically translation invariant Banach space, continuously embedded into \( W(L^2, \ell^1) \). Assume that \( \varphi \in B \) generates a Riesz basis for its closed linear span in \( L^2(\mathbb{R}^d) \). Then the dual atom \( \tilde{\varphi} \) also belongs to the same space \( B \).

**Note:** In this theorem we are using atom instead of generator, because the atom \( \varphi \) generating a Riesz basis in \( L^2 \) for the spline-type space \( V(\varphi) \) is, in general, not of compact support and convergence of sums of the form \( \sum_{k \in \mathbb{Z}^d} c_k T_k \tilde{\varphi} \) in \( W(C_0, \ell^2) \), for coefficients \( c \in \ell(\mathbb{Z}^d) \) have to be controlled.
Assume that \((T_k \varphi)_{k \in \mathbb{Z}^d}\) with \(\varphi \in W(C^{(k)}, \ell^1)\) is a Riesz basis for \(V(\varphi)\) in \(L^2(\mathbb{R}^d)\). Then due to the properties of a Riesz basis \(f \in V(\varphi)\) if and only if

\[
f = \sum_{k \in \mathbb{Z}^d} c_k T_k \varphi = \left( \sum_{k \in \mathbb{Z}^d} c_k \delta_k \right) \ast \varphi
\]

for the sequence \(c = (c_k)_{k \in \mathbb{Z}^d} = (\langle f, T_k \varphi \rangle)_{k \in \mathbb{Z}^d} \in \ell^2(\mathbb{Z}^d)\) and the \(\ell^2\)-norm of the coefficient sequence \(c\) is equivalent to the \(L^2\)-norm of \(f\).

In [30] we find following convolution and embedding relations for \(0 \leq l \leq k\):

\[
W(M, \ell^2) \ast W(C^{(k)}, \ell^1) \subseteq W(C^{(l)}, \ell^2) \hookrightarrow W(C^{(l)}, \ell^2) \hookrightarrow L^2(\mathbb{R}^d).
\]

Then it follows that the convolution on the right-hand side of (7.15) is convergent in the \(W(C^{(k)}, \ell^2)\)-sense.

Let \(\mu = \sum_{k \in \mathbb{Z}^d} c_k \delta_k\) and \(f = \mu \ast \varphi\). Then the last relation in (7.16) also implies the estimate

\[
\|f\|_{W(C^{(l)}, \ell^2)} = \|\mu \ast \varphi\|_{W(C^{(l)}, \ell^2)} \leq C(l) \|\mu\|_{W(M, \ell^2)} \|\varphi\|_{W(C^{(l)}, \ell^1)} = C(l, \varphi) \|c\|_{\ell^2}, \quad 0 \leq l \leq k.
\]

With this estimation Feichtinger and Werther proof in [30] following theorem, which points out the possibility of combining sampling of derivatives with amalgam techniques:

**Theorem 18 (Feichtinger, Werther).** Assume that \((T_k \varphi)_{k \in \mathbb{Z}^d}\) with \(\varphi \in W(C^{(k)}, \ell^1)\), is a Riesz basis for \(V(\varphi)\) in \(L^2(\mathbb{R}^d)\). Then, on \(V(\varphi)\), the norms of \(L^2(\mathbb{R}^d)\) and \(W(C^{(l)}, \ell^2)\) are equivalent, for any \(l \in \{0, 1, 2, \ldots, k\}\). With other words, \(V(\varphi)\) is continuously and closed in \(W(C^{(l)}, \ell^2)\) for any \(l \in \{0, 1, 2, \ldots, k\}\).
for the coefficients \( (c_k) \). The relations between the coefficient sequence \( c \) and the samples is given by

\[
Uc = F,
\]

(7.18)

where \( U \) is the infinite matrix

\[
U_{jk} = \varphi(x_j - k),
\]

(7.19)

and \( F \) is the sequence \( f(x_j), j \in \mathbb{Z} \).

With these notations Aldroubi and Gröchenig present in [5] following different terminologies that are used in the context of sampling theory (see also [1], [21], [52]).

**Theorem 19 (Aldroubi and Gröchenig [5]).**

The following are equivalent:

1. \( X = \{x_j, j \in \mathbb{Z}\} \) is a set of sampling for \( V^p(\varphi) \), i.e., \( X \) is a sampling set for which the reconstruction is stable.

2. The matrix \( U \) in (7.19) defines a bounded coercive operator on \( \ell^p \), i.e. there exists \( a, b > 0 \) such that

\[
a \|c\|_p \leq \|Uc\|_p \leq b \|c\|_p
\]

for all \( c \in \ell^p \).

3. There exist two positive constants \( A > 0 \) and \( B > 0 \), such that

\[
A\|f\|_p \leq \left(\sum_{i \in \mathbb{Z}} |f(x_i)|^p\right)^{1/p} \leq B\|f\|_p
\]

for all \( f \in V^p \).

4. The set of reproducing kernels \( \{K_{x_j}, j \in \mathbb{Z}\} \) is a frame for \( V^2(\varphi) \).

**Proof.** (2) \( \iff \) (3): This immediately follows from (7) and (7.3), and the fact that

\[
\sum_{j \in \mathbb{Z}} |f(x_j)|^p = \|Uc\|_p^p.
\]

For \( p = 2 \), (3) \( \iff \) (4): Since \( f(x_j) = \langle f, K_{x_j} \rangle \) for \( f \in V^2(\varphi) \), the sampling inequality

\[
c_2\|f\|_2 \leq (\sum_{i \in \mathbb{Z}} |f(x_i)|^2)^{1/2} \leq C_2\|f\|_2
\]

is equivalent to saying that the kernels \( K_{x_j}, j \in \mathbb{Z} \), form a frame for \( V^2(\varphi) \).

\[\square\]

Now we make following important restriction on the generator \( \varphi \): Let \( \varphi \) be a continuous function with compact support of size \( S \) so that

\[
\text{supp} \ \varphi \subset [-S, S]. \tag{7.20}
\]

For convenience we assume that \( S \) is a positive integer and thus \( \varphi(\pm S) = 0 \).

Basically Theorem 19 leads to some reconstruction techniques that are always applicable:
7.3.1. Linear Algebra Solution

One could simply try to solve the (infinite) system of linear equations

$$\sum_{k \in \mathbb{Z}} c_k \varphi(x_j - k) = f(x_j) \quad \forall j \in \mathbb{Z},$$  \hspace{1cm} (7.21)

for the coefficients $(c_k)$, or in the notation of (7.19) with $f|_X = (f(x_j))_{j \in \mathbb{Z}}$

$$Uc = f|_X.$$  \hspace{1cm} (7.22)

7.3.2. The Normal Equations

Frequently it is better to consider the associated system of normal equations [31]

$$U^* U c = U^* f|_X.$$  \hspace{1cm} (7.23)

This approach has the advantage that the matrix $T := U^* U$ is a positive operator on $\ell^2(\mathbb{Z})$. Furthermore, if the input $y = (y_j)_{j \in \mathbb{Z}}$ does not consist of a sequence of exact samples of $f \in V(\varphi)$, then the function $f = \sum_{k \in \mathbb{Z}} c_k \varphi(\cdot - k)$ corresponding to the solution $c = (U^* U)^{-1} U^* y$ solves the least squares problem

$$\sum_{j \in \mathbb{Z}} |y_j - f(x_j)|^2 = \min_{h \in V(\varphi)} \sum_{j \in \mathbb{Z}} |y_j - h(x_j)|^2.$$  \hspace{1cm} (7.24)

7.3.3. Frame Approach

Theorem 19.4 suggests to use versions of the frame algorithm to find a reconstruction of $f$. By Theorem 19.4 the frame operator which is defined as

$$Sf(x) = \sum_{j \in \mathbb{Z}} \langle f, K_{x_j} \rangle K_{x_j}(x) = \sum_{j \in \mathbb{Z}} f(x_j) K_{x_j}(x)$$  \hspace{1cm} (7.25)

is invertible and its inverse defines the dual frame $\overline{K}_{x_j} = S^{-1} K_{x_j}, j \in \mathbb{Z}$. Then the reconstruction is given by

$$f(x) = \sum_{j \in \mathbb{J}} \langle f, K_{x_j} \rangle \overline{K}_{x_j} = \sum_{j \in \mathbb{J}} f(x_j) \overline{K}_{x_j}(x).$$  \hspace{1cm} (7.26)

We observe that the linear algebra solution (7.22) and the frame method are equivalent. By definition the vector of samples is given by $Uc = f|_X$. The sampled energy of $f \in V(\varphi)$ is

$$\sum_{j \in \mathbb{Z}} |f(x_j)|^2 = \langle f|_X, f|_X \rangle_{\ell^2} = \langle Uc, Uc \rangle_{\ell^2} = \langle U^* Uc, c \rangle_{\ell^2}.$$  \hspace{1cm} (7.27)

Thus $X$ is a set of sampling if and only if $U^* U$ is invertible on $\ell^2(\mathbb{Z})$. 


Chapter 7

In [70] Sun and Zhou show that every function in a spline subspace is uniquely determined and can be reconstructed by its local averages near certain points.

Let be the Zak transform defined by

**Definition 10 (Zak Transform).**

\[ Zf(x, \omega) = \sum_{k \in \mathbb{Z}} f(x + k) e^{-ik\omega}. \]  

(7.28)

Let be \( \varphi_N \) a B-spline of order \( N - 1 \). Then the main result of [70] is:

**Theorem 20 (Sun and Zhou [70], Theorem 2.2).**

Let \( \hat{S}(\omega) = \frac{\hat{\varphi}_N(\omega)}{Z\varphi_{N+1}(N/2, \varphi)} \). Then \( \{S(\cdot - k)\} \) is a Riesz basis for \( V_{\varphi_N} \) and for any \( f \in V_{\varphi_N} \),

\[ f(x) = \sum_{k \in \mathbb{Z}} \langle f, u(\cdot - k) \rangle S(x - k), \]  

(7.29)

where \( u(x) = \chi_{[N/2-1, N/2]}(x) \) and the convergence is both in \( L^2(\mathbb{R}) \) and uniform on \( \mathbb{R} \).

7.3.4. Iterative Frame Methods

In nonuniform sampling problems it is usually difficult to calculate the entire dual frame, therefore one often resorts to iterative methods. Since the Richardson-Landweber iteration in the original paper of Duffin-Schaeffer [21] is slow and requires good estimates of the frame bounds, we recommend the conjugate gradient acceleration of the frame algorithm for all problems without additional structure [33]. It converges optimally and does not require the estimate of auxiliary parameters.

7.4. Exploiting the Structure of the Problem

So far we have discussed general purpose methods for the reconstruction of \( f \in V^p(\varphi) \). These could be applied in any situation involving frames and do not take into consideration the particular structure of the sampling problem in shift-invariant spaces. More precisely we exploit the special structure that appears when the generator of the shift-invariant space has compact support.

Following the description of this section yields to an efficient fast algorithm for the reconstruction problem in shift-invariant spaces (first presented by Gröchenig and Schwab [35]). We describe this algorithm in detail, because we will extend it in the next chapter for the case of averaging sampling.

We make following assumptions:
1. Let be \( p = 2 \), i.e., \( V(\varphi) = V^2(\varphi) \), defined as in (7).

2. For the generator of \( V(\varphi) \) let be
   \[
   \text{supp } \varphi \subseteq [-S, S].
   \]  
   \hspace{1cm} (7.30)

3. We want to reconstruct a function \( f \in V(\varphi) \) from a finite number of samples \( f(x_k) \) taken from an interval \([M_0, M_1]\).

4. Let be the sampling set \( X = \{x_j\}_{j=1}^J \) separable with
   \[
   J \geq M_1 - M_0 + 2S - 1
   \]  
   \hspace{1cm} (7.31)
   and \( \delta \) defined as in (5.4) holds
   \[
   \delta < 1.
   \]  
   \hspace{1cm} (7.32)

### 7.4.1. A Localization Property

The following lemma is simple, but crucial. It is a consequence of the assumption that the generator of \( V(\varphi) \) has compact support.

