Sonification for Exploratory Data Analysis

Dissertation

zur Erlangung des akademischen Grades

Doktor der Naturwissenschaften

der Technischen Fakultät der Universität Bielefeld

vorgelegt von Thomas Hermann am 12. Februar 2002

Acknowledgement

First, I would like to thank Helge Ritter for his excellent advisory support of my work and in particular for the many inspiring and helpful discussions.

I would also like to thank all members of the Neuroinformatics Group at Bielefeld for providing a pleasant workplace and a friendly atmosphere. My special thanks go to Peter Meinicke for many fruitful discussions, his interest in this work and for reading and commenting on a draft of this thesis. Further thanks go to Mark Hansen for his invitation to the Bell Labs and the excellent cooperation concerning the McMC model and to Tim Nattkemper for reading parts of the applications chapter.

Many thanks to Andreas Tupak for helping me with the production of the accompanying CD. For linguistic proof-reading the thesis I thank Melanie Werner.

I had the pleasure to be a member of the Graduate Programm “Aufgabenorientierte Kommunikation” and would like to thank the current and former members for many interdisciplinary discussions.

Finally, I am deeply grateful to my girlfriend Heike Knerich for her continual support and her sympathy.

This work was funded by a DFG scholarship (GK 256).

Contents

Chapter 1

Introduction

The relevance of methods for “intelligent” data processing is growing rapidly. The current trend towards a digitalization of almost all processes in our society like electronic cash, banking, electronic commerce, biomedical research and certainly the rapid growth of electronically stored data in the Internet has lead to an increasing need for methods to process them and to extract knowledge from data. This goes parallel with a rapid growth in computational power which seems to have problems to keep pace with the growth of data amount. But despite the increase of computational power, the intelligence of computers in discerning ’meaning’ from data is still very limited so that in any case the help of a domain expert is currently required to solve such knowledge extraction or *data mining problems*. Facing this situation, the question arises how the interface between humans and machine can be improved to support the data mining. Currently, data mining consists of two approaches: (a) *machine learning*: computers are supplied with perceptional abilities, in contrast to human perceptual inputs here optimized for the detection of regularities and structures within high-dimensional data spaces. An example for this is given by artificial neural networks [Bis95]. (b) *exploratory data analysis*: human-machine interfaces are developed which allow human researchers to inspect complex data in such a way that their understanding of important relations in the data increases. For this purpose mainly visual displays have been focused, and the field of scientific visualization has grown strongly during the last decades [Tuk77, dTSS86].

This work will be concerned with various aspects of both approaches. On the one hand, methods within (a) can be suited precursors for data presentation techniques, on the other hand, there is a large potential in optimizing and extending the communication between humans and machine focused in (b), as certain perceptual channels have not yet been exploited adequately. This work focuses in particular on the auditory perception and thus aims to develop techniques which allow to present data in form of sound.

Using sound to convey information is not a new idea. In many domains, auditory interfaces are well established. Some professional examples are the pulsoximeter which is used during medical surgeries to track a patient’s condition [Pul02], acoustic alarms within airplane cockpits or the acoustic monitoring of single neuron cell activities, which is in neurobiology a standard method applied in electro-physical laboratories. However, the number of applications today is extremely small compared to visualizations, if one takes into account how ubiquitous and profitable listening is in our everyday life. Sound and Hearing in the above sense draws our attention to certain (maybe dangerous) events and provides us with an awareness of our (even invisible) environment. It is the channel for linguistic communication and provides us with important feedback on our actions, e.g. while manipulating objects. Besides that, sound can have a strong effect on our emotions: imagine a thriller without background music, or the emotional effect that a symphonic concert can have. We mostly use our auditory system in various ways simultaneously, e.g. while listening to a human speaker and thus focusing on the words and their meaning, we are able to extract subtle changes in prosody and conclude about the speakers emotional state, we further notice sound events in the background and so on. Our auditory system is a very high-developed pattern recognition system and we are able to learn to understand new sounds and to adapt to various acoustic contexts. Considering how important the hearing sense is for us in the real world, the domain of data mining and a corresponding computer workplace for data analysis is astonishingly silent. The reason for this is that there is no canonical way to put data into sound. Data are intrinsically silent. But in the same sense as arbitrary data can be used to create visual displays which allow to draw conclusions about the data from the visualizations, auditory presentations can now be computed which provide new perspectives on data by listening to them.

1.1 Exploratory Data Analysis

The process of data mining can be divided into two parts, an exploratory data analysis (EDA) and a confirmatory data analysis (CDA). EDA is an inherent part of the data mining process and aims at allowing the human researcher to bring in his capabilities for detecting patterns. An EDA is in most cases necessary because an explicit knowledge of pattern in the data is often missing (otherwise the data mining problem is already solved). Even in the case that neural networks or other machine learning techniques are intended to be applied, EDA can provide a valuable help, e.g. to select the network size or the complexity of the model. In general, the more explicit knowledge about the data distribution is available, the better an adequate machine learning technique can be chosen. The second part of data mining is that of confirmatory data analysis. Here, a hypothesis about the data is already given and the question is addressed how well the data can be explained by this hypothesis.

Research on methods of EDA is quite old. Tukey [Tuk70] coined the expression in his famous book of the same title which presents various visualizations of one- and multivariate data like leaf plots, histograms and stem plots. Since then, many new approaches for data visualization have been invented, some examples are self-organizing maps (SOM) [RMS92], multidimensional scaling (MDS) [KB97], projection pursuit display (PP) [Sco92], principal curve visualization (PCV) [Mei00] and principal component projection scatter plots [Sco92]. All these methods aim at reducing the dimensionality of the given data while maintaining the 'main' structure of the data, allowing to present them on a two-dimensional display, the computer screen. With increasing computational power, these approaches can be extended to three-dimensional displays and even interactive dynamic visualizations or interactive animated computer graphics can be used to investigate data, e.g. in a CAVE [CN95].

In comparison to this very elaborated field, the development of sonification is in an rather early stage. The reason for this may be the qualitative differences between visual and auditory perception and their respective media (see Chapter 4). Furthermore there is a cultural bias towards visualization, as techniques to store, reproduce and manipulate graphics are much older than those concerned with sound. Depending on the task and data at hand one of these two modalities may be superior and the idea of sonification is to apply auditory data presentation in those situations where it facilitates the task. The ultimate goal is to use both visualization and sonification complementary to enhance EDA. However, since scientists are very much trained to work with visualizations, sonifications have to cope with a negative bias: the users must first learn how to interprete the sound before they can profitably use sonification. It is furthermore difficult to extract quantitative information from sonifications, and communication about the sound is difficult as we cannot just point onto certain elements in sound as we can in a graphics. Some of these problems can be solved. The learning effort can be reduced by appropriate tutorials and "pointing into sounds" can be solved by suitable human-computer interfaces. Publication of sonifications can now easily be realized via the Internet. What remains is that the interpretation of sonifications must be learned. But this shall not be seen as a disadvantage of sonification, but as an important feature of the human brain and auditory system which is aimed at: humans are able to learn to understand sonifications, and once we are familiar with *listening* as a scientific method, sonification may offer new insight and perspectives to high-dimensional data.

1.2 Sonification

Sonification is the presentation of data using sound. The idea behind it is to render an acoustic data presentation which enables humans to draw conclusions about data properties like trends, clustering or other distributional patterns by listening to the sound.

Auditory displays have already found a broad field of applications. Besides data exploration they can be used to augment perception when the eyes are occupied by another task, like in medical surgeries. Sonification can be applied as a new means of interaction with computer technology for blind people. It can increase the use of acoustic information in virtual reality applications or enhance alarm systems. In many situations, sound is already used for alarm purposes. Sonification is concerned with the improvement of these alarms by studying the required information and designing alarm sounds that are processed easier so that even an increased number of auditory signals can be interpreted correctly (e.g. in airplane cockpits). In some situations, computers are not able to detect an alarming situation properly from the available dynamic data. Then a suited acoustic monitoring of the data can help: humans can easily habituate to auditory streams (consider the sound of the fan in a computer). Then, sudden changes of the sound are rapidly recognized as these changes draw the listener's attention to them. The monitoring of process data provides an important application for sonification, e.g. for stock market transactions, power plant operation, network traffic, manufacturing processes and so on.

The present thesis focuses on sonification for exploratory data analysis. The goal is to provide researchers with a toolbox of methods to render sonifications for new, non-analyzed datasets in order to enhance their understanding and insight into the data, to assist the choice of models to explain or characterize the data. This leads to the question, how data can or should be transformed into sound, which sonification techniques are available to transform data into useful sonifications. In previous research, different types of approaches can be identified which will be discussed in detail in Chapter 3. The most common sonification techniques were obtained by transferring a visualization technique to the auditory domain. Parameter Mapping Sonification, the most common sonification is the acoustic version of a scatter plot. This and other techniques are presented in Chapter 7. A new contribution to sonification techniques developed in this work is Model-Based Sonification. The corresponding *sonification models* help to avoid some of the problems encountered in Parameter Mapping Sonification and can be regarded as a framework for the development of sonifications that are designed to provide information for a specific analysis task at hand.

An important remark must be made concerning the relation to visualization: most data properties and patterns which can be detected from sonifications can also be made visible in specialized visualizations. Obviously, it is no problem for humans to develop an appropriate visualization which just shows what has been detected by listening to the data. A suited visualization may even show a pattern better than the sound does. This is due to the following reasons: firstly, we are very well trained to interprete visual displays, as we learn to extract knowledge from them from our early school days. The lack of practice and maybe also the lack of suited methods here gives sonification a negative bias. Secondly, the way we imagine data is highly influenced by visual concepts (e.g. we frequently consider data as points in vector spaces). From this point of view, 'understanding' is equivalent to acquiring a visual imagination of data. Thus, it is easy for us to develop a visual presentation. So, the reader should keep in mind that the main goal in EDA is that of pattern detection! And for this, the sonification may indeed be superior, less fatigable, faster. Visualizations which show a structure clearer than it can be heard are therefore no argument against sonification.

1.3 Overview

This section gives an overview about the structure of the thesis. Chapter 2 presents some methods for exploratory data analysis. Chapter 3 gives a broad overview about the current state of research on sonification. Some example applications are discussed and some taxonomies introduced which help to classify sonification methods. Chapter 4 then focuses on the basis of sonification: the auditory perception. Some important results from psychoacoustics are reported and the consequences for auditory display are summarized. The physical perspective on sound is taken into consideration in Chapter 5. Here, the mechanisms of sound generation and sound propagation in physical systems are illustrated as far as necessary for the understanding of the rest of the thesis. In the case of sonification, however, the sound is not generated by physical systems but by a computer program. Chapter 6 therefore addresses the field of sound computing, that is how sounds are represented, synthesized and manipulated in a computer. The central part of this thesis is presented in Chapter 7, which introduces Model-Based Sonification, a framework for the generation of task-oriented sonifications. Examples for sonification models are given in greater detail in Chapter 8. Since however in some cases the classical approaches like Parameter Mapping are very well suited for auditory display, it is then discussed in Chapter 9, how Parameter Mapping can be enriched by helpful acoustic clues to facilitate their interpretation. The application of sonic scatter plots as a front-end to MDS is demonstrated as well. Chapter 10 then presents some applications where sonification has already shown its utility: acoustic monitoring of verbatim protocols from psychotherapeutic sessions, sonifications for the exploratory analysis of EEG data and sonifications to assist the browsing of multi-channel image data will be presented. Finally, the thesis closes with a discussion and conclusion.

1.4 Conventions and Notation

In this thesis, a dataset is always considered to be given by a set of d-dimensional vectors $\mathbf{x}_i$, $i = 1, \ldots, n$. Vectors are written in lowercase boldface letters and thought of as column vectors, i.e. $d \times 1$ matrices. For matrices, uppercase boldface letters are used. Datasets will often be written as a matrix with records in rows, e.g. $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n]^{\mathrm{T}}$. The ith column vector of $\mathbf{X}$ is selected by $\mathbf{X}_{*i}$, the ith row vector is $\mathbf{X}_{i*}$, but in most cases $\mathbf{x}_i^{\mathrm{T}}$ is used to address the ith row of $\mathbf{X}$. The following special symbols are used:

- $\boldsymbol{I}_d$ is given by $(\boldsymbol{I})_{ij} = \delta_{ij}$ and thus is the $d \times d$ identity matrix.
- $\mathbf{1}_n$ is a $n \times 1$ matrix with all elements equal to 1.

- $\mathbf{1}_{n\times d}$ is a $n \times d$ matrix with all elements equal to 1.

- $\mathcal{N}_d\{\boldsymbol{\mu}, \boldsymbol{\Sigma}\}$ denotes a normal distribution in d dimensions.

- $\mathrm{Tr}(\mathbf{A})$ denotes the trace of square matrix $\mathbf{A}$.

- $\mathbf{C}_p(X)$ denotes the covariance matrix of a random variable X with density p given by $\int(\mathbf{x}-\boldsymbol{\mu})(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}} p(\mathbf{x}) d\mathbf{x}$ where $\boldsymbol{\mu}$ is the expectation of $\mathbf{x}$ under p.

- $\mathbf{S} = \hat{\mathbf{C}}(\mathbf{X})$ denotes the sample covariance matrix of dataset $\mathbf{X}$ given by

$$\mathbf{X}_0^{\mathrm{T}}\mathbf{X}_0/N \text{ with } \mathbf{X}_0 = \mathbf{X} - \mathbf{1}_{N\times N}\mathbf{X}/N\ . \tag{1.1}$$

- $g(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ is the density of $\mathcal{N}_d\{\boldsymbol{\mu}, \boldsymbol{\Sigma}\}$, given by

$$\det(2\pi\boldsymbol{\Sigma})^{-1/2}\exp((\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})) \tag{1.2}$$

- $\|\mathbf{X}\|_F = \sqrt{\sum\limits_{i,j}|x_{ij}|^2}$ is the Frobenius norm of matrix $\mathbf{X}$.

Also the following abbreviations will be used

AIB:	Auditory Information Bucket
CDA:	Confirmatory Data Analysis
DCS:	Data Crystallization Sonification
DSG:	Data Sonograms
EDA:	Exploratory Data Analysis
ID:	Intrinsic Dimensionality
GNG:	Growing Neural Gas
GNGS:	Growing Neural Gas Sonification
GUI:	Graphical User Interface
McMC:	Markov chain Monte Carlo
MDS:	Multidimensional Scaling
MBS:	Model-Based Sonification
PCA:	Principal Component Analysis
PC:	Principal Curve
PCS:	Principal Curve Sonification
PMS:	Parameter Mapping Sonification
PP:	Projection Pursuit
PTS:	Particle Trajectory Sonification
SOM:	Self-Organizing Map
SVD:	Singular Value Decomposition
w.r.t.	with respect to

Sound is represented either by real functions $v(t) = v_t$ or by row vectors $\mathbf{v}^{\mathrm{T}}$, whose components represent the successive sound pressure values for a time-discrete signal. For the time-discrete representation also both $v[n]$ and v_n are used to refer to the nth sample.

Sound Examples

Essential for the understanding of the sonifications and their discussion is the presentation of sound examples. It is difficult to include audio into the printed text. Therefore, sound examples are provided on the accompanying compact disk which is attached to the book cover. The recommended way of reading the thesis is to view the **HTML document** `index.html` on a web browser. All sound examples in the thesis are underlined in the text and can be accessed from the HTML document.

There are two further alternatives to access the sound examples:

- **PDF-File**: On the CD-ROM, the dissertation thesis is available in Portable Document Format (PDF), so that all sound examples can be activated by following the links. Sound examples are underlined and their activation will cause the sound files to be played.
  - Windows/Mac users have to access the PDF document from `index.html`, so that the PDF-file is embedded into the Browser. This is necessary since relative links are used in the PDF document.
  - On Unix/Linux, the document has to be accessed directly and a “Base URL” has to be specified in the ’File/Document Info’ Menu of the Acrobat Reader, since relative links seem not to work in the Browser plug-in. Assume the CD is mounted to `/cdrom`. Then the Base URL is ’`file:///cdrom`’.
- **Audio-CD**: the CD-ROM may be played with any audio CD player. To keep the number of tracks small, sound examples are grouped. The track indices are given with the pointers to the examples.

Chapter 2

Exploratory Data Analysis

The objective of this thesis is to develop methods to render auditory presentations of datasets in such a way that the users ideas or knowledge about regularities and patterns hidden in the data increases by listening to them. This is based on the idea that before one of the many available models and machine learning techniques can be applied, it is necessary to decide which one to use. Therefore, the first step in the process of data mining is to understand what *can* be done rather than to learn how well this method or the other *has done* it. Tukey used the notions 'Exploratory Data Analysis' (EDA) and 'Confirmatory Data Analysis' (CDA) for these two very different steps [Tuk77]. He compared knowledge extraction to the process of criminal justice, since the latter can be divided into investigation or detective work which aims at finding evidence, and jurisdiction, which evaluates the strength of the evidence. The same distinction can be helpful in understanding knowledge extraction: exploratory data analysis is like detective work, confirmatory data analysis is quasi-judicial in character.

EDA provides the vehicle for discovering the unexpected and usually has a strong impact on the insight and understanding of the data at hand. As a result it enables the researcher to formulate hypothesis about how the data are structured. A time series, for example, may have a long range periodicity - but without the idea that such a structure could be hidden in the data, one perhaps wouldn't check for it. Appropriate hypothesis about regularities in the data can therefore accelerate the data mining process. Suited techniques for the EDA are of course dependent on the data and the task at hand. Both the choice and the concrete application are usually inspired by any suspicions the researcher has and they allow to bring in explicit knowledge about the data domain.

After EDA we enter the stage of confirmatory data analysis. It consists of choosing or developing an appropriate model (according to the hypothesis gained beforehand) whose parameters are then optimized from the available data. In the case of neural networks, the EDA may help to choose the input features or the network architecture. The neural net is then adapted by applying some learning rule. For optimization or learning, usually the minimization of some error measure is used.

EDA and CDA both go hand in hand and contribute to increase the researchers knowledge. In addition to their application on the data EDA techniques are directly used to monitor the progress or results of machine learning techniques. Results of the CDA step usually refine the researchers perspective. With an updated perspective, it may be useful to repeat from the beginning with a new cycle through EDA/CDA.

Whereas CDA provides quantitative measures, e.g. in terms of an error or quality measure, EDA gives only qualitative indications. This makes it difficult to compare different EDA techniques. The perception of patterns is subjective and depends on the user's perceptual preferences. Assessment of the use of different EDA techniques can therefore only be done by psychophysical experiments. For the comparison of EDA techniques observables like the required time to perform a task and the error rate are of interest which can be measured during such experiments. Questionnaires about individual preferences or the stress during the task might be further used. It can also be of interest to compare mental fatigue and learning effects. But such comparisons and validation of EDA techniques fall into the range of psychology and are out of the scope of this work. However, some qualitative remarks will be made reflecting the author's experiences and comments of colleagues.

As pointed out before, this thesis concentrates on EDA techniques. This chapter will provide an overview of previous research in the field of Exploratory Data Analysis. At the beginning some few remarks shall be made about the data under consideration, followed by a brief overview of visualization techniques. In the last section some data transformation methods will be introduced in more detail, whose results are typically presented by a 2d plot. With each method will be pointed out how it can be applied to convert data into suitable input of a sonification method.

2.1 High-Dimensional Data

This thesis deals with the analysis of a certain type of data given by high-dimensional datasets. A dataset consists of a set of N data records $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{id})^{\mathrm{T}}$ of numbers. Sometimes, the records can be divided into input and output variables: typical output variables are class labels or function values. If output variables are given we talk about supervised learning problems, otherwise the data are subject to unsupervised learning. The dataset can be represented by a set of vectors $\mathbf{x}_i \in \mathbb{R}^d$ in a d-dimensional Euclidean vector space.

It is difficult for humans to directly detect patterns in high-dimensional data since our perceptual senses are not optimized to operate in such data spaces. One of the most basic data representation is a table of numbers. Perception of patterns in this form of presentation is difficult and more or less limited to a columnwise inspection of data values. Even a simple graphical representation like pairwise scatter plots of two variables greatly improves the inspection of the data but it does not reveal any relationships among more than two variables. For the interpretation of graphical presentations we implicitly assume the kind of geometrical relationships that we are familiar with from our (low-dimensional) everyday experiences. Unfortunately, some geometrical assumptions do not hold true in higher-dimensional settings, especially concerning assumptions about local neighborhoods. This mismatch is referred to as the *curse of dimensionality* [Sco92]. Some of the problems which typically occur in high-dimensional data-spaces are:

Visualization: high-dimensional data can not be visualized without loss of information. Scatter plots either have to drop certain axes or project the data onto a manifold of lower dimension (usually 1 or 2).

Empty-space phenomenon: high-dimensional data spaces are in general only sparsely filled with data: consider 1000 data points are given in a 20-dimensional space: the min./max. values of each variable define a cuboid. Splitting each dimension in just two halves, the 20d cube consists of $2^{20} \approx 10^6$ cubes. Even if each data point falls into a different cube, more than 99.9 % of the data space remains empty.

Nonlinear dependencies: the data points of a dataset are often found on a lower-dimensional sub-manifold, i.e. a nonlinear hypersurface embedded in data space. Whereas linear dependencies are easily determined, it is difficult even to imagine nonlinear dependencies. Uncovering the topology and shape of such manifold is one of the aims of data mining [Fri95].

Intrinsic dimensionality (ID): high-dimensional data often can be well described by a nonlinear latent variable model [Mei00] $\mathbf{x} = \mathbf{f}(\mathbf{z}) + \mathbf{u}$ where $\mathbf{z}$ is a random vector of much lower dimension q than the data, $\mathbf{u}$ is random noise and $\mathbf{f}(\cdot)$ defines a q-dimensional surface in data space. In physical systems, $\mathbf{f}$ models the dependencies between the observed variables and $\mathbf{u}$ models noise due to measurement. However, if the noise is small, a q-dimensional latent variable vector suffices in principle to explain the data quite well and q will be referred to as the ID of the data (see [VD95, BS98] for details). A PCA might find a higher data dimensionality as the surface can be nonlinear. Usually local estimators for ID are used, like local PCA [FO71] or Optimal Topology Preserving Maps (OTPM) [BS98].

Geometry: "local" neighborhoods are "empty", "non-empty" neighborhoods are not "local". E.g. consider a spherical Gaussian distribution given by $p_d(\mathbf{x}) = (2\pi)^{-d/2}e^{-\mathbf{x}^{\mathrm{T}}\mathbf{x}/2}$ in a d-dimensional data space. Outside the sphere with $p_d(\mathbf{x}) > 0.01p_d(0)$, that is within the tails of the distribution, the probability mass increases with d. For $d = 1$, less than 0.3 % of the data are in the tails. For $d = 20$, already more than 99.8 % of the data are in the tails. Most data points are found "far outside"! The neighborhood of the origin is "empty". Firstly, this contradicts our expectation to find most data points close to the point of maximum density. Next, this means that local averages computed over nearest neighbors (usually done to reduce the variance) will increase the bias due to the large neighborhood [Sco92].

Integration: Integration over high-dimensional functions gets difficult. For many statistical assessments however, it is necessary to integrate over a distribution. Monte-Carlo integration then replaces the "classical" integration.

An important kind of structure in high-dimensional data is a clustering of the data. Therefore many algorithms exist to fit mixtures of Gaussians or other multimodal distributions to the data in order to identify clusters. There are also a lot of graphical tools for detecting such clusters by generating two-dimensional graphical data presentations which aim at maintaining the clustering structure in the data. Such techniques will be discussed in Section 2.3. Later, they will also be applied to perform the same task for an auditory display.