**Lemma 4.** If \( \text{supp } \varphi \subseteq [-S, S] \), then \( T = U^*U \) is a band matrix of (upper and lower) band-width \( 2S \).

**Proof.** By definition the entries of \( U^*U \) are
   \[
   (U^*U)_{kl} = \sum_{j \in \mathbb{Z}} U_{jk} U_{jl} = \sum_{j \in \mathbb{Z}} \varphi(x_j - k) \varphi(x_j - l).
   \]  
   \hspace{1cm} (7.33)

Since \( \varphi \) has compact support, the sum is always locally finite and its convergence does not pose any problem. Since \( \varphi(x_j - k) = 0 \) if \( |x_j - k| \geq S \), we find that \( (U^*U)_{kl} \) can be non-zero only if both \( |x_j - k| < S \) and \( |x_j - l| < S \). In other words, \( (U^*U)_{kl} \neq 0 \) implies that
   \[
   |k - l| \leq |k - x_j| + |x_j - l| < 2S.
   \]  
   \hspace{1cm} (7.34)

This means that only \( 4S - 1 \) diagonals of \( U^*U \) contain non-zero entries. \( \square \)

**Remarks:** 1. Banded matrices and the resulting numerical advantages occur in a number of related problems. For instance, in the interpolation of scattered data by radial functions with compact support, the interpolation matrix is banded, see [11] and the references there. Likewise, the calculation of the optimal smoothing spline on a finite set of arbitrary nodes requires the inversion of a banded matrix [43].
Chapter 7

2. Lemma 4 combined with a result of Demko, Moss, and Smith [19] or of Jaffard [44] implies that the inverse matrix possesses exponential decay off the diagonal, i.e., there exist $C, A > 0$ such that

$$|(U^*U)^{-1}|_{kl} \leq Ce^{-A|kl|} \quad \forall k, l \in \mathbb{Z}. \quad (7.35)$$

To make our treatment more realistic, we take into account that in any real problem only a finite (arbitrary great) number of samples is given. It turns out that the model of shift-invariant spaces with compactly supported generator possesses excellent localization properties. These are quantified in the next lemma.

**Lemma 5.** The restriction of $f \in V(\varphi)$ to the interval $[M_0, M_1]$ is determined completely by the coefficients $c_k$ for $k \in (M_0 - S, M_1 + S) \cap \mathbb{Z}$.

**Proof.** Since $\varphi(x - k) = 0$ for $|x - k| \geq S$ and $x \in [M_0, M_1]$, we obtain that

$$M_0 - S \leq x - S < k < x + S \leq M_1 + S. \quad (7.36)$$

Consequently, as $S \in \mathbb{N}$, we have

$$f(x) = \sum_{k \in \mathbb{Z}} c_k \varphi(x - k) = \sum_{|x - k| < S} c_k \varphi(x - k) = \sum_{k = M_0 - S + 1}^{M_1 + S - 1} c_k \varphi(x - k).$$

In other words, the exact reconstruction of $f \in V(\varphi)$ on $[M_0, M_1]$ requires only the $M_1 - M_0 + 2S - 1$ unknown coefficients $c_k$ with $k \in (M_0 - S, M_1 + S) \cap \mathbb{Z}$. By counting dimensions, we find that we need at least $M_1 - M_0 + 2S - 1$ samples in $[M_0, M_1]$ for the coefficients to be determined uniquely. Usually the length $M_1 - M_0$ is large compared to $S$, therefore the additional $2S - 1$ coefficients amount to a negligible oversampling.

Lemma 5 demonstrates an important theoretical and practical advantage of shift-invariant spaces with compactly supported generators. A function $f \in V(\varphi)$ can be reconstructed exactly on an arbitrary interval solely from samples in that interval. In contrast, the restriction of a band-limited function to an interval is not uniquely determined by any finite number of samples in that interval, but can only be approximated by these samples. The localization property expressed in lemma 5 is one of the main reasons to work with shift-invariant spaces with compactly supported generators as a sampling model!

Finally we remark that uniform sampling at critical density is not local and may even be unstable in this model. If $f \in V(\varphi)$ is sampled at $\xi + k, k \in \mathbb{Z}$, for some $\xi \in [0, 1)$, then there exists an interpolating function $\psi_\xi$ of exponential decay, such that

$$f(x) = \sum_{k \in \mathbb{Z}} f(\xi + k) \psi_\xi(x - k) \quad [45].$$

In this case the restriction of $f$ to $[M_0, M_1]$ is not
determined exclusively by the $M_1 - M_0$ values $f(\xi + k)$ for $\xi + k \in [M_0, M_1]$. Moreover, if $\varphi$ is continuous, then there always exists a $\xi \in [0, 1)$ such that the reconstruction $\{f(\xi + k)\} \rightarrow f$ is unstable. Janssen’s results in [45] indicate that a small amount of oversampling is an essential hypothesis to guarantee the locality and the stability of its reconstruction.

### 7.4.2 A Local Reconstruction Algorithm

In practice we perform the calculations with a truncated version of the matrices $U$ and $T$. We now combine lemmas 4 and 5 to a first version of an efficient numerical reconstruction algorithm.

**ALGORITHM**

**Input.** We assume that finitely many sampling points $x_1, \ldots, x_J \in [M_0, M_1]$ are given with associated sampling vector $y = (y_1, \ldots, y_J) \in \mathbb{R}^J$. Assume that $J \geq M_1 - M_0 + 2S - 1$ and that the truncated matrix $\mathcal{T}$ defined below is invertible.

**Step 0.** First we define and compute the truncated matrices $\mathcal{U} = \mathcal{U}^{M_0, M_1}$ and $\mathcal{T} = \mathcal{T}^{M_0, M_1} = \mathcal{U}^* \mathcal{U}$, given by their entries

$$\mathcal{U}_{jk} = \varphi(x_j - k)$$

$$\mathcal{T}_{kl} = \sum_{j=1}^{J} \varphi(x_j - k) \varphi(x_j - l)$$

for $j = 1, \ldots, J$ and $k, l = M_0 - S + 1, \ldots, M_1 + S - 1$.

**Step 1.** Compute $b = \mathcal{U}^* y$, i.e.,

$$b_k = \sum_{j=1}^{J} \varphi(x_j - k) y_j \quad \text{for } k = M_0 - S + 1, \ldots, M_1 + S - 1.$$  

**Step 2.** Solve the system of equations

$$c = \mathcal{T}^{-1} b.$$  

**Step 3.** Compute the restriction of $f$ to $[M_0, M_1]$ by

$$f(x) = \sum_{k=M_0-S+1}^{M_1+S-1} c_k \varphi(x - k) \quad \text{for } x \in [M_0, M_1].$$

Then $f$ is the (unique) least square approximation of the given data vector $y$ in the sense that

$$\sum_{j=1}^{J} |y_j - f(x_j)|^2 = \min_{h \in V(\varphi)} \sum_{j=1}^{J} |y_j - h(x_j)|^2.$$  

If $y$ arises as the sampled vector of an $f \in V(\varphi)$, i.e., $y_j = f(x_j)$, then this algorithm provides the exact reconstruction of $f$.  

65
Proof. The least square property (7.41) is clear, since this is exactly the property of the solution of the system of normal equations $\mathcal{U}^*\mathcal{U}c = \mathcal{U}^*y$. See [31] for details.

In the case of $B$-splines a sufficient condition on the sampling density can be extracted from the proofs of Theorem 2.1 and 2.2 of [5]. Assume that $x_{j+1} - x_j \leq \delta$ and that

$$
\delta \leq \frac{M_1 - M_0}{M_1 - M_0 + 2S - 1} < 1.
$$

(7.42)

Then $\mathcal{T}$ is invertible. Condition (7.42) guarantees that there are at least $M_1 - M_0 + 2S - 1$ samples in $[M_0, M_1]$. Then the Schoenberg-Whitney Theorem [64, p. 167] implies that $\mathcal{T}$ is invertible. See [5] for the detailed arguments.

Data Segmentation

A further optimization of the reconstruction procedure is possible by data segmentation. Instead of solving the large system of equations

$$
\mathcal{T}^{M_0,M_1} = (\mathcal{U}^{M_0,M_1})^*y,
$$

(7.43)

with a band matrix of dimension $M_1 - M_0 + 2S - 1$, we will solve $t$ systems of smaller size. For this purpose we partition the large interval $[M_0, M_1]$ into $t$ smaller intervals $[m_r, m_{r+1}]$, $r = 0, \ldots, t - 1$ with $M_0 = m_0$ and $M_1 = m_t$.

Now we apply Algorithm 7.4.2 to each interval separately. More precisely, given the data $(x_j, y_j)$ where $x_j \in [m_r, m_{r+1}]$, we set up the matrices $\mathcal{U}^{m_r,m_{r+1}}$ and $\mathcal{T}^{m_r,m_{r+1}}$ and solve $t$ equations

$$
\mathcal{T}^{m_r,m_{r+1}}c^{(r)} = (\mathcal{U}^{m_r,m_{r+1}})^*y^{(r)}
$$

(7.44)

where the vector $y^{(r)}$ consists of those data $y_j$ for which $x_j \in [m_r, m_{r+1}]$ and the coefficient vector $c^{(r)} = (c_{m_r-S+1}, \ldots, c_{m_{r+1}+S-1})$.

The segmentation technique has a number of practical advantages:

1. The dimension of vectors and matrices can be reduced drastically. Using data segmentation, we solve $t$ small systems of size $(M_1 - M_0)/t + 2S - 1$ instead of the large system of size $M_1 - M_0 + 2S - 1$.

2. Parallel processing can be applied because non-adjacent intervals can be handled simultaneously.

3. The function can be reconstructed on specified subintervals at smaller cost. See figure 7.6.

On the other hand, data segmentation also comes with some caveats:

1. The coefficients $c_k$ with indices $k \in [m_r - S + 1, m_{r+1} + S - 1]$ are computed at least twice because of overlap. Heuristically it has proved best to take averages of the multiply computed coefficients.
2. For a successful execution of the segmentation method it is necessary that each of the small matrices $\mathcal{T}^{m_r,m_{r+1}}$ is invertible. Again by dimension counts we find that the number of data in the interval $[m_r,m_{r+1}]$ should exceed the number of variables, i.e.,

$$\# \left( X \cap [m_r,m_{r+1}] \right) \geq m_{r+1} - m_r + 2S - 1.$$  \hspace{1cm} (7.45)

Obviously this condition imposes an upper bound for the possible number of segmentations.

**Implementation Issues**

1. In Algorithm 7.4.2 the most expensive step is the calculation of the matrix $U$ because it requires the point evaluations of $\varphi$. However, if the sampling points $x_j$ are given, then $U$ and $\mathcal{T}$ can be computed in advance and stored. Thus Step 0 can be taken care of before solving the reconstruction problem.

   We handle the pointwise evaluation of $\varphi$ by “quantizing” the generator. This means that for $\delta > 0$ sufficiently small we create of vector $\psi$ consisting of entries $\varphi(\frac{l}{N})$ for $l = -NS, \ldots, NS$ such that

$$|\varphi(x) - \varphi(\frac{l}{N})| < \delta \quad \text{for} \quad |x - \frac{l}{N}| < \frac{1}{2N}.$$  \hspace{1cm} (7.46)

   Thus to build the matrix $U_{jk} = \varphi(x_j - k)$ amount to selecting the appropriate entries of $\psi$. This approximation of $U$ works remarkably well and fast in the numerical simulations.

2. For the solution of the banded system (7.39) a number of fast algorithms is available. Golub-van Loan [31, Ch. 4.3] offer several efficient algorithms for this task; other options for the inversion of a banded matrix are mentioned in [43]. Since $\mathcal{T}$ is assumed to be positive definite, the band Cholesky algorithm seems to be a good choice that minimizes the operation count for Step 2. MATLAB provides the commands SPARSE and CHOL to deal with this task.

3. Usually $f$ is reconstructed on a grid $G = \{\frac{l}{N} : l = M_0N, \ldots, M_1N\}$. Then (7.40) amounts to a discrete convolution, and thus Step 3 can be performed quickly. Again, since $\varphi$ has compact support, we can use the banded structure of the associated matrix to perform this step.

**Operation Count**

We estimate the number of multiplications for Algorithm 7.4.2. Recall that $J$ is the number of samples and $D = M_1 - M_0 + 2S - 1$ is the dimension of the problem.

(a) According to (7.38) each of the $D$ entries of the vector $b$ requires $\# \{j : |x_j - k| < S \}$
multiplications. Consequently Step 1 requires
\[ \sum_{k=M_0-S+1}^{M_1+S-1} \sum_{j=1}^{J} \chi(k-S,k+S)(x_j) \]
\[ = \sum_{j=1}^{J} \sum_{k=M_0-S+1}^{M_1+S-1} \chi(k-S,k+S)(x_j) \]
\[ \leq \sum_{j=1}^{J} 2S = 2SJ \]
operations, because a point \( x \) is in at most \( 2S \) translates of the open interval \((-S, S)\).

(b) Likewise to calculate an entry of \( T \) requires
\[ \# \left( \{ j : |x_j - k| < S \} \cap \{ j : |x_j - l| < S \} \right) \] (7.47)
multiplications, see (7.37). As in (a) we estimate the number of operations to set up the matrix \( T \) by
\[ \sum_{k=M_0-S+1}^{M_1+S-1} \sum_{l=M_0-S+1}^{M_1+S-1} \# \left( \{ j : |x_j - k| < S \} \cap \{ j : |x_j - l| < S \} \right) \]
\[ = \sum_{j=1}^{J} \left( \sum_{k=M_0-S+1}^{M_1+S-1} \chi(k-S,k+S)(x_j) \right) \left( \sum_{l=M_0-S+1}^{M_1+S-1} \chi(l-S,l+S)(x_j) \right) \]
\[ \leq J \cdot (2S) \cdot 2S \] (7.48)
operations (and no square roots), see [31, Ch. 4.3.6].

(c) For the solution of the banded system \( Tc = b \) by means of the band Cholesky algorithm we need at most
\[ D((2S)^2 + 16S + 1) = (M_1 - M_0 + 2S - 1)((2S)^2 + 16S + 1) \leq J(4S^2 + 16S + 1) \]
operations and no square roots), see [31, Ch. 4.3.6].

(d) To compute the reconstruction \( f \) on a grid \{ \frac{L}{N} : l = M_0N, \ldots, M_1N \} we need to calculate \((M_1 - M_0)N\) point evaluations of \( f \) via (7.40). Since \# \{ \( k \in \mathbb{Z} : |x - k| < S \} \leq 2S \), each point evaluation requires at most \( 2S \) multiplications. Thus for the reconstruction on the grid we need at most
\[ (M_1 - M_0)N \cdot 2S \leq J \cdot 2SN \] (7.49)
multiplications.

Combining these estimates, we find that Algorithm 7.4.2 requires
\[ O(J(S^2 + SN)) \] (7.50)
operations. In other words, the cost of the algorithm is linear in the number of data and quadratic in the size of the generator!
Numerical Simulations

In our simulation we have used MATLAB. We used the shift-invariant spline spaces with the $B$-spline of order 3

$$\varphi = \chi_{[-1/2,1/2]} \ast \cdots \ast \chi_{[-1/2,1/2]}$$

4 times as the generator of $V(\varphi)$. Thus $\text{supp} \, \varphi \subseteq [-2,2]$ and $S = 2$. See figure 7.1.

![Figure 7.1: Generator $\varphi$: a B-spline of order 3 with $\text{supp} \, (\varphi) \subseteq [-2,2]$](image)

Figure 7.2 is a plot of the operation count as a function of the number of sampling points. We have reconstructed examples of size 114, 226, 444, 667, 887, 1085 and used the MATLAB function FLOPS to count the number of operations.

![Figure 7.2: Number of multiplications for reconstruction problems of different size. The operation count is linear in the number of samples.](image)
Chapter 7

The example in figure 7.3 uses a signal on the interval \([0, 128]\). Since \(S = 2\), we need at least \(M_1 - M_0 + 2S - 1 = 131\) samples. The actual sampling set of figure 7.3 consists of ca. 200 points and satisfies the maximum gap condition \(\max_j(x_{j+1} - x_j) \approx 0.67 < 1\).

To make the example more realistic, we have added white noise to the sampled values of a given function \(f \in V(\varphi)\). Instead of using the correct values \(f(x_j)\) in the reconstruction algorithm, we use the noisy values \(f_{\text{err}}(x_j) = f(x_j) + e_j\) so that

\[
f_{\text{err}}|X = f|X + e.
\]

The relative error between the original signal and the noisy signal is measured by

\[
er_{\text{err}}|X - f|X = \left( \frac{\sum_{j=1}^f |e_j|^2}{\sum_{j=1}^f |f(x_j)|^2} \right)^{1/2}.
\]

In our example \(er_{\text{err}}|X = 63.8\%\).

Figure 7.3 shows the plots of the original signal (top left), of the noisy signal (top right), the plot of the noisy samples which looks rather chaotic (bottom left). The last plot (bottom right) displays the reconstruction (continuous line) means of Algorithm 7.4.2. For comparison we have added the original function as a dotted line. The relative error \(er_{\text{rec}}\) of the reconstruction measured at the sampling points with respect to the correct samples is now

\[
er_{\text{rec}}|X = f|X = 18.5\%.
\]

The noise reduction is thus

\[
er_{\text{err}}|X = 63.8\% \rightarrow er_{\text{rec}} = 18.5\%.
\]
Figure 7.3.: Reconstruction with noisy nonuniform samples: The top right plot shows the signal with additive $\text{err}_{\text{samp}} = 63.8\%$ noise. Bottom left shows the noisy signal sampled on a nonuniform grid with maximal gap $\approx 0.67$. Bottom right shows the reconstructed function (continuous line) and original function (dotted line).

In figure 7.4 we investigate the dependence of the reconstruction on the generator $\varphi$. In each subplot the generator is a $B$-spline of order $N$, i.e. $\varphi_N = \chi_{[-1/2,1/2]} * \cdots * \chi_{[-1/2,1/2]}$ ($N + 1$-fold convolution). The data set $(x_j, y_j)_{j=1,\ldots,J}$ is generated by sampling a function $f \in V(\varphi_0)$ and then we have added noise. The top left picture shows the original signal and the noisy sampled data. Then each subplot depicts the optimal approximation of these data in the spline space $V(\varphi_N), N = 0, \ldots, 6$, starting with an approximation by a step function $f_{\text{rec}} \in V(\varphi_0)$ via an approximation by a piecewise linear function $f_{\text{rec}} \in V(\varphi_1)$ and ending with a smooth approximation $f_{\text{rec}} \in V(\varphi_6)$.

In each case we have also plotted the original function $f$ (dotted line) for comparison. In addition, the relative error $\text{err}_{\text{samp}}$ is indicated. The dependency of this error of $N$
is a typical $L$-curve as it occurs in regularization procedures. In all our examples the best approximation is obtained in the correct space in which $f$ was originally generated. This observation is consistent with the extended literature on smoothing splines, e.g., [14,43,73]. The main difference between those methods and the algorithm of Section 7.4.2 is in the underlying function space. The reconstruction algorithm 7.4.2 finds the best local reconstruction in the shift-invariant spline space $V(\varphi)$, whereas the smoothing spline of [14] is based on the nodes $x_j$ and does not belong to a fixed function space.
Iterative Reconstruction Algorithms for Shift Invariant Spaces

Figure 7.4.: Reconstruction of a signal from noisy samples in shift-invariant spaces with B-splines of different orders as generator. Top left: Original Signal (dotted line) and the noisy samples are marked (×). Other plots: Original Signal ... dotted line - Reconstruction ... solid line

Figure 7.5 displays the associated banded matrix $\mathcal{T}$ of the linear system (7.39). White squares correspond to zero entries, dark squares signify large entries of $\mathcal{T}$, the shading being proportional to the size. The banded structure is clearly visible.
Figure 7.5.: Structure of the banded matrix $\mathcal{F}$

Figure 7.6 exhibits the power of the method of data segmentation. Instead of reconstructing the entire signal $f$, we have reconstructed only the restriction to two disjoint intervals. In the absence of noise the reconstruction is exact. Since $\text{supp}(\varphi) \subset [-S, S]$, the calculation for the two intervals can be done locally and simultaneously. This property can be used for parallel processing.

Figure 7.6.: Reconstruction of the function on disjoint intervals (without noise)
7.5. Reconstruction on a Lattice

The preceding section describes the continuous situation, i.e., the reconstructed function \( f \in V(\varphi) \) is evaluated at any arbitrary point \( x \in \mathbb{R} \).

In this section we want to examine the situation when \( f \in V(\varphi) \) is evaluated only on a lattice \( \Lambda \in \mathbb{R} \), with lattice size \( a < 1 \):

\[
\Lambda = \{ x_i : \cdots < x_{-2} < x_{-1} < x_0 < x_1 < \cdots \text{ with } x_{k+1} - x_k = a \}
\]

The great numerical advantage of reconstruction on a lattice is the fact that we can benefit by using the Fast-Fourier transformation:

Let \( \vec{x} \) be the positions where we want to reconstruct the function \( f \), i.e.

\[
\vec{x} = (M_0, M_0 + a, M_0 + 2a, \ldots, M_0 + na = M_1)
\]

and the vector \( \vec{y} \) the desired reconstruction

\[
\vec{y} = f(\vec{x}) = (f(M_0), f(M_0 + a), f(M_0 + 2a), \ldots, f(M_1)).
\]

Let us suppose that we have already calculated the coefficients \( \vec{c} = (c_0, c_1, \ldots, c_{M_1-M_0+2S}) \) of the function \( f \) by the algorithm of chapter 7.4. As described in the section before, we need \( M_1 - M_0 + 2S + 1 \) coefficients for the restriction of \( f \in V(\varphi) \) to the interval \( [M_0, M_1] \).

Then we can define a vector \( \vec{C} \) in the following way:

\[
C(k) = \begin{cases} 
  c(j), & \text{for } j = k \cdot a \in \mathbb{N} \\
  0, & \text{else.}
\end{cases}
\]

for \( k = 0, 1, \ldots, (M_1 - M_0 + 2S)/a \) and \( j = 0, 1, \ldots, M_1 - M_0 + 2S \).

In other words \( \vec{C} \) is the extension of vector \( \vec{c} \) in the sense that at every \( \frac{1}{a} \)th entry of \( \vec{C} \) corresponds to the vector \( \vec{c} \).

With this notations we can rewrite the equation

\[
f(\vec{x}) = \vec{y} = \sum_{k=M_0-S+1}^{M_1+S-1} c_k \varphi(\vec{x} - k)
\]

as

\[
\vec{y} = \vec{C} \otimes \vec{\varphi},
\]

where \( \otimes \) denotes the convolution and can be carried out by pointwise multiplication \( \odot \) in the fourier domain, i.e.,

\[
\vec{y} = \mathcal{F}^{-1} \left( \mathcal{F}(\vec{C}) \otimes \mathcal{F}(\vec{\varphi}) \right).
\]

(7.55)

Now one of the features of the described algorithm in the section before plays an important role for numerical efficiency: the segmentation!

Since we can divide the interval \( [M_0, M_1] \) in segments of arbitrary size we can choose the
segments \([M_{0}^{seg}, M_{1}^{seg}]\) in such a way that the lengths become a power of 2. As a consequence the convolutions are carried out by fast-fourier-transformations, which improves the algorithm significantly.

More precisely, we have to determine the segments in the following way: For reconstructing the function \(f\) within the interval \([M_{0}^{seg}, M_{1}^{seg}]\) we need \(M_{1}^{seg} - M_{0}^{seg} + 2S + 1\) coefficients. To avoid boundary effects of the fourier-transformation we have to extend the vector \(\vec{C}\) by zeros padding to the size

\[
\frac{(M_{1}^{seg} - M_{0}^{seg} + 2S + 1 + 2S)}{a},
\]

because \(\text{supp} (\varphi) = [-S, S]\).