For the sonification techniques presented in this thesis, mainly synthetically rendered datasets are used as examples. This offers several advantages: the structure within the data is known and can be controlled. Furthermore, the dataset size and data dimensionality can be changed according to the needs. Some frequently used datasets are:

- d-dimensional noisy circle dataset: points are generated by the random vector

$$\mathbf{x} = \cos(\phi)\mathbf{a} + \sin(\phi)\mathbf{b} + \mathbf{u}, \tag{2.1}$$

  where ϕ is a random variable with uniform distribution over the interval $[0, 2\pi]$ and $\mathbf{u} \sim \mathcal{N}\{0, \sigma^2\boldsymbol{I}_d\}$. $\mathbf{a}$ and $\mathbf{b}$ are two arbitrary orthogonal unit vectors.

- 2d-noisy spiral dataset: points are generated by the random vector

$$\mathbf{x} = r(\cos(\phi), \sin(\phi))^{\mathrm{T}} + \mathbf{u}, \tag{2.2}$$

  where ϕ is a random variable with uniform distribution over the interval $[0, 2\pi k]$ (k is the number of rotations), the radius is $r = \text{const} \cdot \phi$, and some random noise $\mathbf{u} \sim \mathcal{N}\{0, \sigma^2\boldsymbol{I}_d\}$ is added. In a 3d-noisy spiral, x_2 is set proportional to ϕ.

- tetrahedron cluster dataset: a d-dimensional tetrahedron is a set of vertices $\{\mathbf{v}_1, \dots, \mathbf{v}_{d+1}\}$ in $\mathbb{R}^d$, so that $\|\mathbf{v}_i - \mathbf{v}_j\| = 1 \; \forall i \neq j$. Points of the tetrahedron cluster dataset are generated by the random vector $\mathbf{x} = \mathbf{v}_k + \mathbf{u}$, where k is a discrete uniform random variable with $\Pr(k = i) = 1/(d+1), \; i \in \{1, 2, \dots, d+1\}$ and $\mathbf{u} \sim \mathcal{N}\{0, \sigma^2 \boldsymbol{I}_d\}$.

In addition, the following real-world datasets are used:

Iris Plants

This dataset is created by Fisher [Fis36]. The dataset contains 3 classes referring to different types of iris plants, namely Iris Setosa, Iris Versicolour and Iris Virginica. For each class 50 instances are given. The dataset has 5 attributes: sepal length, sepal width, petal length and petal width, all measured in cm. The 5th attribute is the class. The main structure is that the three classes cluster, and that one class is linearly separable from the other two whereas the latter are not linearly separable from each other.

Wisconsin Breast Cancer

This dataset contains 699 records with measurements for tumors gathered from microscopic examination. A record is given by 9 attributes like climp thickness, uniformity of cell size and cell cell shape, etc., and a class label which is either benign or malignant. The source for the dataset is [Wol90].

2.2 Visualization Techniques

Many visualization techniques have been invented for the graphical displaying of data. For one-dimensional datasets, histograms, box plots and diagrams (e.g. pie chart) are used [Cle94]. Two-dimensional data are represented by scatter plots or graphs. Specifically for multi-variate datasets, the visualization by Chernoff faces, Andrew curves, parallel coordinate curves or multivariate glyphs is used [dTSS86]. These techniques can be combined in many ways, e.g. locating Chernoff faces as symbols within a scatter plot. Besides that, many specialized visualizations exist whose aim it is to assist certain visualization tasks, e.g. dendrograms [JD88] are used to depict the results of hierarchical clustering, or iso-surface plots for inspecting volume data [ea92].

The most frequently used technique for graphing data is a 2d scatter plot. A recent trend is to develop visualization techniques which use the 2d scatter plot as a visual front end. Such techniques are for example: Principal Component Visualization (PCA) , Projection Pursuit (PP), Self-Organizing Maps (SOM), Multidimensional Scaling (MDS). The objective with such visualization techniques is to find suitable methods to transform the data to a 2d space under the constraint that some structure of interest is maintained during the transformation. In MDS, e.g. the aim is to be truthful with respect to the pairwise distances between data points. The same strategy is later applied to the Parameter Mapping Sonification, the analogue to scatter plots in the auditory domain.

For the design of auditory display, it is helpful to learn from the visual counterpart by analyzing the elements of a visualization and to classify their role within the display. The following discussion is the basis for Chapter 9, where it will be considered what elements can be realized in auditory data display and how this can be done. This is going to be performed on the 2d scatter plot, summarizing its elements and their function within the display.

2.2.1 Basic Elements of Graphing Data

Figure 2.1 is a typical scatter plot in this case showing the time series of the yearly number of sunspots. The plot consists of the following elements:

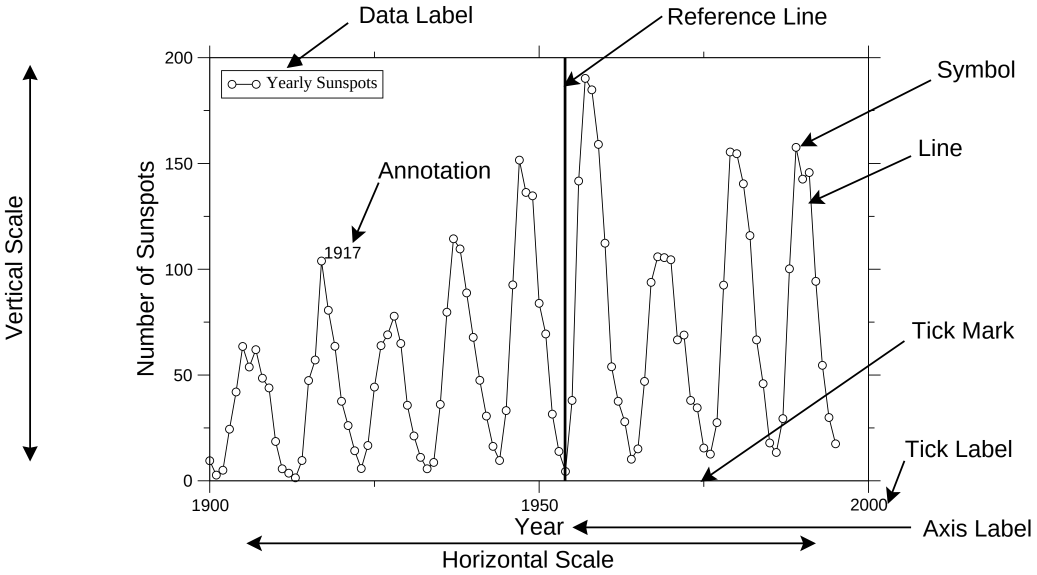


Figure 2.1: Scatter Plot showing the dataset of yearly counted sunspots (sunspot data source [Ton91]). This plot illustrates the basic elements of a graphical data display.

Axis/Scales: Vertical and Horizontal Scale provide basic orientations and directions of data values growth. They tag two at most orthogonal directions to be used for different variables. They further limit the perceptual field for data presentation.

Axis Labels: The function of axis labels is to associate an attribute of the data to an axis. The information is symbolic.

Tick Marks: They define reference points on a scale. Tick marks are integrated to define the position of certain values on a scale. They are mainly used for associating data values with data points by visual interpolating the data point coordinate between the neighboring ticks.

Tick Labels: They assign unique variable values to the tick marks.

Graphs: are used to represent the data. They can use various elements: symbols, connecting lines, bars, error bars and data point annotations. Multiple graphs can be presented in one plot so that they can be compared easily.

Symbols: In a scatter plot, each data point is represented by a visual marker, a symbol. Coordinates, color, shape and size are the most important attributes which can be controlled by the data. If several graphs are represented it is of advantage to use different types of symbols to make the graphs distinguishable. However, a set of symbol-types can also be applied for the presentation of a single categorical variable of a dataset. Color, shape and symbol size can both be applied to carry categorical and continuous variable values. Chernoff faces are an example for complex symbol shapes derived from data [dTSS86]. However, symbols require display space, which limits the spatial resolution. In addition, several symbols may occlude each other and this can impair visual perception.

Lines: Lines are connections between symbols expressing a topological neighborhood in data thus allowing to interpolate between data. Line thickness and pattern are also sometimes data-driven, i.e. used to represent data variables.

Data Label: It associates a dataset with a given graph. This information is symbolic.

Reference Elements: They provide a reference to facilitate comparison or interpretation. Often reference lines are used, but also areas can be superimposed in the plot.

Different information can be withdrawn from a plot, depending on the kind of visual inspection which is determined by the task at hand:

Structural Perception: here, the plot is inspected to find the coarse structure, e.g. clustering of data, dependencies, or periodicities as shown in the plot above. In structural perception, some of the presented basic elements like ticks, labels, etc. are not being used.

Searching: in this task, specific properties are looked for in the graph, whose visual shape is known or expected. In the plot shown above, we can search for the highest peak or find outliers.

Analytical Perception: focuses on individual elements of the plot and aims at perceiving quantitative information and to answer detail questions. From the plot above, we can answer how many sunspots there have been in 1970. This kind of plot inspection makes explicit use of all textual elements, tick labels, etc.

Two important functions of graphing data shall be stressed further: on the one hand, we are able to integrate hypothesis in visual form into a graph, e.g. by inserting trend lines, grouping subsets of the symbols by drawing a line, etc. On the other hand, we can easily communicate with others about a plot, because it is simple to point onto specific elements. Another advantage of graphs is that they can easily be embedded into publications whereas an audio output normally can not be embedded.

2.3 Projection-Based Visualization

The techniques presented in this section illustrate a trend in modern visualization research: to plug-in computational methods which optimize a criterion in order to find a compromise between information loss and maintenance of structure while reducing the data dimensionality in a way, that is necessary to present the data, e.g. to plot the data in a 2d display. The techniques applied to reach this dimensionality reduction are inspiring for two reasons: firstly, they may be applied with minor modifications to a different output modality like auditory displays; secondly, their analysis provides an insight into what information is relevant and what information is lost by the data transformation. The “lost” dimensions are potentially interesting candidates to be presented by sonification.

2.3.1 Projection on Principal Components

Principal Component Analysis (PCA) [Jol86] provides a standard technique for dimensionality reduction by projection onto the q-dimensional linear manifold which maximizes the projected variance. Therefore, a given data matrix $\mathbf{X}$ is first transformed by a shift so that the dataset mean (the column mean of $\mathbf{X}$) is 0, then the coordinate system is rotated so that the first axis is aligned to the direction of largest variance in projection space, successive basis vectors being uncorrelated to the former but fulfilling the same maximum variance requirement w.r.t. the remaining variance. Expansion of the data samples by the first q principal components thus can deliver a reconstruction which captures as much of the data variance as possible with a q-dimensional linear model, and so with the minimal reconstruction error using the L_2 error norm [Rip96].

On the computational side, PCA is usually done by a singular value decomposition (SVD) of the data matrix

$$\mathbf{X}_o = \mathbf{X} - \frac{1}{N}\mathbf{1}_{N\times N}\mathbf{X} \tag{2.3}$$

which is the dataset shifted to zero mean. SVD computes a matrix $\mathbf{U} \in M(N \times d, \mathbb{R})$ with orthogonal column vectors, an orthogonal matrix $\mathbf{V} \in M(d \times d, \mathbb{R})$ and a diagonal matrix $\mathbf{\Lambda}$ with decreasing non-negative diagonal elements $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d \geq 0$, which fulfill

$$\mathbf{X}_o = \mathbf{U\Lambda V}^T \; . \tag{2.4}$$

Thus the columns of $\mathbf{X}_o\mathbf{V} = \mathbf{U\Lambda}$ contain the linear projections, and the row vector $\mathbf{V}_{*i}$ contains the ith principal direction. Computing the dataset covariance matrix allows additional insight into the connection to an older perspective on PCA: the data covariance matrix $\mathbf{C}$ is given by

$$\mathbf{C} = \frac{1}{N}\mathbf{X}_o^T\mathbf{X}_o = \frac{1}{N}\mathbf{V\Lambda U}^T\mathbf{U\Lambda V}^T = \frac{1}{N}\mathbf{V\Lambda}^2\mathbf{V}^T = \mathbf{VDV}^T \tag{2.5}$$

with $\mathbf{D} = \mathbf{\Lambda}^2/N$. Thus, SVD leads to the same result as the eigendecomposition of the dataset covariance matrix $\mathbf{C}$. Furthermore, this shows that singular values λ_i are related to the eigenvalues of $\mathbf{C}$ by $d_i = \lambda_i^2/N$. Figure 2.2 illustrates a 10d noisy circle dataset (see p. 9). PCA uncovers the otherwise invisible structure because the circle axes carry the highest variance.