By choosing

\[
M_{1}^{seg} = 2^{p} \cdot a - 4S - 1 + M_{0}^{seg}
\]

the lengths of the vectors in (7.55) are \(2^{p}\) and the convolution is done by fast-fourier-transformation.

**Numerical Example**

Figure 7.7 shows a reconstruction of a function using segmentation, which is marked by vertical lines.

The interval where we want to reconstruct is \([M_{0} = 200, M_{1} = 400]\) and the lattice size is \(a = 0.001\). Therefore 200,000 points have to be calculated.

The algorithm divides the interval into 10 segments where nine are of the size \(2^{15}\).

The reconstruction of the interval \([M_{0}, M_{1}]\) is done within 0.6 seconds. The same reconstruction without segmentation needs 1.2 seconds and a larger amount of memory space. Therefore using segmentation makes the algorithm twice as faster. The effect is even more dramatic if we reduce the size of the lattice to \(a = 0.0001\). The 2,000,000 points are calculated five times faster with segmentation than without.
Figure 7.7.: Reconstruction of the function using segmentation of size $2^{15}$
8. Fast Recovery of Shift Invariant Functions from Local Averages

As mentioned in the chapters before fast iterative schemes for the reconstruction of functions from their samples have been introduced by Feichtinger and Gröchenig for the case of bandlimited functions [27]. Sun and Zhou have generalized some of these results for average-sampling [69–72]. The schemes of [27] have been extended by Aldroubi and Feichtinger to general shift-invariant spaces [3]. In [35] an effective non-iterative reconstruction algorithm for function in shift-invariant spaces has been developed by Gröchenig and Schwab [35] as presented in chapter 7.4. Now we want to focus on fast iterative reconstruction schemes for the case of average sampling presented by Aldroubi in [2]. Finally we will extend the non-iterative algorithm presented in chapter 7.4 by Gröchenig and Schwab to the situation of reconstructing a function of a shift-invariant space from averages.

8.1. Fast Iterative Algorithm

Let us recall some very useful facts about Wiener amalgam spaces, described like in [6]:

1. \( W_0(L^p) = W(C, L^p) \subset W(L^p) \) is also a Banach space [24].

2. Let \( f \in L^p(\mathbb{R}^d) \) and \( g \in W(L^1(\mathbb{R}^d)) \), then \( f * g \in W(L^p(\mathbb{R}^d)) \) and we have
   \[ \|f * g\|_{W(L^p)} \leq C\|f\|_{L^p}\|g\|_{W(L^1)}. \]

3. Let \( c \in \ell^n(\mathbb{Z}^d) \) and \( \varphi \in W(L^1(\mathbb{R}^d)) \), then
   \[ \left\| \sum_{k \in \mathbb{Z}^d} c_k \varphi(\cdot - k) \right\|_{W(L^p)} \leq \|c\|_{\ell^n}\|\varphi\|_{W(L^1)}. \]

4. If \( f \in L^p(\mathbb{R}^d) \) and \( g \in W(L^1(\mathbb{R}^d)) \), then the sequence \( d \) defined by
   \[ d_k = \int_{\mathbb{R}^d} f(x)g(x - k)dx, \quad k \in \mathbb{Z}^d, \] belongs to \( \ell^p(\mathbb{Z}^d) \) and we have
   \[ \|d\|_{\ell^p} \leq \|f\|_{L^p}\|g\|_{W(L^1)} \quad 1 \leq p \leq \infty. \]
For the following algorithm described by Aldroubi in [2] we make the assumption that the averaging function \( u \) belongs to \( W(L^1) \). Furthermore we need the definition of a bounded partition of unit:

**Definition 11 (BUPU).** A **bounded partition of unity** (BUPU) associated with \( \{B_\gamma(x_j)\}_{j \in J} \) is a set of functions \( \{\beta_j\}_{j \in J} \) that satisfy

1. \( 0 \leq \beta_j(x) \leq 1, \ \forall j \in J; \)
2. \( \text{supp} \ \beta_j \subset B_\gamma(x_j); \)
3. \( \sum_{j \in J} \beta_j = 1. \)

For the iteration scheme the algorithm uses a quasi-reconstruction operator \( A_{X,a} \), which is defined as follows:

\[
A_{X,a}f = \sum_{j \in J} \langle f(\cdot)u_a(\cdot - x_j) \rangle \beta_j = \sum_{j \in J} (f * u_a^*_j(x_j)) \beta_j, \tag{8.1}
\]

where \( u_a(\cdot) = \frac{1}{a^\alpha}u(\frac{\cdot}{a}) \), and where \( u_a^*_j(x) = u_a(\cdot - x) \).

Obviously the quasi-reconstruction operator \( A_{X,a}f \) does not belong to the space \( V^2(\phi) \), but can be used for the reconstruction algorithm for \( f \in V^2(\phi) \):

**Theorem 21 (Aldroubi [2]).** Let \( \phi \) be in \( W_0(L^1) \), let \( u \) be a compactly supported function in \( L^1 \), and \( P \) be the orthogonal projection from \( L^2 \) onto \( V^2(\phi) \). Then there exists a density \( \gamma > 0 \ (\gamma = \gamma(u, \phi)) \) and \( a_0 > 0 \) such that any \( f \in V^2(\phi) \) can be reconstructed from its samples \( \{f(x_j) : x_j \in X\} \) an any \( \gamma \)-dense set \( X = \{x_j, j \in J\} \) and for any \( 0 < a < a_0 \) by the following algorithm:

\[
\begin{align*}
\{ & f_1 = PA_{X,a}f \\
& f_{n+1} = PA_{X,a}(f - f_n) + f_n \}
\end{align*} \tag{8.2}
\]

Then \( f_n \) converges to \( f \) uniformly and also in the \( W(L^2) \)-norm and the \( L^2 \)-norm. The convergence is geometric, that is

\[
\|f - f_n\|_{L^2} \leq C\|f - f_n\|_{W(L^2)} \leq C_1\|f - f_1\|_{W(L^2)} \alpha^n, \tag{8.3}
\]

for some \( \alpha = \alpha(\gamma, a, \phi, u) < 1 \) and \( C_1 < \infty \).

In the presence of noise the data are given by \( f' = \{f'_j : j \in J\} \), where we assume that \( f' \) belongs to \( \ell^2 \), but not that \( f' \) are samples of a function \( f \in V^2(\phi) \).

For this case the algorithm of Theorem 21 is modified in the following way:

\[
f_1 = PQ_X\{f'_j\} := P\left(\sum_{j \in J} f'_j \beta_j\right), \tag{8.4}
\]
where \( \{\beta_j : j \in J\} \) is the BUPU of definition 11. Furthermore (8.2) becomes

\[
f_{n+1} = f_1 + (I - PA_{X,a})f_n.
\]

(8.5)

With this modifications Aldroubi formulates the following theorem:

Theorem 22 (Aldroubi [2], Theorem 3.2).
Under the same assumptions as in Theorem 21, the algorithm 8.5 converges to a function \( f_\infty \in V^2(\varphi) \) which satisfies

\[
P(A_{X,a}f_\infty - Q_x\{f'_j\}) = 0.
\]

(8.6)

Also the convergence rate \( \alpha \) of (8.3) is presented:

Theorem 23 (Aldroubi [2], Theorem 3.5).
Assume that \( \text{supp} \, \varphi \subset B_{r_1}(0) \) and that \( \text{supp} \, u \subset B_{r_2}(0) \). Then the convergence rate \( \alpha \) in 8.3 satisfies

\[
\alpha \leq \frac{1}{A} [2r_1 + 2^d \|\nabla \varphi\|_{L^\infty} (\gamma + ar_2(6^d + 1)\|u\|_{L^1})], \quad 0 < \gamma \leq 1, 0 < a \leq \frac{1}{r_2}
\]

(8.7)

where \( m \) is the lower bound constant in (7.3) and \( [t] \) denotes the smallest integer bigger than or equal to \( t \).

The relevance of Theorem 23 is that it allows to find the density \( \gamma \) and the value \( a_0 \) needed for the reconstruction algorithm to converge.

### 8.2. Fast Algorithm for Reconstruction of Shift-Invariant Functions from Averages

In this section we want to extend the algorithm from Gröchenig and Schwab described in chapter 7.4 to the case of averaging sampling. The algorithm is based on the technique of "moving average" where one important application is the reconstruction of the density function from a given histogram. The proposed algorithm is compared with different standard algorithm on this topic like Kernel-Density Estimation and also a basic numerical example is presented.

#### 8.2.1. Reconstruction of shift invariant functions

As presented in the chapter 7.4 (see also K.Gröchenig and H.Schwab [35]) we have a fast local reconstruction method for sampling in shift invariant spaces.

Let be \( f \in V(\varphi) \), i.e.,

\[
f(x) = \sum_{k \in \mathbb{Z}} c_k \varphi(x - k),
\]

(8.8)
Chapter 8

which means that the function $f$ is completely determined by the coefficients $c_k$ or, in other words, the problem of finding the exact reconstruction $f$ is equivalent of finding the coefficients $c_k$. In chapter 7.4 it was shown how to find the correct coefficients from a sampling set $\{f(x_j)\}_{j=1}^J$ in an efficient way by using bandlimited matrices.

Since in practice the assumption that the exact samples $\{f(x_j)\}$ is not realistic we want to reconstruct from averages of the form $y_j = \langle f, u_j \rangle$.

In the situation of moving averages, i.e., the averages are always taken with respect to the same average function $u$, we have:

1. $u_j(\cdot) = u(\cdot - x_j)$;
2. $\text{supp } u = [-\frac{\mu}{2}, \frac{\mu}{2}]$;
3. the data that we can use for reconstruction are given by

$$y_j = \langle f, u_j \rangle = (f * u)(x_j),$$

(8.9)

where $*$ denotes the convolution.

Applying the convolution with $u$ onto (8.8), we see

$$(f * u)(x) = \sum_{k \in \mathbb{Z}} c_k(\varphi * u)(x - k),$$

(8.10)

that the function $(f * u)$ is determined by the same coefficients as $f$ and generated by $\varphi * u$, i.e.,

$$(f * u) \in V(\varphi * u).$$

(8.11)

With this knowledge our strategy in reconstruction functions from shift-invariant spaces from averages becomes clear: First we calculate from the given data $y_j = \langle f, u_j \rangle$ using the algorithm of chapter 7.4 the unknown coefficients $\{c_k\}$ and then we reconstruct the desired function $f$ with these coefficients.

To keep the notation clear we define

$$\psi = \varphi * u.$$ (8.12)

Then the averages, which we use as samples for the function $(f * u)$ can be written as

$$y_j = \langle f, u_j \rangle = (f * u)(x_i) = \sum_{k \in \mathbb{Z}} c_k \psi(x_i - k)$$

(8.13)

Let $U$ be the infinite matrix with entries

$$U_{jk} = \psi(x_j - k) \quad j, k \in \mathbb{Z},$$

(8.14)
Fast Recovery of Shift Invariant Functions from Local Averages

Then with \((f * u)|_X = ((f * u)(x_j))_{j=1,...,J}\) the equation (8.13) can be written as

\[ Uc = (f * u)|_X. \quad (8.15) \]

or in the sense of normal equations

\[ U^*Uc = U^*(f * u)|_X. \quad (8.16) \]

Since \(\text{supp} \varphi \subset [-S, S]\) and \(\text{supp} \ u \subset [-\frac{\mu}{2}, \frac{\mu}{2}]\) the support of \(\psi\) is given by

\[ \text{supp} \ \psi \subset [-S - \frac{\mu}{2}, S + \frac{\mu}{2}]. \quad (8.17) \]

Then the reconstruction of \((f * u) \in V(\psi)\) on \([M_0, M_1]\) requires only \(M_1 - M_0 + 2S + \mu - 1\) unknown coefficients \(c_k\) with \(k \in (M_0 - S - \mu, M_1 + S + \mu) \cap \mathbb{Z}\).

After we evaluate the coefficients \(\{c_k\}\) from the function \((f * u)\) we can use them to construct the restriction of \(f\) to \([M_0, M_1]\) by

\[ f(x) = \sum_{k=M_0-S+1}^{M_1+S-1} c_k \varphi(x - k) \quad \text{for} \ x \in [M_0, M_1]. \quad (8.18) \]

8.2.2. Density Estimation for Histograms in Shift Invariant Spaces

We will explain the reconstruction from averages in shift invariant spaces by solving a concrete problem: the reconstruction of a density function from a given histogram. Histograms are among the most widely used methods of data presentation. In many applications we want to vary the width of the bins or the origin of the histogram. This motivates to reconstruct a good approximation of the underlying density function, which creates the histogram. For some concrete applications of this problem see eg. [8], [7], [13], [37] or [61].

Introduction

Histograms can be viewed as a particular example of a density estimate and their appearance depends on both the choice of origin of the histogram and the width of the intervals used. Our goal is to find a good approximation of the underlying density function.

There are many different techniques, which solves this problem: For example the kernel density estimation (KDE) is a well known technique in statistics. The simplest way of KDE can be thought as a smoothed version of histograms. The main problem in practice is to obtain a sufficiently smooth representation of the data while also the retaining its main features. In this context the choice of bin-width is critical.
First we describe the standard technique of KDE and refer to [8] who presents a solution of the under/over smoothing problem.

These KDE techniques require an explicit knowledge of the observations, which perform the histogram. Our goal is to avoid this assumption and to reconstruct the density function directly from the histogram. We can look at this problem as a kind of resolution enhancement, i.e., if a multiple sensors take measurements over disjunct intervals we only get sums, respectively averages over these intervals and our intention is to get a resolution of arbitrarily small intervals.

The critical difference in this approach is that we don’t use the histogram as a presentation of data but as input for the reconstruction of the density function.

**Kernel Density Estimation**

Let \( X_i, \ i = 1, \cdots, N, \) be scalar measurements drawn from an arbitrary probability distribution \( f \). One well known approach (the kernel density estimation KDE, see also [63]) of finding an approximation \( f_A \) for the unknown density function \( f \) is obtained based on a kernel function \( K(u) \) and a bandwidth \( h \) as

\[
    f_A(x) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x - X_i}{h}\right). \tag{8.19}
\]

This method can be viewed as placing a 'bump' at each point and then summing the height of each bump at each point of the x-axis.

Compared to the histogram the shape of \( f_A \) does not depend upon the choice of origin but critical on the choice of bandwidth \( h \). Large values of \( h \) over-smooth, while small values under-smooth the data.

In [8] Baxter and Beardah present a choice of \( h \), which 'optimizes' the kernel density estimation in the sense that the calculated \( h \) makes the KDE as 'close' as possible. The measurement of 'closeness' is the asymptotic mean integrated square error (AMISE) which can be shown to have the form

\[
    AMISE(f_A) = \frac{1}{nh} A + \frac{1}{4} h^4 B. \tag{8.20}
\]
As described in [8] the terms $A$ and $B$ in equation (8.20) are dependent on the known kernel, while $B$ is also dependent on the integral of the squared second derivative of the unknown $f$. Then the optimal value of $h$ which minimizes the AMISE is given by

$$h_{AMISE} = \left[ \frac{A}{nB} \right]^{1/5}.$$  (8.21)

This expression, which through $B$ depends upon the second derivative of the unknown density $f$, is the starting point for many methods for automatic selection of $h$.

Here we will not go into further details of selecting the optimal $h$, which can be found for example in [74] or [8].

Reconstruction with parabolic splines

In [17] Carl de Boor gives a spline interpolation for this problem: Let $f_A$ to be a parabolic spline, i.e., a piecewise polynomial function of order 3 with continuous first derivative,

$$f_A \in \mathbb{P}_{3,\zeta} \cap C^1,$$  (8.22)

where $\mathbb{P}_{3,\zeta}$ is the linear space of piecewise polynomial functions of order 3 with breakpoint sequence $\zeta$, which coincide with the sequence $a_j$, $j = 1, \ldots, J + 1$.

If the underlying density function $f$ is smooth and vanishes outside the interval $[a_1, a_{J+1}]$, then we have $f^{(i)}(a_1) = f^{(i)}(a_{J+1}) = 0$ for $i = 0, 1, \ldots$ to the extent of the smoothness of $f$.

This gives altogether $J$ interpolation conditions and $2J$ homogeneous conditions, for a total of $3J$ conditions on the $3J$ polynomial coefficients. For the solution of the resulting linear system the reader is referred to [17].

8.2.3. Reconstruction of the Density from Histograms with the proposed Algorithm

Now we want to focus on another problem: Following the technique described above an a-priori knowledge of the points $X_i$ is assumed. In that case a histogram is only used as a method of data presentation. Now we want to concentrate on the problem that the histogram is the only input we have, i.e., the histogram does not appear as a data representation but as data itself.

Therefore our goal is to find a good approximation of the density function from the histogram itself without using the data/observations $X_i$.  

85
Chapter 8

As mentioned above the input of our reconstruction algorithm is not the observations \( X_i \) because they might be unknown, but the histogram, i.e., the position, the width and the height of the bins.

**Averages created by one average-function \( u \)**

We can formulate the problem as follows: Let be given \( J \) intervals of same length, i.e., for \( j = 1, \ldots, J \) and fixed \( d > 0 \):

\[
I_j = [a_j, a_{j+1}), \\
a_{j+1} - a_j = d.
\]

Then we can describe a histogram by a step-function \( f_H \)

\[
f_H(x) = \begin{cases} 
  w_j, & \text{for } x \in [a_j, a_{j+1}) \\
  0, & \text{for } x < a_1 \text{ or } x \geq a_{J+1}
\end{cases}
\]

with \( \int f_H(x)dx = 1 \) to get a normalized histogram.

The problem we want to handle is to get an approximation \( f_A \) of the unknown density \( f \) from the given function \( f_H \).

The first step is to define the samples for the reconstruction. Let the average function \( u \) be defined as follows:

\[
u(x) = \begin{cases} 
  1/\mu, & \text{for } x \in [-\frac{\mu}{2}, \frac{\mu}{2}] \\
  0, & \text{otherwise}
\end{cases}
\]

Considering the convolution

\[
(f \ast u)(x) = \int f(\xi)u(\xi - x)d\xi = \int_{x-\mu/2}^{x+\mu/2} f(\xi)u(\xi - x)d\xi
\]

\[
= \int_{x-\mu/2}^{x+\mu/2} f(\xi)\frac{1}{\mu}d\xi
\]

\[
= \frac{1}{\mu} \int_{x-\mu/2}^{x+\mu/2} f(\xi)d\xi,
\]

it is obvious that

\[
(f \ast u)(x_j) = \frac{1}{\mu} \int_{x_j-\mu/2}^{x_j+\mu/2} f(\xi)d\xi
\]

is the average of the function values over the interval \([x_j - \frac{1}{\mu}, x_j + \frac{1}{\mu}]\).

Since the bins of a histogram are normalized by their width \( d \) the following equation holds for \( j = 1, \ldots, J \):

\[
(f \ast u)(x_j) = f_H(x_j)
\]

with

\[
x_j = \frac{a_j + a_{j+1}}{2}
\]
the midpoints of each bin. Therefore we can use the values \( \{ f_H(x_j) \}_{j=1}^J \) as sampling values for the function \( f * u \) and calculate the unknown coefficients of the function \( f * u \). As described in chapter 8.2.1 we can use them to reconstruct an approximation of the density function \( f \).

**Numerical Simulations**

For exact reconstruction the set of samples, which are the midpoints of the bins, has to fulfill certain conditions. If \( \varphi \) is a B-spline of order \( N \), i.e., \( \varphi = \chi_{[0,1]} * \cdots * \chi_{[0,1]} (N + 1 \) convolutions), then the main result of [5] implies that the maximum gap condition

\[
\sup_{i \in \mathbb{Z}}(x_{i+1} - x_i) = \delta < 1
\]

(8.29)

is sufficient for exact reconstruction. To hold this condition we have to factorize the histogram, i.e., we have to choose a \( \lambda > 0 \) such that

\[
\hat{x}_i = \frac{a_i + a_{i+1}}{2\lambda},
\]

\[
\sup_i (\hat{x}_{i+1} - \hat{x}_i) = \delta < 1,
\]

(8.30)

for \( i = 1, \ldots n - 1 \) and the step function in (8.23) becomes

\[
\tilde{f}_H(x) = \begin{cases} 
    w_i, & \text{for } x \in \left[ \frac{a_i}{\lambda}, \frac{a_{i+1}}{\lambda} \right) \\
    0, & \text{for } x < \frac{a_i}{\lambda} \text{ or } x \geq \frac{a_n}{\lambda}.
\end{cases}
\]

(8.31)

Consequently the length of the modified histogram is

\[
\hat{d} = \frac{a_{i+1} - a_i}{\lambda} \quad \text{for } i = 1 \ldots n - 1.
\]

(8.32)

With these modified data we can reconstruct from the samples

\[
(f_A * u)(\hat{x}_i) = \tilde{f}_H(\hat{x}_i)
\]

(8.33)

the desired density function by

\[
f(x) = f_A(x \cdot \lambda).
\]

(8.34)

Figure 8.1 shows the histogram with the underlying density function. We want to find a good approximation for the density function from the given histogram by using approximations from the space of shift invariant functions.
Again we use the shift-invariant spline spaces with the $B$-spline of order 3 as the generator of $V(\varphi)$.

As we can see in figure 8.2 the algorithm yields to a good approximation of the original density function.
In figure 8.3 the difference between approximation and original density function is plotted.

![Figure 8.3. Error: Reconstructed versus original density function](image)

In figure 8.4 we see the exact reconstruction of the midpoints of the bins $x_i$ by the function $(f_A * u)$, i.e., for $i = 1, \ldots, n - 1$

$$(f_A * u)(x_i) = w_i \quad (8.35)$$

and therefore

$$\int_{a_i}^{a_{i+1}} (f_A * u)(x)dx = w_i \cdot d, \quad (8.36)$$

which means that the area under the function $(f_A * u)$ is the same as the area of the bins. The function $(f_A * u)$ can be viewed as moving average.

![Figure 8.4. The histogram with marked midpoints, which are used as samples for the function $(f * u)$. Solid line: Reconstructed function $(f_A * u)$.](image)
Chapter 8

Averages created by different average-functions $u_j$

As described above we have examined the situation, that the averages are always taken with respect to the same window $u$.

Now let us study the more general situation that we are taking averages over areas of different width, i.e.
\[
U_j(x) = \begin{cases} 
\frac{1}{\mu_j}, & \text{for } x \in \left[-\frac{\mu_j}{2}, \frac{\mu_j}{2}\right] \\
0, & \text{otherwise}
\end{cases} \quad (8.37)
\]

Figure 8.5 shows this situation. We are using the same density function, but the averages are taken over areas of different length.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure85.png}
\caption{The density function and the histogram with different bin-width.}
\end{figure}

Consequently the sampling values $w_j$ are given by
\[
w_j = (f \ast u_j)(x_j) = \frac{1}{\mu_j} \int_{x_j - \mu_j/2}^{x_j + \mu_j/2} f(\xi) d\xi \quad (8.38)
\]

Therefore we have to modify the algorithm described in chapter 7.4 in the following way:

1. For each sample-value $w_j$, we have to calculate the appropriate average-function $u_j$, $j = 1, \ldots, J$. Let be $f \in V(\varphi)$, then
\[
\psi_j = \varphi \ast u_j. \quad (8.39)
\]
2. We have to store these $\psi_j$ to compute $b = U^*w$

$$b_k = \sum_{j=1}^{J} \psi_j(x_j - k)w_j.$$  \hfill (8.40)

3. Also the calculation of the matrix $T$ has to be modified:

$$T_{kl} = \sum_{j=1}^{J} \psi_j(x_j - k)\psi_j(x_j - l).$$  \hfill (8.41)

Having a closer look on the system matrix $T$ we can verify, that the matrix $T$ does not have the same structure as before.