The dimension may be reduced by choosing the smallest $q < d$ such that

$$\frac{\sum\limits_{i=1}^{q} \lambda_i^2}{\mathrm{Tr}(\mathbf{\Lambda}^2)} > 90\% \; . \tag{2.6}$$

This provides a dimension reduction while retaining more than 90% of the variance (“information”). Using the reconstruction

$$\tilde{\mathbf{X}} = \mathbf{U\Lambda}_q\mathbf{V}^T \tag{2.7}$$

where $(\Lambda_q)_{ii} = \lambda_{ii} \; \forall \; i \leq q$, it can easily be seen that the mean squared error (MSE) equals the sum of the variance in all unconsidered components [Rip96]

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}\|\mathbf{x}_i - \tilde{\mathbf{x}}_i\|^2 = \frac{1}{N}\|\mathbf{X} - \tilde{\mathbf{X}}\|_F^2 = \frac{1}{N}\|\mathbf{\Lambda} - \mathbf{\Lambda}_q\|_F^2 = \frac{1}{N}\sum_{i=q+1}^{d}\lambda_q^2 \; , \tag{2.8}$$

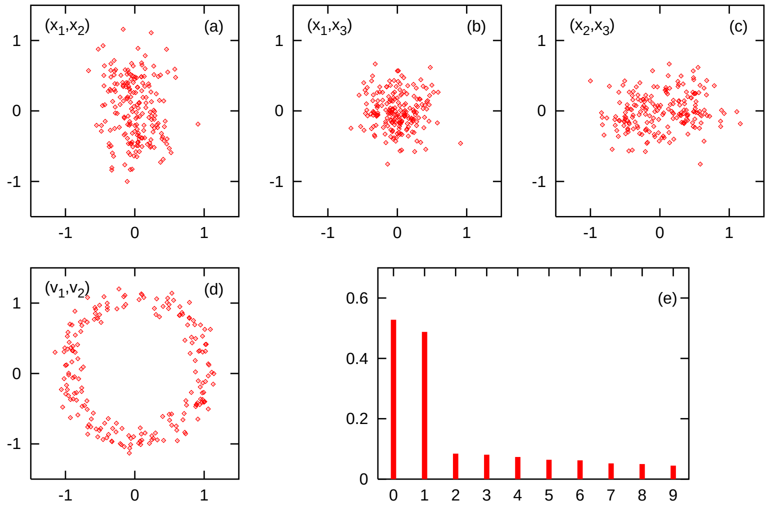


Figure 2.2: Illustration of Principal Component Analysis for a 10d noisy circle dataset with 200 points and $\sigma^2 = 0.1$. (a-c) show some random 2d projections which obviously hide the circle structure. Projection onto the first two principal components shown in (d) allows to detect the structure. (e) shows the eigenvalue spectrum.

where $\|\cdot\|_F$ is the Frobenius norm.

Thus, PCA is especially useful for dimension reduction, if the eigenvalue spectrum shows a sharp decay after few dimensions. Most applications for visualization use the first 2 or 3 components for a scatter plot visualization. The presentation above shows the main weakness of PCA: as the only criterion is variance, PCA depends critically on the units of the dimensions. Unless there is a strong evidence in the data for the comparability of the dimensions, it is recommended to rescale all dimensions to have variance 1. However, PCA still is a useful transformation in this case, as the variance rescaling does not remove the correlations between dimensions. Removing the obviously redundant dimensions should always be the first step in exploratory data analysis, as the goal is to present the data by suited visualizations and sonifications which both have limited dimensionality. For further investigation, variance is no longer of importance and the information can be removed by a 'sphering transformation', which in this case means to scale the retained columns of $\mathbf{U}_{N\times q}\mathbf{\Lambda}_q$ to have variance 1. This is easily done by simply taking $\mathbf{U}_{N\times q}$ as new dataset.

In Chapter 8, auditory displays which use PCA tranformed datasets for combined visualization and sonification will be presented. In Chapter 9, PCA is utilized for data preprocessing for an auditory display.

2.3.2 Projection Pursuit

While Principal Component Analysis selects the linear projection manifold only under the criterion of variance maximization (and thus no other structural properties influence the choice of manifold), Projection Pursuit (PP) generalizes the manifold selection by the introduction of a structure evaluation measure. PP is a numerical optimization of a criterion in search of the most 'interesting' linear projection of dimensionality q, usually for $q = 1$ or 2 so that it can be visualized. Different measures of 'interestingness' have been considered. In the following, $\mathbf{Y} = \mathbf{XP}$

is the matrix of projection coefficients while projecting $\mathbf{X}$ on the manifold given by the column vectors of $\mathbf{P}$.

The first approach by Friedman [FT74] was an index for an univariate projection that aimed at revealing groupings in the data:

$$I_T = \hat{\sigma}_\alpha(\mathbf{Y}) \sum_{i,j} \max\left(0, h - |\mathbf{y}_i - \mathbf{y}_j|\right) \tag{2.9}$$

where $\mathbf{y}_i$ is the projection of $\mathbf{x}_i$ and $\sigma_\alpha(\mathbf{Y})$ is the estimated α-trimmed standard deviation[1]. Huber showed that this index is just a special case of a more general class of projection indices, defined for a given density $f_P(\mathbf{y})$ in projection space

$$I_H(f_P) = \sigma(\mathbf{y}|P) \int_{\mathbb{R}^q} f_P(\mathbf{y})^2 d\mathbf{y} \ . \tag{2.10}$$

The Tukey index (2.9) follows with a slight modification of σ by clipping the edges and using kernel density estimation with a uniform kernel function $U_{[-0.5,0.5]}$ to compute f [Hub85]. If $q > 1$, then σ is replaced by a product of the marginal standard deviations in projection space. For the following short presentation of different indices, it will be assumed that $q = 1$ and a probability density function $f(y)$ is given in projection space. Further the dataset $\mathbf{X}$ is assumed to be sphered to have covariance matrix $\mathbf{I}_d$. Then the generalized Tukey index can be written

$$I_T(f) = \int f(y)^2 dy \tag{2.11}$$

with integration over projection space.

However, different alternatives have been considered, either to improve the view reached so far or to accelerate computation of the index. As it could be shown, that a randomly selected projection of a high-dimensional dataset appears similar to a sample from a normal distribution [DF84], various indices have been considered which favor departure from normality, like the Friedman index

$$I_{Fr}(f) = \int \frac{[f(y) - \phi(y)]^2}{2\phi(y)} dy \ . \tag{2.12}$$

Other prominent indices are the standardized Fisher information and negative Shannon entropy, given by

$$I_F(f) = \int \frac{f'(y)^2}{f(y)} dy \qquad\qquad I_{Sh}(f) = \int f(y) \log f(y) dy \ . \tag{2.13}$$

After choosing an index, a projection is chosen by numerically maximizing the index. However, as the index shows many local maxima, gradient ascent will rarely lead to good views. Random sampling of projection matrices or starting local optimization from many different starting-points can help to find good views. A dynamic visualization of the rotation of multidimensional datasets is called *the grand tour* and can be computed with the `XGobi` tool [SCB98], which also allows to experiment with various PP indices. Figure 2.3 shows some plots of the cancer dataset.

Within the grand tour, both the time-variate index value and the PP plot are shown visually, but the user cannot pay attention to both displays at the same time. Here, PP indices may be suitable observables for acoustic presentation while the user focuses on the animated plot. An index sonification can e.g. be done by varying the pitch of a time-variant oscillator (see Section 6.2.1) dependent on the index value.

[1] $\alpha/2$ of the data are deleted on each side of the 1d-distribution.

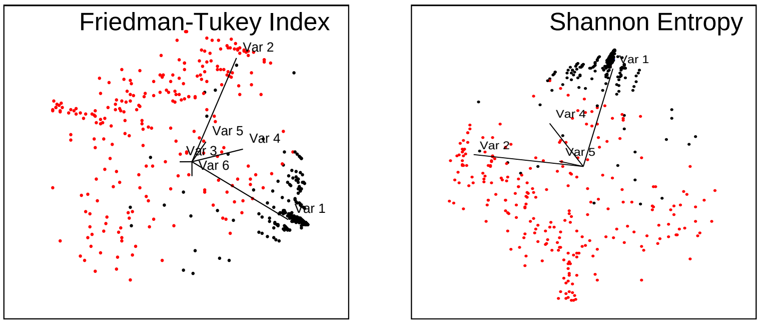


Figure 2.3: Projection Pursuit Visualizations for the 9d cancer dataset (see p. 10). Shown are two projections that (locally) maximize the Friedman-Tukey index and the Shannon Entropy Index. Var i indicates the direction along the ith principal component.

2.3.3 Self-Organizing Maps

So far, visualizations of high-dimensional data have been demonstrated that base on projections on linear manifolds. The Self-Organizing Map (SOM) [Koh82] offers an alternative way to visualize data, to reduce their dimensionality and to cluster them at the same time. A regular grid of neurons (mostly in one or two dimensions) is provided with stimulus vectors from data space. The neurons' weight vectors might be regarded as points in the data space. The adaptation algorithm updates all neurons within a grid neighborhood σ of that neuron whose weight vector was nearest to the stimulus. The idea of the SOM is to get a low-dimensional mapping of the data to the neuron grid so that data which are closely in the data space are represented by neurons which are closely on the neuron grid.

In contrast to linear projection techniques the mesh formed by the topologically ordered neurons (weight vectors) approximate a nonlinear manifold of the grid dimensionality.

Given a q-dimensional grid of neurons (prototype vectors) $\mathbf{w}_r$, topologically ordered by their index vectors $r = (r_1, \ldots, r_q)$, $1 \leq r_j \leq N_S$, the SOM algorithm starts with a random initialization of the prototypes $\mathbf{w}_r$. A learning rate ϵ and a neighborhood radius σ are initialized. During adaptation data points are randomly selected and used to update the SOM neurons. The neuron s with minimum distance to the data point $\mathbf{x}$ is looked for:

$$s = \arg\min_r(\|\mathbf{w}_r - \mathbf{x}\|) \; . \qquad (2.14)$$

All prototypes within a σ-neighborhood of s are updated by

$$\mathbf{w}_r^{new} = \mathbf{w}_r^{old} + \epsilon h_\sigma(r,s) \cdot (\mathbf{x} - \mathbf{w}_r^{old}) \qquad (2.15)$$

where h is a kernel function which in the case of the 1d-SOM can be chosen as

$$h_\sigma(r,s) = \exp\left(-\frac{(r-s)^2}{2\sigma^2}\right) \; . \qquad (2.16)$$

Beginning the adaptation loop with rather large $\sigma \approx 0.7N_S$ and slowly decreasing both σ and ϵ leads to an organization of the neurons in data space which may be described as topologically ordered. Furthermore, the neuron density in data space approximates the data density [RMS92]. Thus, a higher resolution is provided in those parts of the data space where more data are found.

Although the SOM algorithm is rather heuristic and a theoretical analysis is made difficult by the lack of an energy function [EOS92] or an optimality criterion, the SOM is widely used in many data mining tools (e.g. ASOC Sphinx Vision[2], Neo/NST [Rit00], SPSS Clementine[3]).

There is a variety of ways to visualize datasets by using SOMs. For educational purposes it is nice to look at the neuron mesh in data space, which can for instance be plotted by projecting both the dataset and the SOM neurons weight vectors onto the first principal components of the data. Figure 2.4 shows different adaptation steps of a 2d-SOM, trained on a two-dimensional distribution. Within the dashed square, the density is twice that of the outer region and as a consequence more neurons are drawn into this area. In (c) an 1d-SOM unfolded into a curved distribution. Here, it can be seen that with ongoing adaptation, the SOM may overfit the data. If the SOMs are used for visualization, in most cases two-dimensional network topologies are used. The SOM grid is mapped to the display coordinates and the data points are attached to their corresponding nearest neuron. Figure 2.5 shows 2 different visualizations which base on this visualization technique. Other SOM visualizations use the z-axis of a plot to show a specific component of the weight vectors. For binary feature components as shown in Figure 2.6 it can be seen where the attribute values change.

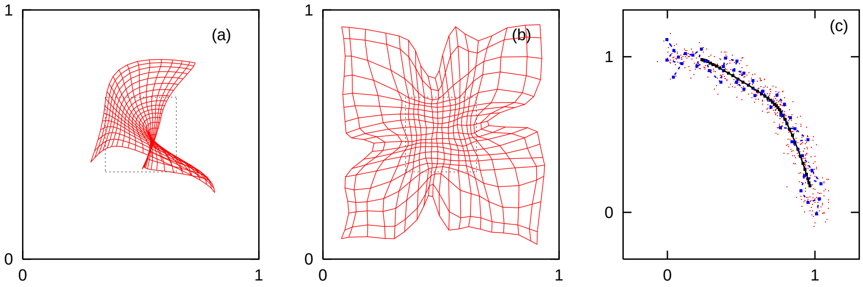


Figure 2.4: Visualization of SOM mesh in data space - a 20 × 20 SOM was used to adapt to a distribution, where the density is uniform but twice as high within the dashed square shown in the plot. (a) shows the net after a few iterations where the neighborhood σ is still large. (b) shows a later step. It can be seen that the SOM allocated more neurons in the region of higher density. (c) shows the same for a 1d-SOM adapted to a “curved” one-dimensional distribution with added noise. The black line is an intuitively suited approximation, the dashed blue line shows that SOMs tend to overfit the data with ongoing adaptation.

The SOM can be used for all sonification methods developed in this thesis which make use of auditory maps. A direct browsing technique for auditory neuron maps applies the mouse pointer to probe certain neurons. The weight vector of the selected neuron is then used to specify a point where for instance a sonification model is excited. Sonification models and possible excitations are introduced in Chapter 7.

Another topic is the sonification of SOM learning. The adaptation process can be sonified by playing every adaptation step as a tone whose acoustic attributes are determined by the neuron’s position, its topological environment and its learning step size. Another interesting observable may be the neurons average distance to its topological neighbors. From such a sonification one can expect to perceive the SOM dynamics and its elasticity. However, this SOM-adaptation sonification has not been realized so far.

[2] http:///www.asoc.de

[3] http://www.spss.com/clementine

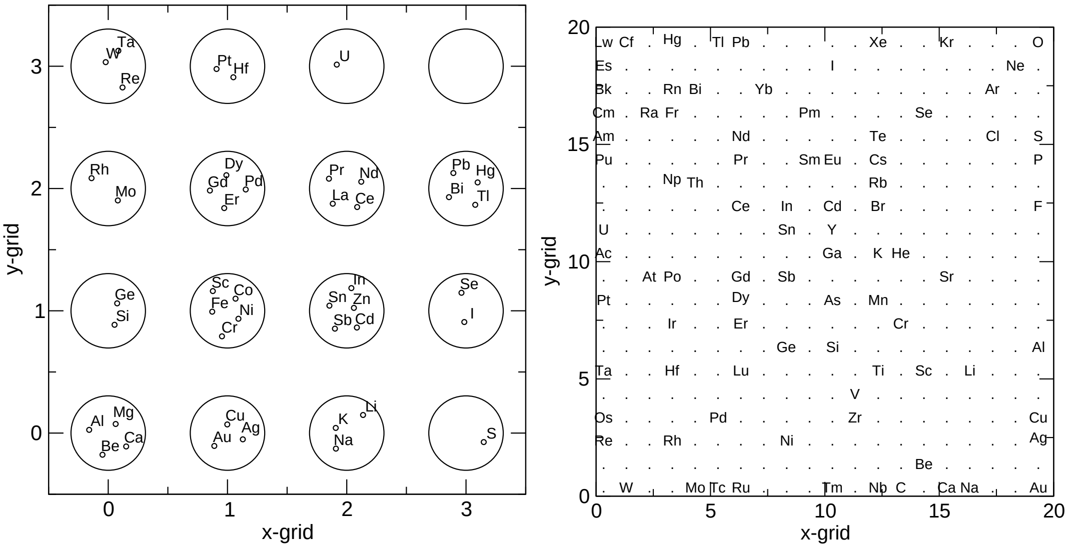


Figure 2.5: Visualization with 2d-SOMs. In (a), a 4×4 SOM is used to visualize a dataset with some chemical elements. Features are melting point, boiling point, density, specific heat and others. From this data, the SOM is able to uncover similarities. It groups alkaline metals, noble metals, etc. Each neuron is represented by a circle located at its grid coordinates. All data points are mapped to their nearest neighbor neuron and plotted within the circle at a random position. In (b), a 20×20 SOM of the same dataset is shown. The label of each data point is inherited to the neuron with least distance to the data point.

2.3.4 Multi-Dimensional Scaling

Multidimensional scaling (MDS) is a technique for computing low-dimensional representations from a dissimilarity matrix [CC94]. Dissimilarities are non-negative numbers d_{rs} which provide a measure about how distant data $\mathbf{x}_r$ and $\mathbf{x}_s$ are. In the case of a given dataset $\mathbf{X}$ in d-dimensional Euclidean space, a dissimilarity matrix can be constructed by $d_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|$. However, MDS also allows to compute low-dimensional representations when the dissimilarity matrix is not produced by a metric and thus does not satisfy the triangle inequality $d_{ij} \leq d_{ik} + d_{kj}$.

In this thesis, a special MDS technique is used called Sammon mapping [Sam69]. Given a d-dimensional dataset $\mathbf{X}$, the Sammon map aims at finding a low-dimensional projection $\phi : \mathbb{R}^d \to \mathbb{R}^q,\ q < d$, which minimizes the cost function

$$E_\phi = \sum_{i<j} \frac{(d_{ij} - \tilde{d}_{ij})^2}{d_{ij}} \tag{2.17}$$

where $\tilde{d}_{ij} = \|\phi(\mathbf{x}_i) - \phi(\mathbf{x}_j)\|$. A suited starting configuration for MDS is given by the projections of the data onto the first q principal components. Minimization of E_ϕ then is done by a gradient descent, e.g. by moving randomly selected points $\phi(\mathbf{x}_j)$ to a position in $\mathbb{R}^q$ which reduces E_ϕ. As a result only a local optimum can be found. This energy function favors configurations $\{\phi(\mathbf{x}_1), \ldots, \phi(\mathbf{x}_n)\}$ where the representation is more accurate with small distances than with large distances. For the purpose of visualization, usually $q = 2$ is used and the final configuration is plotted in a scatter plot. Figure 2.7 shows an MDS visualization for the 9d tetrahedron cluster dataset (see p. 10) with 10 clusters, which the Sammon map nicely separate.

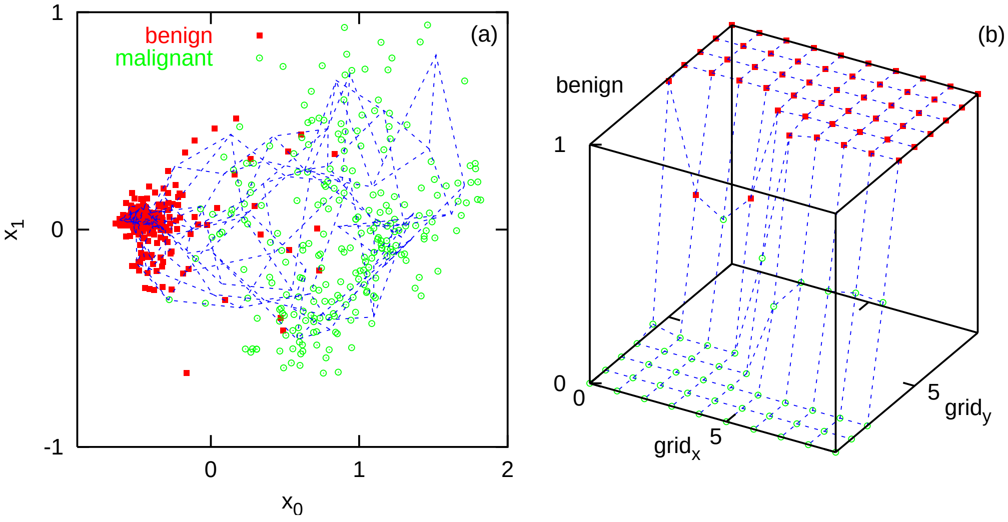


Figure 2.6: 3d-visualization of a 2d-SOM by mapping an feature (component of the neurons weight vector) to the z-axis of the display. Here in (a) the cancer dataset is shown. The dashed mesh is the projection of the SOM. In (b) it can be seen that benign and malignant regions are assigned to topologically neighbored areas of the neuron ensemble.

In this thesis, MDS scaling is used to compute a time-index for sound events in an auditory display that uses Parameter Mapping. The sonification will be discussed in Section 9.2.

2.3.5 *Principal Curves*

Regarding datasets which are sampled from high-dimensional distributions in 'real world situations', it is often found that the intrinsic dimensionality[4] is much lower than that of the embedding data space. Therefore PCA is usually the starting point to achieve some dimensionality reduction. This approach can be motivated from a minimization of a cost function which measures the squared distance of the data w.r.t. a linear manifold. The data are projected onto a subspace to achieve a dimensionality reduced representation, which captures the main variation of the data. However, while maintaining the dimension of the projection space, the dataset can often be much better represented by allowing a higher flexibility of the projection manifold. This can be achieved by using a nonlinear model. Principal curves are continuous one-dimensional manifolds that approximate the data in this sense and thus pass through the "middle" of a d-dimensional dataset. PC's have been successfully applied to solve practical problems, like the alignment of magnets of the Stanford linear collider [HS89] or Ice Floe Identification in Satellite Images [BR92]. Recently PC's have also been presented within the framework of statistical learning theory [KKLZ00, SMS89].

Let $\mathbf{f}(\lambda)$ be a parameterization of a curve. All d components $f_i(\lambda)$ are continuous functions of a single variable λ, which parameterizes the curve. A unique and natural parameterization can be given in terms of the arc length. A projection index $\lambda_f : \mathbb{R}^d \to \mathbb{R}$ can be defined by

$$\lambda_f(\mathbf{x}) = \max_{\lambda} \left\{ \lambda : \|\mathbf{x} - \mathbf{f}(\lambda)\| = \min_{\nu} \|\mathbf{x} - \mathbf{f}(\nu)\| \right\}, \tag{2.18}$$

[4] see p. 9.

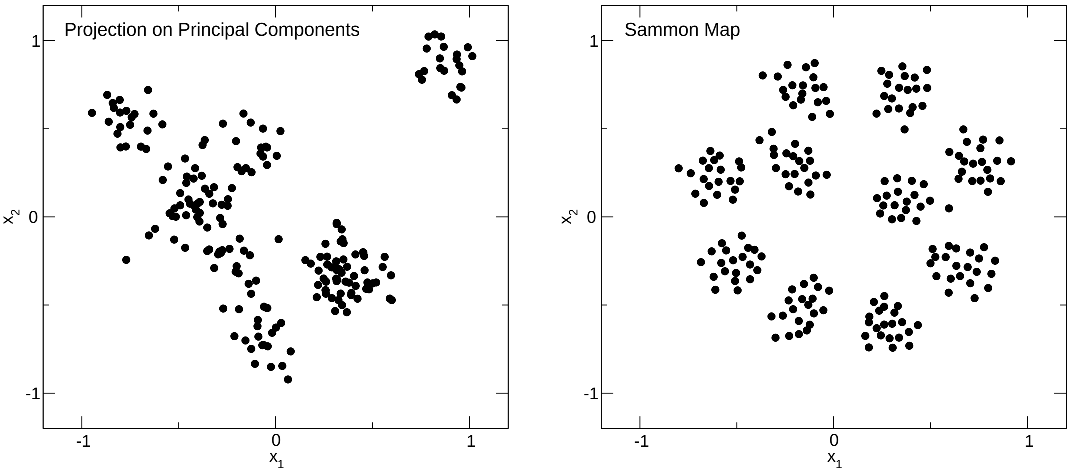


Figure 2.7: Illustration of MDS visualizations. A 9d tetrahedron cluster dataset (see p. 10) is used. The left plot shows the first two principal components. Obviously the clustering structure is hidden. The right plot shows the 2d Sammon map of the dataset. The clusters can be resolved although the high symmetry (pairwise cluster distance is a constant) is lost.

which is the largest value of λ for which the curve is closest to the point $\mathbf{x}$.