![Figure 8.6: The system matrix $T$ in two different situations.](image)

Since the support of $\psi_j$ depends on the width of the bins (compare with (8.17)), i.e.

$$\text{supp } \psi_j \subset \left[-S - \frac{\mu_j}{2}, S + \frac{\mu_j}{2}\right]$$  \hfill (8.42)

the width of the band in the matrix $T$ varies too.

Applying the modifications described above we get following approximation of the density function as plotted in figure 8.7:
Chapter 8

Figure 8.7.: Reconstruction (solid line) and the original density function (dotted line)
9. Conclusion and Summary

In this chapter we want to give a short summary about the different approaches on the topic reconstruction from averages and also compare the algorithms, which are presented in this work. This summary should illustrate in which situations we get better results by using the new algorithms developed in this work.

Basically we distinguish between two situations depending which space the unknown function \( f \) that we want to reconstruct or at least approximate, belongs to: The space of bandlimited functions \( f \in B_\Omega \) and the space of shift-invariant functions \( f \in V(\varphi) \).

Although the space of bandlimited functions is a special case of the space of shift-invariant functions (if \( \varphi(x) = \frac{\sin \pi x}{\pi x} \), then \( V(\varphi) \) consists exactly of the band-limited functions with spectrum in \([-1/2, 1/2]) \) it is worth to handle the case of \( B_\Omega \) separably, because special features of \( B_\Omega \) can be used to perform efficient algorithms.

Recall following notations:

- \( \delta \), which defines the maximal gap between the sampling points
  \[
  \delta = \sup_{k \in \mathbb{Z}} (x_{k+1} - x_k);
  \]

- \( \mu \), which defines the maximal length of the support of the averaging function \( u \) concentrated around the position \( x_k \)
  \[
  \text{supp } u_k \subset [x_k - \frac{\mu}{2}, x_k + \frac{\mu}{2}];
  \]

- \( \gamma \), which describes the convergence rate of an iterative algorithm
  \[
  \|f - f_n\| \leq \gamma^{n+1}\|f\|.
  \]

9.1. Algorithms for the space of bandlimited functions

Starting with the fundamental statement of Feichtinger and Gröchenig [28] that for any bounded operator \( S \) on a Banach space \( B \) that satisfies

\[
\|f - Sf\|_B \leq \gamma\|f\|_B,
\]

93
Chapter 9

with $\gamma < 1$ the following reconstruction schema can be applied

Iterative schema (IS1) \[
\begin{align*}
  f_0 &= Sf \\
  f_{n+1} &= f_n + S(f - f_n).
\end{align*}
\]

For $f \in B_\Omega$ we can use weighted frames, defined as in Theorem 8, to calculate the frame operator

\[ Sf := \frac{\pi^2}{\pi^2 + \delta^2 \Omega^2} \sum_{k \in \mathbb{Z}} f(x_k) w_k \frac{\Omega}{\pi} T_{x_k} \text{sinc}_\Omega \]

and apply this operator to the algorithm with $\gamma = \frac{2\pi \delta \Omega}{\pi^2 + \delta^2 \Omega^2}$. The restriction on the sampling set is

$\delta < \frac{\pi}{\Omega}$.

In the same article [28] a version for average-sampling is presented with

$\mu \leq \delta < \frac{\pi}{\Omega}$,

\[ u_k = \frac{1}{y_k - y_{k-1}} \chi_{[y_{k-1}, y_k)}, \]

where $y_k$ are the midpoints of the sampling points. For this situation the convergence is

$\gamma = \frac{\delta \Omega}{\pi}$.

Based on Theorem 9 (Gröchenig [32]) that says that for

$\mu = \delta < \frac{1}{\sqrt{2\Omega}}$

every $f \in B_\Omega$ is uniquely determined by the local averages $\langle f, u_k \rangle$, Sun and Zhou [71] improved this statement for

$\delta < \frac{\pi}{\Omega}$

and

$\mu < \frac{\pi}{\Omega} - \delta$.

In the last years Sun and Zhou provided a couple of papers on the topic of reconstruction from averages. Several improvements were presented especially on the estimates of the frame-bounds (see Theorem 11 or Theorem 14), which are critically for defining the frame-operator.
All the algorithms above use the frame-operator $S$ and the *iteration schema* $(IS1)$ described above.

Following the philosophy that efficient numerical algorithms always have to use the special structure of the problem, we present in chapter 6.2 an alternative algorithm:

Focus on the numerical aspect of the discretized reconstruction problem the main feature of bandlimited signals is obviously that the number of relevant coefficients $R = \#\{c_r : c_r \neq 0\}$ is usually much smaller than the length of the signal $N$. Using this specific structure ($R << N$) of $B_\Omega$ it is reasonable to act on the coefficient not on the whole signal, as the iterative schema $(IS1)$ does. As described in chapter 6.2 the frame-operator, combined with a projection on the space of bandlimited functions, appears as

$$S = F_\Omega \ast B_2 \ast B_2^t \ast F_\Omega^t \ast T^t \ast T,$$

which can be calculated on the knowledge of the averages and the number of coefficients. Notice that $S$ is a matrix of $R \times R$ and can be calculated and stored before iterations start. This property of the algorithm makes it possible that even long smooth signals can be reconstructed in an efficient and memory economical way.

A further very essential improvement of the proposed algorithm is the technique how the linear system

$$c_{\text{given}} = S \ast c_{\text{new}}$$

is solved. Because of the fact that $S$ is positive definite we can apply the *Conjugate Gradient-Method*, which is one of the efficient techniques for solving linear systems.

The third important aspect of the algorithm is that by introducing a relaxation parameter $\nu$, we can handle the problem of gaps between the regions of averaging in a satisfied way (see figure 6.14).

**Remark:** As in many applications on bandlimited functions the problem raises that an a priori knowledge of the size of spectrum $\text{supp} (\hat{f}) \subset [-\Omega, \Omega]$ is required at least approximatively. There exist different approaches for solving this problem, which is usual an ill-posed one. One way in finding an approximation of this optimal bandwidth results from a special aspect of regularization techniques - the *L-curve* - and can be found for example in Schwab [65].
9.2. Algorithms for the space of shift-invariant functions

The iterative schemes for the reconstruction of functions from their samples have been introduced by Feichtinger and Gröchenig for the case of bandlimited functions [27] and have been extended by Aldroubi and Feichtinger to general shift-invariant spaces [3].

In [2] Aldroubi presented following iterative algorithm:

By using a BUPU \( \{ \beta_j \}_{j \in J} \) (see definition 11) a quasi-reconstruction operator

\[
A_{X,a}f = \sum_{j \in J} (f \ast u_a^*)(x_j) \beta_j
\]

is defined and used in combination with the orthogonal projection onto \( V(\varphi) \) for an iterative algorithm (see Theorem 21)

\[
(IS2) \left\{ \begin{array}{l}
        f_1 = PA_{X,a}f \\
        f_{n+1} = PA_{X,a}(f - f_n) + f_n.
    \end{array} \right.
\]

The convergence is geometric that is

\[
\|f - f_n\|_{L^2} \leq C\alpha^n \|f - f_1\|_{W(L^2)},
\]

where under some conditions the convergence rate \( \alpha \) can be specified as in Theorem 23.

Studying the structure of the space of shift-invariant functions \( V(\varphi) \) under the assumption that the generator \( \varphi \) has compact support Gröchenig and Schwab [35] presented a fast reconstruction algorithm. We showed that calculating the coefficients \( \vec{c} \) of the unknown signal is equivalent with solving the linear system (7.39)

\[
T\vec{c} = \vec{b},
\]

where the matrix \( T \) is a band matrix, whose bandwidth is \( 2S + 1 \), where \( S \) is the length of the support of the generator.

This new approach of the proposed algorithm of chapter 7.4 involves following significant advantages:

1. **The reconstruction is done locally:** Instead of solving the large system of equations we can apply data segmentation and solve several systems of smaller size. This plays an important role in the situation of large signals, which cannot be handled at once because of memory problems. Furthermore the features of local reconstruction and data segmentation make it possible to apply parallel processing because non-adjacent intervals can be handled simultaneously.

2. **Efficient solution of the linear system:** Since the matrix \( T \) is a positive definite bandmatrix the band Cholesky algorithm is a very fast method that minimizes the operation count. This results in an "optimal" algorithm in the sense that the operations of the algorithm are linear to the number of data.
As in chapter 8.2 presented this fast algorithm can be applied to the situation of moving averages. Since the functions $f * u$ and $f$ share the same coefficients but different generators we are able to calculate the coefficients of the averaging function $f * u$ and reconstruct the original function $f$ with these coefficients. Later in chapter 8.2.3 it is shown that this algorithms can handle also situations, where the average functions vary slightly.

We presented a relevant application on reconstruction from averages using the proposed algorithm namely the reconstruction of the density from histograms:

This problem is a well studied one and several algorithms like kernel density estimation are presented. Although these techniques are wildly used they have one disadvantage in common: the explicit knowledge of the measurements $\{X_i\}_{i=1}^N$. Our goal was to reconstruct the density function not from the measurements $X_i$, because they might not be available or disturbed by some error, but to use only the data of the histogram for reconstruction. In this new approach for density-estimation histograms do not only appear as representation of data but as data themselves. A satisfying solution of this problem can be viewed in figure 8.2.
Appendix A
A. Program-Codes

In this appendix we want to present the program-codes for the proposed algorithms. All numerical examples in this work were done by using MatLAB. To keep the programmes readable we only want present here Pseudo-Codes. E.g. the explicit description of the MatLAB-code of following set

\[ \bar{x}_p^{rel} = \{ x \in \bar{x}_p : k - S < x < k + S \}, \]

would be to the disadvantage of readability. Therefore we write it in the following way

\[ \text{xp\_rel} = \{ \text{xp} : k - S < \text{xp} < k + S \}, \]

which is not a runnable MatLAB-Code but self-explanatory.

To receive the complete MatLAB-code of the following algorithms please contact the author!
A.1. Reconstruction Algorithm for Band-limited Function from Averages

function xa = REC_AVCG(ax, B, r, maxiter);

% REC_AVCG.M 'REC'onstruction of a bandlimited function,
% from 'AV'erages using 'CG'-Method
% ----------------------------------------------------------
% USAGE : xa = rec_avcg(ax, B, r, maxiter);
% %
% % INPUT : ax ... averages (1 x p)
% % B ... BUPU created eg. by B=bupuvor(xp,n)
% % r ... max. relevant frequency
% % maxiter ... max. number of iterations
% %
% % OUTPUT : xa ... reconstruction
% 
% % AUTHOR(s) : H.Schwab 05.2000
% %
% % COPYRIGHT : (c) NUHAG, Dept.Math., University of Vienna, AUSTRIA
% % http://nuhag.mat.univie.ac.at/
% % Permission is granted to modify and re-distribute this
% % code in any manner as long as this notice is preserved.
% % All standard disclaimers apply.
% %

[b1,n]=size(B);

%%% calculation of matrix S
BD=B./(diag(B*B')*ones(1,n)); %%% matrix B2
F = fft([diag(ones(1,r)) zeros(n-r,r)]'); %%% fourier-transformation-matrix
Fi = fft([diag(ones(1,r)/n) zeros(n-r,r)]'); %%% inverse fourier-transformation-matrix
S=Fi'*BD'*B*F'; %%% system matrix S
BF=B*F'; xa=zeros(1,n);

%%% iteration
for iter=1:maxiter
   c_given=ax*BF; %%% given coefficients
   c_new=cg(S,c_given); %%% solving sytem: c_given = S * c_new %%% using CG-METHOD!!
   xs=ifft([c2 zeros(1,n-r)]); %%% calculating function from coeff.
xa=xa+xs;          \textit{%%% update}
ax=ax-xa*BD';      \textit{%%% new averages}
end
Appendix A

A.2. Local Reconstruction Algorithm for a Shift-invariant Function

function [yr, c] = REC_SP(xp, xs, x_rec, gen, S, segments);