Hastie and Stuetzle defined the principal curve of a continuous data distribution $p(\mathbf{x})$ by the self-consistency property [HS89], which says that the mean of all data projecting on a point $\mathbf{f}(\lambda)$ just is $\mathbf{f}(\lambda)$:

$$\mathsf{E}(X \mid \lambda_f(X){=}\lambda) = \mathbf{f}(\lambda) \ \ \forall \ \lambda \, , \tag{2.19}$$

where X is a d-dimensional random variable. Normally the continuous probability distribution is unknown, since only a finite sample of that distribution is available. Unfortunately this definition of the principal curves cannot be carried over to discrete datasets without modification. The reason for this is that in general all data points project on different λ of the PC, so that arbitrary curves which pass through all data points fulfill equation (2.19). Such curves, however, are not what is intended to be a PC. We rather wish for a smooth fit of the data. The problem can be solved by introducing a regularization constraint, which allows to control the complexity of the principal curve and thus prevents the curve from overfitting the data. This can e.g. be done by restricting the length of the curve [KKLZ00] or by a smoothness constraint [HS89], which is done by estimating local means over a suitable neighborhood of $\mathbf{f}(\lambda)$.

The Principal Curve Algorithm

In this work a similar algorithm like that of Hastie and Stuetzle [HS89] is applied to compute the PC of a dataset. For computational simplicity, the PC is represented by polygonal lines with a fixed number of vertices and thus completely given by the ordered set of vertex coordinates. Figure 2.8 illustrates Principal Curves for different curve smoothness for a 2d distribution. Obviously, the PC is capable to catch the main variation of the data.

Initialization of the PC is done by setting the vertices equally spaced along the first principal axis, between the minimal and maximal projection indices of the dataset. For computation of the PC, the vertex coordinates are iteratively updated in cycles of projection steps and expectation steps.

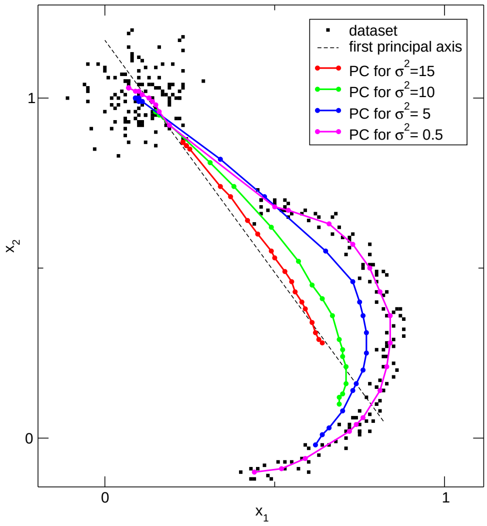


Figure 2.8: (a) Principal Curve of a 2d toy dataset with data points in a Gaussian cluster and on a noisy half-circle. The dashed line shows the 1st principal axis. The PCs are represented by 30 vertices. PCs for different values of smoothness parameter σ are shown.

The **projection step** computes the projection index $\lambda_f(\mathbf{x}_i)$ for all data points $\mathbf{x}_i$ according to definition (2.18). In addition the distance to the curve and the projection point coordinates are computed. To monitor the degree of approximation, the mean squared distance to the curve is computed both for the training set and an independent test set of the same distribution.

In the **expectation step**, the vertices are updated to reduce the empirical error. This is done by shifting all vertices to the mean of the subset of data points that project to their neighborhood. Computationally this is done by local kernel regression on the projection index using a Gaussian kernel with bandwidth σ.

If the kernel bandwidth σ is chosen too small, in some cases the data are badly approximated by the curve in a way that can be described as topological disorder. This can be avoided by applying deterministic annealing [RGF90]: starting with a high value of the smoothness parameter σ, it is gradually decreased during optimization. A suitable starting value for σ^2 is given by the largest eigenvalue of the dataset covariance matrix. For large values of σ, the curve is contracted towards a single point at the dataset mean. With decreasing σ, the curve gradually adapts to the data. After each expectation step, σ is reduced by a factor of 0.95 until the mean squared distance in the test set begins to increase due to overfitting. To apply deterministic annealing the projection step of Hastie and Stuetzle was slightly modified in order to allow to project the data also onto a linear continuation of the curve beyond the first and last vertex. This keeps the curve's squared length in the order of magnitude of the data variance and thus avoids a contraction of the PC which otherwise occurs for large values of σ.

The computation time scales with $O(N^2)$. Computation can be severely accelerated by starting with a small random sub-sample of the dataset and also fewer polygon vertices. During adaptation both the sample size and the number of vertices are increased. This modification makes the algorithm well suited for large datasets.

Principal Curves are applied in this thesis as a trajectory along which data are sonified. Principal Curve Sonification is introduced in Section 8.4.

Chapter 3

Sonification – An Overview

3.1 Definition

Sonification is a rather young discipline and both the scope and its definition are still under debate. Therefore this chapter starts with a summary of some definitions of sonification, which also enlighten essential aspects and properties of a sonification system.

The most accepted definitions are given by Kramer et al. [KWB$^+$99]:

Definition 1: Sonification is the use of non-speech audio to convey information.

Definition 2: Sonification is the transformation of data relations into perceived relations in an acoustic signal for the purpose of facilitating communication or interpretation.

Definition 1 restricts sonification on the use of non-speech sound to discriminate it from speech interfaces. However, speech can be a valuable element in auditory displays as it is able to provide annotations or explanations for other acoustic entities without changing the media. Furthermore, there are many prosodic attributes in speech like pitch, articulation, roughness, accentuation which are suited to be driven by data. Such data-driven use of speech-like sounds should also be called sonification.

Definition 2 stresses the purpose of sonification: the communication or interpretation of data in any given domain of study. This mainly discriminates sonification from data-driven music composition where the primary intent is different: with data-driven music composition, the sound is the product, whereas the linkage to the data is a by-product, thus the main concern lies onto the sound and on its (mostly emotional) effect onto the recipient and not in learning something about the data.

Another working definition is given by Scaletti [Sca94]:

Definition 3: Sonification is a mapping of numerically represented relations in some domain under study to relations in an acoustic domain for the purpose of interpreting, understanding, or communicating relations in the domain under study.

This definition seems to be very similar to definition 2. The difference is that this definition restricts the sonification techniques to a mapping. Thus definition 2 is more general.

Starting from Scaletti's definition, Barrass [Bar97] developed the concept of Auditory Information Design. He distinguishes two parts in Scaletti's definition: *information requirements*, which is about specifying information that is useful for a task at hand) and *information representation*, which is about the display of information. Considering the relevance of the task (either understanding, interpreting or communication) for the design of an auditory display, he presents the following definition:

> Auditory Information Design is the design of sounds to support an information processing activity, focusing on the specific task like interpreting, understanding or communicating relations in the data.

Summarizing, there are two requirements for a sound to be called a sonification:

- the sound is synthesized depending upon the data of the domain under study, and
- the intention for generating the sound is to learn something about the data by listening to it. The sound is only regarded as the medium of communication.

In this thesis, any sound which fulfills these requirements is called a sonification. Speech sounds may also be used within sonifications where appropriate, and from a conceptual level no restrictions are made to the sound rendering technique.

3.2 Research field of Sonification

Sonification is a very interdisciplinary research field. Figure 3.1 shows a typical information flow in a sonification system. Several research disciplines contribute to the implementation and understanding of the involved processes. For the development of a sonification system, it must first be understood what data are available and what the measurements mean. Furthermore there may be domain specific dependencies among the variables and therefore domain expertise helps in designing a sonification system. For sonifications with a rather simple purpose like alarm systems or enhancement of graphical user interfaces, no further knowledge is required at this point. In case of high-dimensional data to be sonified, however, statistics and data mining contribute techniques for an intelligent data preprocessing, e.g. for dimensionality reduction. The discipline of human computer interaction (HCI) is concerned with many aspects of sonification systems. HCI topics include design guidelines for tools, human information processing, ergonomics, system design and usability. HCI contributes valuable insights into how such topics may be analyzed and evaluated. Computer Science contributes to the realization of a sonification system in different aspects: software engineering copes with how to program the interface and how to implement the rendering of the sonifications from the data, signal processing provides techniques to manipulate sound signals. The fundamentals of sound generation are focused in Acoustics which is a part of Physics. Examining the physics of sound generating processes can be inspiring for the selection of sound synthesis techniques to represent data. For Model-Based Sonification (see Chapter 7), Acoustics can supply templates for a model setup and the dynamics. As a result of sonification rendering, a digital representation of the sound is produced. The transformation from this digital representation to real sound waves in air is performed by sound cards or synthesizers, amplifiers and loudspeakers. Depending on the needs, solutions from a mono loudspeaker system to complex multi-speaker arrays for high-resolution spatialization of the sound are used. Sound Engineering is concerned with the technical realization and the sound signal changes due to reflections in the listening room. The disciplines physiology and neurobiology are concerned with the processing of the sound signal after it reaches the human ear. They help to understand how the signal is processed within the ear and what a listener is able to discern from a biological perspective of signal processing. Psychology, Psychoacoustics and Auditory Perception are concerned with higher-level perceptual processing which take place in the auditory brain. Many results like

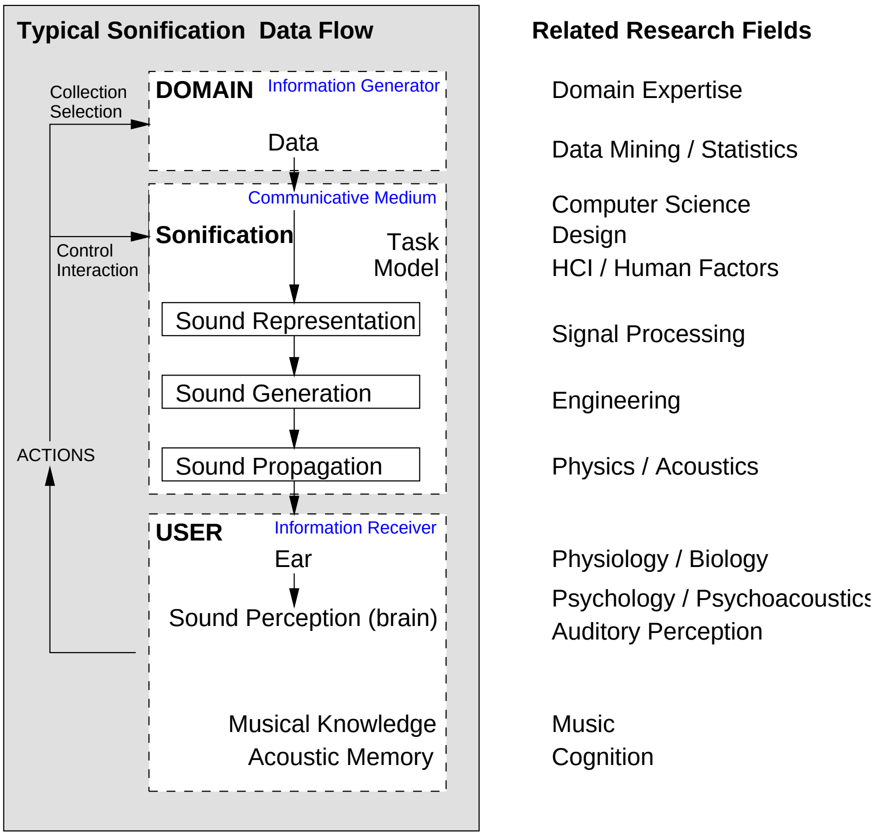


Figure 3.1: Schematic Illustration of the information flow in a typical Auditory Display System. Related research disciplines are indicated on the right side.

the Auditory Gestalt Principles [Wil94] or Auditory Streaming [Bre90] provide guidelines for the usage of sound in sonifications. Musicology contributes to understanding different aspects of sound: it provides a framework to organize acoustic material concerning its rhythm, measure, harmony and it delivers tools for documentation (e.g. a score) and analysis of musical pieces. These techniques can also be applied to control and describe sonifications. Finally cognition focuses on various aspects of the listener like acoustic memory, processing speed for auditory signals and the coupling of sound and emotional states.