% REC_SP.M 'REC'onstruction of a function belonging to a 'SP'line-space
% from given sampling values.
% The spline-space is generated by the function
% defined in the file GEN.M, where a sampled
% version of the generator is used!
% ------------------------------------------------------------------------
% USAGE : [yr, c] = rec_sp(xp, xs, x_rec, gen, S, segments); %
% % INPUT : xp ... sampling positions (continuous)
% xs ... sampling values at position xp
% x_rec ... positions where function should be
% reconstructed (continuous)
% segments... number of segmentations (default == 1)
% gen ... generator (=spline)
% S ... supp(gen)=[-S,S]
% %
% OUTPUT : yr ... reconstructed values at position xr
% c ... reconstructed coefficients
% %
% AUTHOR(s) : H.Schwab 11.2001
% %
% COPYRIGHT : (c) NUHAG, Dept.Math., University of Vienna, AUSTRIA
% http://www.univie.ac.at/NuHAG/
% Permission is granted to modify and re-distribute this
% code in any manner as long as this notice is preserved.
% All standard disclaimers apply.

step = round((max(x_rec)-min(x_rec))/segments);
for k=min(x_rec):step:max(x_rec)
    xp_rel = {xp: k-S < xp < k+step+S} %relevant sampling positions
    xp_min = min(xp_rel);
    xp_max = max(xp_rel);
    J = length(xp_rel); %number of sampling points

    % calculation of the left side b:

102
for j=1:J
    mi=ceil(xp(j))-S;
    ma=floor(xp(j))+S;
    for l = mi : ma
        b(l-mi+1) = b(l-mi+1) + xs(j)*gen(xp(j)-l);
    end
end

% calculation of the matrix T:
T=zeros(xp_max-xp_min+1+2*S,xp_max-xp_min+1+2*S);
for j=1:J
    mi=ceil(xp(j))-S;
    ma=floor(xp(j))+S;
    for k=mi:ma
        for l=mi:ma
            T(k-mi+1,l-mi+1) = T(k-mi+1,l-mi+1) + gen(xp(j)-l)*gen(xp(j)-l);
        end
    end
end

% calculation of the coefficients
    c_part = chol(T,b);  \% solving the system T*c_part=b with a
    \% banded Cholesky algorithm
    c(xp_min-S:xp_max+S) = c(xp_min-S:xp_max+S) + c_part;
    n(xp_min-S:xp_max+S) = n(xp_min-S:xp_max+S) + ones(xp_max-xp_min+1+2*S);
    \%n ... normalization of coefficients because of overlapping
end

c = c ./ n;  \%normalization of coefficients because of overlapping

% calculation of the reconstruction
yr = get_fv(x_rec, c, gen);
A.3. Local Reconstruction Algorithm for a Shift-invariant Function on a Lattice using Segmentation

function [yr,xr] = REC_SP_GL(xp, xs, M0, M1, a, gen, S);

% REC_SP_GL.M 'REC'onstructs a function of a 'SP'line-space from given
% sampling values on a 'L'attice using 'F'ast segmentation.
% Reconstruction for the interval [M0, M1] and samples are
% on a grid with grid-size 'a'. fast segmentation is done
% for discrete FFT of size 2^power_of_2.
% ---------------------------------------------------------------------
% USAGE : [yr,xr] = rec_sp_lf(xp, xs ,M0 ,M1 ,a ,ord);
%
% INPUT : xp ... sampling positions (on grid with size 'a')
% xs ... sampling values at position xp
% [M0,M1]... interval where function should be
% reconstructed (on a lattice with size 'a')
% a ... lattice-size
% gen ... generator (=spline)
% S ... supp(gen)=[-S,S]
%
% OUTPUT : yr ... reconstructed values within the interval [M0, M1]
% on grid with size 'a'
% xr ... xr = [M0:a:M1];
%
% AUTHOR(s) : H.Schwab 04.2003
%
% COPYRIGHT : (c) NUHAG, Dept.Math., University of Vienna, AUSTRIA
% http://www.univie.ac.at/NuHAG/
% Permission is granted to modify and re-distribute this
% code in any manner as long as this notice is preserved.
% All standard disclaimers apply.

%%% initialization

power_of_2=13;
if nargin<5 a=0.1;end
if nargin<6 ord=3;end
xr=[M0:a:M1];
yr=zeros(size(xr));

%%% segmentation to 2^power_of_2
ST=M1-M0;

length_c = (M1-M0+4*S)/a;
llog = log(length_c)/log(2);

if lllog > power_of_2
    ST = 2^power_of_2*a-4*S;
    M1_seg = min(M0+ST,M1);
end M0_seg=M0;

while M1_seg <= M1
    length_c = round((M1_seg-M0_seg+4*S)/a);
    (xp_seg, xs_seg) = {(xp,xs) : M0_seg-S < xp < M1_seg+S};
    J = length(xp_seg);
    k_min = ceil(M0_seg-S);
    k_max = floor(M1_seg+S);
    ms = k_max-k_min+1;

%%% calculation: T*c
b = zeros(1,ms);
kk = 0;
for k = k_min : k_max
    kk = kk+1;
    w = abs(xp-k);
    f = find(w<=S);
    w = round(w/a);
    b(kk) = gen(w(f)+1) * xs(f)';
end

%%% calculation: matrix T
T = sparse(zeros(ms,ms));
for k = k_min : k_max
    for l = max(k-S, k_min) : min(k+S, ms+k_min-1)
        ll = l-k_min+1;
        kk = k-k_min+1;
        T(1,kk) = T(1,kk) + b(ll) * x(1);
Appendix A

\[
\begin{align*}
w1 &= \text{abs}(x_p-k); \\
w2 &= \text{abs}(x_p-l); \\
f &= \text{find}(w1<S); \\
w1 &= w1(f); w2 &= w2(f); \\
f &= \text{find}(w2<S); \\
w1 &= w1(f); w2 &= w2(f); \\
w1 &= w1/a; w2 &= w2/a; \\
T(kk,ll) &= T(kk,ll) + \text{gen}(w1+1) \ast \text{gen}(w2+1)'; \\
\end{align*}
\]

end
end

%%% solving T*c=b
\[
\text{c\_part} = \text{chol(T,b); \% solving the system T*c = b with a \% banded Cholesky algorithm}
\]

%%% calculation of reconstruction xa
\[
\begin{align*}
c2 &= \text{zeros(1,length_c);} \\
c2( (k_{\text{min}}-M0\_seg+2*\text{S})/\text{a+1} : 1/\text{a} : (k_{\text{max}}-M0\_seg+2*\text{S})/\text{a+1} ) &= \text{c\_part}; \\
gen2 &= [\text{gen(1:S/a)} \ zeros(1,length(c2)-2*S/a) \ gen(S/a+1:-1:2)]; \\
& \% zero padding because \\
& \% gen2 must have the same length as c2 for FFT \\
\end{align*}
\]

\[
\text{yr\_seg} = \text{ifft(fft(c2).*fft(gen2))}; \\
& \% calculating the reconstruction of \\
& \% the segment via FFT (length = 2^\text{power_of_2}!!)
\]

\[
\text{yr\_seg} = \text{yr\_seg( 2*S/a+1 : (M1\_seg-M0\_seg+2*S))/a+1 )}; \\
& \% cutting off unwanted boundary effects of FFT
\]

\[
\text{yr( (M0\_seg-M0)/a+1 : (M1\_seg-M0)/a+1 ) = yr\_seg}; \\
& \% updating reconstruction
\]

%%% calculate new M0\_seg, M1\_seg
\[
\begin{align*}
M0\_seg &= \text{floor(M1\_seg - 2*S)}; \\
\text{ende} &= 0;\text{if M1\_seg == M1} \ \text{ende} = 1; \text{end} \quad \% \text{is M1\_seg == M1 then end loop} \\
M1 &= M0 + \text{ST}; \\
& \% new upper bound of interval \\
\text{if M1 > MM1 | M1 < M0} \ M1 &= \text{MM1}; \text{end} \\
\text{if M1 == M0;} \ M1 &= \text{MM1} + 1; \text{end} \\
M1 &= M1 \ast (\text{ende}+1); \\
& \% if ende==1 then end loop
\end{align*}
\]
end
A.4. Reconstruction of the Density-Function from a given Uniform Histogram

function [yr, xr] = HIST2DENS_SP(x, y, xs, gen, S);

% HIST2DENS_SP 'HIST'ogram-to-'DENS'ity using 'SP'lines;
% Reconstructs the density function from a
% histogram, by using shift-invariant-spaces-
% methods. (e.g. bsplines)
% ----------------------------------------------------------
% USAGE : [yr, xr] = hist2dens_sp(x, y, w, gen);
% INPUT : x,y,xs ... histogram
% x ... vector of starting-points of the bins
% y ... vector of ending-points of the bins
% xs ... heights of the bins
% gen ... generator of shift-invariant space
% S ... supp(gen) = [-S,S]
% OUTPUT : yr ... density function
% xr ... x-values of density function
% AUTHOR(s) : H.Schwab 12.2002
% COPYRIGHT : (c) NUHAG, Dept.Math., University of Vienna, AUSTRIA
% http://univie.ac.at/NuHAG
% Permission is granted to modify and re-distribute this
% code in any manner as long as this notice is preserved.
% All standard disclaimers apply.

%%% sampling points in the middle of the bins
xp=(x+y)/2;

%%% initial input
trans=min(xp);
xp=xp-trans;
fact=max(xp)/(length(xp)-1);
xp=xp/fact+1;

%%% bin-size:
Appendix A

\[
\text{binsz} = \text{round}((\text{xp}(2) - \text{xp}(1)))
\]

\[
\text{gc} = 100;
\]

\[
\text{xr} = \left[\text{min}(\text{xp}) : 1/\text{gc} : \text{max}(\text{xp})\right];
\]

%%% convolution generator & av-fct

\[
\text{filt} = \left[\text{ones}(1, \text{binsz}/2 * \text{gc}) \ z\text{eros}(1, \text{length(\text{gen})) \ ones}(1, \text{binsz}/2 * \text{gc})\right];
\]

\[
\text{filt} = \text{filt}/\text{sum(filt)};
\]

\[
\text{g} = \text{ifft(fft(\text{gen} .* fft(filt)));
\]

%%% support of generator & av-fct

\[
\text{S} = \text{S} + \text{binsz}/2;
\]

%%% INITIALISATION

\[
\text{yr} = \text{zeros}(1, \text{length(xr)});
\]

\[
\text{x_min} = \text{min(xr)};
\]

\[
\text{x_max} = \text{max(xr)};
\]

\[
\text{J} = \text{length(xp)};
\]

\[
\text{xp_min} = \text{ceil}((\text{xp}(1)) - 1);
\]

\[
\text{xp_max} = \text{floor}(\text{xp}(J)) + 1; \quad \text{Sgc} = \text{S} * \text{gc};
\]

\[
\text{yr} = \text{rec_sp}(\text{xp}, \text{x}, \text{xr}, \text{gen}, \text{S}, 1);
\]

\[
\text{xr} = (\text{xr} - 1) * \text{fact} + \text{trans};
\]

%%% normalize density function: \text{integral(\text{yr})} = 1

\[
\text{yr} = \text{yr} / (((\text{xr}(2) - \text{xr}(1)) * \text{sum(\text{yr)}));
\]
A.5. Reconstruction of the Density-Function from a given Nonuniform Histogram

function [yr, xr] = HIST2DENS_SP_NU(x, y, xs, gen, S);

% HIST2DENS_SP 'HIST'ogram-to-'DENS'ity using 'SP'lines
% for 'N'on-'U'niform average-functions;
% Reconstructs the density function from a
% histogram, by using shift-invariant-spaces-
% methods (e.g. bsplines). The average functions
% have different supports.
%----------------------------------------------------------
% USAGE : [yr, xr]=hist2dens_sp_nu(x, y, xs, gen, S);
%
% INPUT : x,y,xs ... histogram
% x ... vector of starting-points of the bins
% y ... vector of ending-points of the bins
% xs ... heights of the bins
% gen ... generator od shift-invariant space
% S ... supp(gen) = [-S,S]
%
% OUTPUT : yr ... density function
% xr ... x-values of density function
%
% AUTHOR(s) : H.Schwab 12.2002
%
% COPYRIGHT : (c) NUHAG, Dept.Math., University of Vienna, AUSTRIA
% http://nuhag.mat.univie.ac.at/
% Permission is granted to modify and re-distribute this
% code in any manner as long as this notice is preserved.
% All standard disclaimers apply.

xp=(x+y)/2;  \% sampling points in the middle of the bins

%%%% initial input
trans=min(xp);
xp=xp-trans;
fact=max(xp)/(length(xp)-1);
xp=xp/fact+1;

gc=100; xr = [min(xp):1/gc:max(xp)]; J=length(xp);
Appendix A

\[ GG = \text{zeros}(J, \text{length}(\text{gen}) \times \text{max}(\text{diff}(\text{xp})); \ SS = []; \]

\%\% calculating the convoluted generators
\for j = 1 : J-1
\%\% bin-size:
\hspace{1cm} \text{binsz} = \text{round}((x(j) - y(j));

\%\% convolution generator & av-fct
\hspace{1cm} \text{filt} = [\text{ones}(1, \text{binsz}/2*gc) \ \text{zeros}(1, \text{length}(\text{gen})) \ \text{ones}(1, \text{binsz}/2*gc)];
\hspace{1cm} \text{filt} = \text{filt}/\text{sum}(%\text{filt});
\hspace{1cm} g = \text{ifft}(%\text{fft}(\text{gen}).*\text{fft}(%\text{filt}));

\%\% store convolution in matrix GG
\hspace{1cm} \text{GG}(j,1:\text{length}(g)) = g;

\%\% support of generator & av-fct
\hspace{1cm} \text{SS} = [\text{SS} \ S + \text{binsz}/2];
\end

%%%%% INITIALISATION
\text{yr} = \text{zeros}(1, \text{length}(%\text{xr})); \ x_{\text{min}} = \text{min}(%\text{xr}); \ x_{\text{max}} = \text{max}(%\text{xr}); \ xp_{\text{min}} = \text{ceil}(%(xp(1))-1; \ xp_{\text{max}} = \text{floor}(%(xp(J))+1;

%%%%% calculation of b \quad (T*c=b)
\text{number_coef} = \text{xp}_{\text{max}} - \text{xp}_{\text{min}} + 1; \quad \%\text{number of coefficients}
\text{b} = \text{zeros}(%1,\text{number_coef}); \ \text{for} \ j = 1:J
\hspace{1cm} \text{S} = \text{SS}(j);
\hspace{1cm} g = \text{GG}(j,:);
\hspace{1cm} \text{Sgc} = \text{S} \times \text{gc} + 1;
\hspace{1cm} \text{mi} = \text{ceil}(\text{xp}(j) - \text{S});
\hspace{1cm} \text{ma} = \text{floor}(\text{xp}(j) + \text{S});
\hspace{1cm} \text{for} \ l = \text{mi} : \text{ma}
\hspace{2cm} \text{b}(l - \text{mi} + 1) = \text{b}(k - \text{mi} + 1) + \text{xs}(j) \times g(xp(j)-1);
\end
\end

%%%% calculation of matrix T
\text{T} = \text{zeros}(\text{xp}_{\text{max}} - \text{xp}_{\text{min}} + 1 + 2*%\text{S}, \text{xp}_{\text{max}} - \text{xp}_{\text{min}} + 1 + 2*%\text{S}); \ \text{for} \ j = 1:J

110
S=SS(j);
g=GG(j,:);
mi=ceil(xp(j))-S;
ma=floor(xp(j))+S;
for k=mi:ma
    for l=mi:ma
        T(k-mi+1,l-mi+1) = T(k-mi+1,l-mi+1) + gen(xp(j)-l)*g(xp(j)-l);
    end
end

% calculation of the coefficients
% solving the system T*c_part=b with a % banded Cholesky algorithm
c_part = chol(T,b);
c(xp_min-S:xp_max+S) = c(xp_min-S:xp_max+S) + c_part;
n(xp_min-S:xp_max+S) = n(xp_min-S:xp_max+S) + ones(xp_max-xp_min+1+2*S);

% normalization of coefficients because of overlapping

c = c ./ n;

% calculation of the reconstruction

yr = get_fv(xr, cl, gen);
xr=(xr-1)*fact+trans;

%%% normalize density function: int(yr)=1
yr=yr/((xr(2)-xr(1))*sum(yr));
Appendix A

A.6. Conjugate Gradient Method for linear system of equations

In order to make this thesis self-contained we present here the code for the CG-method. This code is a modified version of the MatLAB built in function CGS.M. This is the exact MatLAB code and can be used for implementation. For more theoretical information about the CG-method the reader is referred to [20, 36, 40, 41].

function x = CG(A, b, tol, maxit);

% CG.M Conjugate Gradient method for linear system of equations ( A * x = b ).
% -------------------------------------------
% USAGE : x = cg(A, b);
% x = cg(A, b, tol);
% x = cg(A, b, tol, maxit);
%
% INPUT : A ... matrix of size N-by-N
% b ... vector of size N-by-1
% tol ... specifies the tolerance of the method
% [default TOL = 1e-06]
% maxit ... specifies the maximum number of iterations
% [default MAXIT = MIN(N,20)]
%
% OUTPUT : x ... solution (or approximation) of A * x = b
%
% AUTHOR(s) : H.Schwab 05.2000
% modified version of the MatLAB built in function CGS.M
% Original function CGS.M by:
% Penny Anderson, 1996.
% Copyright 1984-2002 The MathWorks, Inc.
% $Revision: 1.17 $ $Date: 2002/04/09 00:26:10 $

[m,n] = size(A);

%% Checking input
if (m ~= n)               %%% size of Matrix A
    error('Matrix must be square.');
end
if ~isequal(size(b), [m,1]) %%% size of vector b
    es = sprintf(['Right hand side must be a column vector of' ...
                  ' length %d to match the coefficient matrix.'], m);
    error(es);
end

...
end

% Assign default values to unspecified parameters
if (nargin < 3)
    tol = 1e-6;     % default value for TOL
end
if (nargin < 4)
    maxit = min(n,20);  % default value for MAXIT
end

% Check for all zero right hand side vector => all zero solution
n2b = norm(b);  % Norm of rhs vector, b
if (n2b == 0) % if rhs vector is all zeros
    x = zeros(n,1);  % then solution is all zeros
    return
end

x = zeros(n,1);

%%% Set up for the method
xmin = x;  % Iterate which has minimal residual so far
imin = 0;  % Iteration at which xmin was computed
tolb = tol * n2b;  % Relative tolerance
r = b - A * x;  % Zero-th residual
normr = norm(r);  % Norm of residual
if (normr <= tolb)  % Initial guess is a good enough solution
    return
end

rt = r;  % Shadow residual
normrmin = normr;  % Norm of residual from xmin
rho = 1;
stag = 0;  % stagnation of the method

%%% Loop over maxit iterations (unless convergence or failure)
for i = 1 : maxit
    rho1 = rho;
    rho = rt' * r;
    if rho == 0 | isinf(rho)
        break
    end
    if i == 1
Appendix A

\begin{verbatim}
Appendix A

    u = r;
    p = u;
else
    beta = rho / rho1;
    if beta == 0 | isinf(beta)
        break
    end
    u = r + beta * q;
    p = u + beta * (q + beta * p);
end
ph1 = p;
ph = ph1;
vh = A * ph;
rtvh = rt' * vh;
if rtvh == 0
    break
else
    alpha = rho / rtvh;
end
if isinf(alpha)
    break
end
if alpha == 0 % stagnation of the method
    stag = 1;
end
q = u - alpha * vh;
uh1 = u+q;
uh = uh1;

% Check for stagnation of the method
if stag == 0
    stagtest = zeros(n,1);
    ind = (x ~= 0);
    stagtest(ind) = uh(ind) ./ x(ind);
    stagtest(~ind & uh ~= 0) = Inf;
    if abs(alpha)*norm(stagtest,inf) < eps
        stag = 1;
    end
end

x = x + alpha * uh; % form the new iterate
normr = norm(b - A * x);
\end{verbatim}
if normr <= tol
    break
end

if stag == 1
    break
end

if normr < normrmin
    normrmin = normr;
    xmin = x;
    imin = i;
end

qh = A * uh;
r = r - alpha * qh;
end

% for i = 1 : maxit
B. Bibliography


Bibliography


Bibliography


List of Figures

3.1. Velocity of a bicycle - Update of the velocity every 5 seconds . . . . . . . 14
3.2. Velocity of a bicycle - Update of the velocity 8 meters . . . . . . . . . . . 15
6.1. Examples of standard averaging functions . . . . . . . . . . . . . . . . . 32
6.2. Average values calculated by box and triangle function . . . . . . . . . . . 32
6.3. Average values calculated by spline and box function with cosine flanks . 32
6.4. Moving averages . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 38
6.5. Best $L^2$ approximation . . . . . . . . . . . . . . . . . . . . . . . . . . 39
6.6. Matrix containing the indicator functions . . . . . . . . . . . . . . . . . . 40
6.7. Regions of averages . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 41
6.8. First approximation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 43
6.9. Original function and the averages . . . . . . . . . . . . . . . . . . . . . . 45
6.10. Iteration steps of reconstruction . . . . . . . . . . . . . . . . . . . . . . . 46
6.11. Logarithmic reconstruction error . . . . . . . . . . . . . . . . . . . . . . . 46
6.12. Positive definite diagonal-dominated system matrix $S$ . . . . . . . . . . . 47
6.13. Original function with averages . . . . . . . . . . . . . . . . . . . . . . . 47
6.14. Reconstruction and error . . . . . . . . . . . . . . . . . . . . . . . . . . . 48
7.1. Generator $\varphi$ . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 69
7.2. Linear operation count of algorithm . . . . . . . . . . . . . . . . . . . . . 69
7.3. Reconstruction from noisy nonuniform samples . . . . . . . . . . . . . . . 71
7.4. Reconstruction with different generators . . . . . . . . . . . . . . . . . . . 73
7.5. Structure of the banded matrix $\mathcal{T}$ . . . . . . . . . . . . . . . . . . . 74
7.6. Reconstruction of the function on disjoint intervals (without noise) . . . . 74
7.7. Reconstruction of the function using segmentation of size $2^{15}$ . . . . . 77
8.1. Histogram with underlying density function . . . . . . . . . . . . . . . . . 88
8.2. Reconstruction from uniform histogram . . . . . . . . . . . . . . . . . . . 88
8.3. Error in reconstruction from histogram . . . . . . . . . . . . . . . . . . . 89
8.4. Reconstruction of the averages . . . . . . . . . . . . . . . . . . . . . . . . 89
8.5. Histogram with different bin-widths . . . . . . . . . . . . . . . . . . . . . 90
8.6. Analyzing the system matrix $T$ . . . . . . . . . . . . . . . . . . . . . . . 91
8.7. Reconstruction from non-uniform histogram . . . . . . . . . . . . . . . . 92
Curriculum Vitae

Name: Harald Schwab
Adress: Guldengasse 16/3/46
A-1140 Vienna, AUSTRIA
Nationality: Austria
Born on 10th of December, 1969 in Vienna

1975-1980: Grammar School in Vienna
1980-1988: High School in Vienna
1989-1996: Studies in Mathematics at the University of Vienna
Feb. 1996: Master's Degree in Mathematics
Master Thesis on the topic: "Regularization of ill-posed problems and some applications"

Since 1994 working with the group NUHAG (Numerical Harmonic Analysis Group)
headed by Prof. Hans G. Feichtinger, University Vienna, Austria
1997: Visiting the Image and Video Processing Lab at Electrical and Computer Engineer-
ing on Northwestern University (USA) headed by Prof. Aggelos Katsaggelos.
1999: Conference SampTA-99, Loen, Norway,
2000: Conference ICASSP, Instanbul, Turkey
Since 2000: Lector for mathematics and statistics at the Department of Statistics and
Decision Support System
2003: Conference SampTA-03, Strobl, Austria