This thesis will present some results from the various disciplines but there is too little space for an in-depth analysis. Instead, some guiding links to literature will be provided.

3.3 History of Auditory Display

Sonification is a rather young discipline. Research publications can be found since about 15 years. The first international conference on auditory display (ICAD) [ICA] was held in 1992. To provide an overview of the research field some early examples of auditory display are reported. Auditory Information is used since the early stages of human evolution for many different purposes like alarm, communication or orientation. One point of interest of this section is to make the reader aware of how ubiquitous sound is and for what purposes humans frequently use it even without taking particular notice. Throughout the last 400 years, there was an increase of technical systems that used sound to convey information. These applications might be regarded as the precursors of sonification. In Section 3.3.2 these and more recent auditory displays will be presented to provide an overview of existing sonification systems.

3.3.1 Using Sound to convey Information

One of the most thoroughly investigated technical systems which make use of sound is sonar. Sonar is the usage of sound for locating and recognizing objects under water. A distinction can be made into passive systems, where simply the measurements of directed underwater microphones (called hydrophones) are listened to, and active systems, where the echo of an emitted sound wave is monitored acoustically for the purpose of determining distance, relative motion of underwater objects or to localize swarms of fishes [Ulr67].

Morse code is an example of how sound is used to convey symbolic information. A rhythmical pattern of short and long tones is assigned to each letter. Combination of these patterns allows to communicate complex messages. The sonification techniques of Earcons[1] uses a similar strategy to convey information and can profit from experiences with this kind of auditory display.

The eldest auditory displays used simple acoustic signals to inform the user about a change of state or to signal an event and thus to draw the user's attention on a particular event. The telephone bell or the siren are typical examples for such auditory displays. Currently, a trend towards more elaborated ringing sounds can be observed: mobile phones allow to generate ringing sounds dependent on the caller. This can be called a categorical sonification.

During medical surgeries the heart beat is monitored with a device called pulsoximeter. It measures photometrically the fraction of oxygenized hemoglobin in the blood. This is done by attaching a sensor consisting of a light bulb and a photometric cell on the patient's finger. On every heart pulse, a 'pip' tone is generated whose pitch corresponds to the oxygen level. In addition to this, the pulse speed can be discerned from the rate of the events. The sound helps the anesthetist to judge how the lungs operate. It provides both qualitative and quantitative information about the patient's condition without disrupting the view on the wound. Apart from that we can find many other instruments with auditory alarms in an operating room which are activated if defined parameter ranges are exceeded, e.g. blood conditioning machines, lung ventilator etc.

Auditory displays are frequently used to alarm people, e.g. by siren sounds on fire engines or ambulance vehicles. In airplane cockpits, both alarms and auditory data displays are used. Forbes [For46] conducted experiments on the use of auditory signals for instrument flying, displaying altitude, tilt, velocity and turning of the craft by auditory streams. The "eye-free" nature of auditory displays make them a suited interface in this field of application.

A rather old example of an auditory display is the well-known Geiger counter which provides a direct auditory display of the number of registered ions per time caught by the electrodes of the Geiger device. It is used to measure the radioactivity and allows the listener to infer quantitative information.

The Auditory Thermometer is a device representing temperature by pitch in a continuous tone, thus allowing to signal trends and convergence.

Cardiologists use ultrasonic emitters to examine the heart and its arteries. They use an auditory display to monitor the blood pumping through arteries. The sound is computed by converting the ultrasonic wave to the auditory frequency range. The Doppler effect allows them to conclude from the pitch to the velocity of the blood.

[1] see p. 36.

Neurophysiologists have listened to neurons firing for at least forty years. They use amplifiers and a loudspeaker to position electrodes. Since neurons fire at a rate between 1 Hz and 1000 Hz, the direct playback of a neuron's electrostatic potential results in a pitched audio signal. The sound can be used to identify neurons reacting to a given sensory input as well as to distinguish different categories of neurons.

In many domains audition is used to gain insight to an observed system. Car mechanics listen to the sound of an automobile engine in order to draw conclusions about causes of malfunction and physicians apply their stethoscope to diagnose disease from auscultation (medical listening) to sounds of the lungs, the heart and other parts of the body. During his education the physician learns how to interprete the sound correctly w.r.t. the organ function. These application shows that humans are very well capable of learning to interprete "new sounds" and to use acoustic clues. Learning the meaning of sounds by listening to examples may indicate how tutorials can be designed to teach new sonification-users how to interprete the sound.

In most cultures music is the most common usage of non-speech sound. It is used to strengthen an emotional state (e.g. in movie sound tracks) or to synchronize human activities (march music, aerobics) or simply for enjoyment. However, music in most cases does not present data and therefore it is not regarded as a sonification in the sense of the definition.

In mathematics, statistics and data mining, sound has only rarely been used so far. The aim of this thesis is to fill the gap and provide techniques for the auditory display of empirical data.

3.3.2 First Steps in Sonification Research

This section provides a brief summary of early research results in the field of auditory display and sonification. A more detailed review is found in [Kra94b]. Pioneering efforts were made by Pollack and Ficks (1954) [PF54] who investigated the usage of abstract auditory variables for the presentation of quantitative information. Their display consisted of alternating tone and noise bursts, using attributes like loudness, pitch, relative duration, total duration, stereo location and others as variables. Their studies showed that displays using multiple sound parameters generally outperformed selected unidimensional displays [Kra94b].

Concerned with the effects of receiver operating characteristics of listeners and the effect of training was a study of Speeth (1961) [Spe61] who used audification of seismic data for the classification of events into two groups: seismic records caused by bomb blasts or by earthquakes.

A 3d auditory display as an enhancement for scatterplots was presented by Chambers, Mathews and Moore [CMM74]. They found that sound was helpful for the classification of data but did not conduct any formal testing.

More fundamental research was conducted by Bly (1982) [Bly82]. In her dissertation, she explored the classification of non-ordered multidimensional datasets using Parameter Mapping to represent the data. She experimented with various mappings and training methods and made comparisons of displays either which used only sound or only graphics or were bimodal. She found that in her setup the auditory display was as effective as the visual display, and that the combined display showed best results.

Since the 1980s, research on auditory display increased rapidly and only a few works are presented here. Frysinger, Lunney, Mansur, Mezrich, Bly, and others[2] contributed with more systematic research about perception issues, applications and showed examples on how to extend visual displays by sonification.

[2] See [Kra94b] and references there

With the spreading of microcomputers a trend to integrate audio signals within general computer interfaces began. Gaver's SonicFinder (1985) [Gav89] for the Apple Macintosh is one of the important contributions which have to be mentioned here. He used auditory icons to associate meaning to sound. Gaver, Smith and O'Shea later studied the use of auditory icons in process control with experiments where subjects had to control a simulated Coca Cola plant [GSO91].

In 1989 Stuart Smith developed an integrated tool for both auditory and visual presentation of multidimensional data called Exvis [Smi91]. He used data-driven geometric "icons" for visualization with a visual texture and a pointing device to probe the auditory representation. Kramer contributed with the development of sonification concepts and sonifications for complex systems. He developed the Clarity's Sonification Toolkit [KE91] and started with sonification of data from a 9-dimensional chaotic system.

In 1991 Scaletti and Craig worked on sonification at the U.S. National Center for Supercomputing. Their research was focused on the presentation of time-varying data (pollution data, forest fire data) which they mapped to animated graphics and sound [SC91].

In 1992, the first meeting of the International Community of Auditory Display (ICAD) took place, the first and so far the only conference which is focused to Auditory Display research and sonification. As a spin-off of this meeting, the book on Auditory Display [Kra94a] was published which is until now an important basic lecture.

Since the 90ies, auditory display research grew steadily. A small selection of the many publications are listed here grouped by research topic.

Sonification Toolkit Development Brewster investigated the usage of auditory icons and earcons for sonification. He analyzed the information requirements on behalf of user-interfaces. Brewster presented a sonically enhanced Interface toolkit [Bre96] and later studied how to sonically enhance drag and drop [Bre98]. Wilson and Lodha developed *Listen*, a sonification toolkit [WL96]. Lodha et al. presented *Muse*, a musical data sonification toolkit [LHJZU97].

Perceptual Studies Walker and Kramer conducted experiments on mapping variables to acoustic attributes [WK96]. Weinstein and Cook developed FAUST, a framework for sonifying algorithms [WC97]. Martins and Rangayyan tested the use of sonification methods for aural analysis of textured images [MR97]. Bonebright et al. discussed analysis techniques for evaluating the perceptual qualities of auditory stimuli [BMGC98] and applied Multidimensional Scaling to understand the perceptual structure of everyday sounds [Bon98]. Bussemakers investigated the cognitive processing of auditory and visual stimuli in a series of experiments using earcons and auditory icons [BdH00].

Sonification Techniques Axen and Choi presented sonifications to summarize the topological structure or connectivity of data [AC96]. Martins et al. developed a technique making it possible to sonify graphical textures in order to assist the examination of MRI images [MRP$^+$96]. The technique of *direct sonification* was introduced by Fernström [FM98], allowing to browse a database of musical patterns. For this he introduced the *Aura* as the range of listening space. The mouse pointer is used to move the Aura in the display and the Aura size can also be adjusted by the user. The sounds for all data within the Aura are played with a level that is reciprocal to the distance of the Aura center. The soundscape is rendered in real-time and thus reacts on moving the Aura in the display.

Barrass wrote his dissertation on Auditory Information design. Here he proposes a systematic approach to design sounds by analysis of the task (TA) and the data (DA) domains - the TADA approach. He created a database of stories about sounds and their function for information processing which helped him to select suited sonification mappings. Kramer invented auditory beacons, a technique to store, retrieve and compare sonifications for different data records or subsets [Kra92].

Applications Program Auralization was investigated by Vickers [AV97] in order to assist programmers in monitoring program flow and debugging. Saue and Fjeld conducted a pilot study investigating the use of sonification technique in seismic interpretation for oil exploration [SF97]. Krueger and Gilden presented an auditory display for blind people [KG97]. In their display, a touchable map, the feedback to touching is given acoustically, using speech and non-speech audio markers. Roth et al. developed techniques for active audio browsing of web pages for blind users [RPAP98]. Rubin designed an auditory display for the NY subway [Rub98]. Leplâtre and Brewster considered Earcons as a means to provide navigational cues in hierarchical menus, e.g. in cell phones [LB98, LB00]. Eckel [Eck98] used sonification within a CAVE to assist the analysis of vector field data from airflow simulations using wind-like sounds. The sound improved the detection of velocity changes. Ballora et al. applied sonification to the analysis of electrocardiogram data [BPG00]. Hansen and Rubin used sonification to keep track on discussions in chat rooms using both speech synthesis and entropy based piano accompaniment [HR01] Barra presented a musical sonification of web servers that allows to perceive abnormal web traffic from a change of style in the music [BCS$^+$01].

Summarizing, more and more applications of sonification and case studies are arising but the applied sonification techniques remain limited to some few classical types, mainly Parameter Mapping, Audification, Earcons, Auditory Icons and Auditory Stream Synthesis which will be introduced in Section 3.7.

3.4 Applications Fields of Sonification

This section will give a broad overview of the application fields of sonification. Although this thesis is mainly concerned with the application of sonification to exploratory data analysis, the wide range of other application should at least be mentioned within this overview.

3.4.1 Auditory Display for Visually Impaired People

Blind People seem to be the best candidates to test and use auditory display. Since they are incapable of using visualizations their listening skills are usually high-developed, so that auditory displays can be particularly useful for them. The first technical audio system for blind people was the Optophone, a reading machine with an audible output developed 1914 by Fornier and D'Albe. It produced a six-tone code for letters in scanned documents. As speech synthesis was impossible at this time, sonification was used as a replacement. As speech systems are widely available today sonification finds another application with blind people. Recently, auditory display and sonification have begun to play a new role as a means for inspecting visual scenery [Mei01] or to convey information about the structure or layout of an document (e.g. while browsing websites on the Internet [RPAP98]). Here, non-verbal auditory streams can be a valuable help for the blind.

3.4.2 Alarm Systems

Alarms are a subset of Auditory Display. Sound is specifically suited to be used in alarm systems, as we have no ear-lids and thus cannot ignore them. We even wake up by auditory alarms (e.g. the alarm clock). One may suspect that alarming was the main purpose for evolution to develop our auditory sense. Alarms are associated with an urgent situation and alarm sounds are designed to stand out in the prevailing acoustic ecology. They usually have a strong effect of drawing the listener's attention towards them. In technical systems, auditory alarms can be found in telephones, doorbells, car horns, alarm clocks, smoke detectors, lift doors and so on. In most alarm systems no further information is given acoustically than to signal the occurrence of an alarming situation. Naturally, the association from an alarming sound to its cause is something that has to be learned. However, with increasing sound technology more elaborate alarm systems are developed which provide details about the alarming situation by the acoustic properties of the alarm sound. As data is involved into sound generation the alarm then is a sonification. A prominent application of alarms is in airplane cockpits. Here about a dozen of different alarms may occur. Patterson [Pat82] did a study about auditory warning systems for civil aircrafts, in which he addressed intensity, spectral and temporal characteristics as well as the ergonomics of auditory warnings. His guidelines stress the necessity to take human factors into account.

3.4.3 Enhancement of Graphical User Interfaces

Most users are familiar with Graphical User Interfaces on computers and their control with a mouse pointer. Integration of sonic clues and acoustic feedback aims at improving the users' performance and at reducing the error rate. One of the first people to use auditory cues as system feedback was Gaver [Gav89] who programmed the SonicFinder for the Macintosh in 1989. Within this auditory display, auditory icons were used to represent data, which are discussed in Section 3.7.3. The association from the sounds to their meaning is given by a metaphor (e.g. screen - desktop, files - paper, delete - trash can) and has to be learned. Intuitive associations help keeping the learning effort low.

Brewster [Bre98] extended Gavers GUI sonification approach and conducted a systematical analysis of the human-computer interaction, e.g. performance, time and error rates. In current operating systems (e.g. MS Windows, Linux with KDE) audio themes exist which bind a set of sounds to various system events. However, these themes are mainly not designed to deliver any useful information but rather as an entertainment and thus they are very often switched off. This stresses the need for a proper design of auditory display. One might call an auditory display properly designed if it is not annoying but helpful to perform a task.

3.4.4 Process Monitoring

Auditory Displays are particularly useful for monitoring processes because of two reasons: firstly auditory monitoring is eyes-free. The user may therefore perform other tasks at the same time. The second reason is the "backgrounding capacity" of our auditory system: certain sounds are given a low attentional priority while the auditory system maintains sufficient awareness of these sounds so that any significant changes will draw the listener's attention.

We are very well suited to live in environments where quasistationary soundscapes are part of the acoustic background. In most offices the sound of the fan in the computer can be heard. If this sound is quiet enough it is not annoying and so the ear habituates to the sound and is able to ignore it. However, if something is wrong with the fan and its sound pattern changes even in a subtle way, this is easily noticed and draws the attention towards it immediately. Other everyday examples of process monitoring are car driving (any abnormal sound of the engine is rapidly detected) or using an electric kettle (the boiling sound indicates that it's time to make the tea). The advantages of auditory process monitoring have triggered the development of many auditory displays for that task. Fitch [FK94] developed an auditory display to monitor the patients condition during medical surgery which included up to 8 continuous observables. In experiments where the task was to keep a computer-simulated "digital patient" alive, he found that subjects performed faster and more accurately when using the auditory display than when using the visual display. Gaver [GSO91] applied auditory process monitoring to the observation of a simulated Coca Cola factory. In this ARKola experiment human subjects were asked to control the process and react to any changes in the whole process, which was presented either by visual or auditory means. He found that auditory displays reduced the error rate and also accelerated reaction time.

Thomas [Tho01] investigated in his Master's thesis the application of auditory display for stock market data. He presented a Parameter Mapping Sonification for real-time process monitoring of option market trading. His tool allows to keep track on many products simultaneously.

3.4.5 Exploratory Data Analysis

In everyday life we frequently use our ears to explore our environment. We explore the consistency of the floor from the sound of our footsteps, we explore the mood of our dialog partners from the sound of their voice and by shaking a present we try to conclude on what it is. Most of these exploration processes take place subconsciously and only in very few situations we become aware of what the auditory system really does. Usually we undervalue the enormous cognitive performance of our auditory brain because we are so much used to it. For instance when we understand speech in noisy environments, the auditory system does an enormous job to separate the background sound from the speech signal to associate the correct meaning to the sound. This outperforms current speech recognition systems by far. In the same time we classify a sound (e.g. as a car engine sound) and we are able to interprete variations within the sound categories (e.g. pitch, level, roughness of the sound) and to infer on the physical implications (e.g. the speed of the car). Most important is that we are able to build up new categories and learn to interprete previously unknown sounds. This makes listening an interesting channel for data exploration.

Whereas in some domains listening is already used as an exploratory tool (e.g. the stethoscope) its application on the exploration of data is so far underdeveloped. There have been some publications about the application of sonification for the exploratory analysis of specific types of data, e.g. for the analysis of certain chaotic systems [MKBC94], the analysis of topological properties of a graph in high-dimensions [AC96] or for fluid dynamics, network traffic or seismology.

3.4.6 Miscellaneous Application Fields

Besides the main application fields of sonification listed above, there are a number of minor fields where sonification can contribute to a better performance:

Sonification in Virtual Environments: Virtual Reality systems aim at immerging the user into a simulated reality. This also requires to create an appropriately simulated sound space. If sound properties here are determined by available data of the objects in interaction, it is a non-verbal presentation that may be called sonification.

Augmented Displays: In contrast to virtual reality, the goal here is to supplement our reality with additional information. Semitransparent head-mounted displays are used for this in the visual domain. Sonification here can contribute by highlighting sonic objects or adding other informative sound clues with acoustically transparent headphones.

3.5 Functions of Sound

This section focuses on the relevance sound has for our performance in the real world. The section begins with a closer look at the importance of sound in everyday life. The function of sound is the answer to the question: what is the purpose of sound. Similar functions of sound may be exploited in sonifications.

We perceive environmental sounds at almost any time. These sounds may include singing birds, passing traffic, clapping doors, somebody knocking at the door, the sound a pencil makes while writing on paper. Every keystroke on my keyboard produces a sound. Whenever we manipulate objects it is accompanied by sound. Sounds can be grouped into two distinct groups: *passive sounds* like the sound of rain and wind and *active sounds* like the the sounds produced by human interaction with the environment

Passive sounds have two functions: they can (a) give us a summary about the state of our environment and they can (b) *alert* us and draw our attention to events that are potentially dangerous for us (like an attacking animal or an approaching car). Our auditory senses are permanently active for our sake. We have no ’ear-lids’ which we can close and which could endanger us. We even pay attention to our acoustic environment while we are sleeping.

It is the function of active sounds to give a *feedback* to our actions and their failure or success. They provide us with information about the material we are manipulating (e.g. its stiffness or surface roughness), about the progress of an action (e.g. snap-in sounds while interconnecting objects), about objects which are out of sight (e.g. shaking an opaque box). In other words, we can use our actions to *explore* the world by causing sounds.

Sound may further be used to assist *communication*: Playing a musical instrument is a means to express our emotions, and rhythmical gestures (e.g. knocking on a table) may be used to emphasize certain points while speaking.

In many situations we are not even aware of how much we use the acoustic information from our environment as these processes run subconsciously. Some examples are: most people automatically turn their heads towards the source of a sudden sound (e.g. a barking dog). The sound is *guiding the eyes*. While filling a bottle the rise in pitch is used to infer to the degree of filling. The sound allows us to *monitor the process*. While using an electric razor we know from the sound where the shave is completed.

These few examples show how auditory information is routinely used without particular or conscious intention. An important function of sound is communication between humans. Different layers of information are available in speech sounds: the meaning of the words, the identity of the speaker, gender, the listener even gets information about the physical or emotional state of the speaker. Table 3.1 summarizes the identified functions of sound.

3.6 Auditory Display: Benefits and Difficulties

In the previous discussion of the functions of sound in everyday usage we already discovered some benefits of using sound in auditory displays. These shall be summarized in this section.

Function	Example
communication	speech, language
feedback on success of actions	contact sounds
guiding the eyes	localization of interesting sound sources
alerting / alarming	getting aware of approaching dangers
enjoyment / relaxation	music
recognition/identification	animals/persons, e.g. in the dark
categorization	engine sounds: vehicle type, voice: gender
process monitoring	e.g. status of coffee machine
orientation	orienting oneself from known reference sounds
exploration	learning material properties from interaction sounds
coordination	cooperative activities

Table 3.1: List of the functions of sounds within human activities.

After this, an overview of typical problems in auditory displays is given.

- Auditory displays can be used **eyes-free**. They can therefore be used when the eyes are already occupied with other tasks or when it is impossible to use visual displays as for visually impaired people.

- Our auditory system is able to **draw attention** to acoustic signals. As we are further capable to conclude on the localization of the sound source, sound can be used to draw attention towards other interfaces which might be out of sight.

- Due to the **high temporal resolution** and **short detection times** for sound signals, auditory display can be used for alarms or alerting purposes.

- Our ability to **listen parallel** to several audio streams makes auditory display both suited for monitoring purposes and presentation of multidimensional data.

- We are able to **habituate** to certain auditory streams and assign to them a lower attentional priority. At the same time we remain sufficiently aware of the sound to be alarmed by any changes of the sound. This property is also called **backgrounding** [Kra94b].

- Our auditory system has the ability of **auditory gestalt formation**, which means to perceive complex sound patterns as a whole without the need to direct attention to its components. Our brain immediately associates a meaning with the gestalt.

- Our auditory system performs a **frequency decomposition** of the input sound signal. We have a high resolution in the frequency variable a well as the time variable. Within the time variable we are very **sensitive to rhythm** and its changes.

- **Large temporal acuity**: the temporal resolution ranges from some milliseconds (perceived as pitch) to some seconds (perceived as rhythm). Throughout the whole range we have a high temporal resolution.

- **Auditory memory** allows us to retrieve many familiar sounds for comparison instantly. Examples for this are the speaker identification or the memory of melodies.

In combination with visualization or other modalities further benefits can be identified:

- Auditory Display does not interfere with visualizations, so that it may be used to supplement or augment visual displays. Sonification can e.g. be used to increase the dimensionality of the displays by distributing some data dimensions to acoustic attributes.
- Engagement: Sound in user interfaces can help to reduce fatigue, decrease learning times and increase the performance and enthusiasm.
- Enhanced Realism: interaction with graphical objects can become more realistic when properly synchronized audio is added.

These benefits strongly indicate that auditory display outperforms other senses in specific points. However, auditory displays do not only have advantages. Problems with Auditory Display arise either from the characteristics of the media or from the context in which Auditory Displays are used. The most important difficulties are:

- **Low resolution** in some auditory variables: precise representation of quantitative data by sound is difficult. For attributes like brightness or attack time, only a limited number of values can be discerned. Furthermore, many acoustic attributes are perceived as distinct acoustic features in different value ranges. In amplitude modulation, the listener either perceives beats, roughness or two detuned tones, depending on the modulation frequency.
- **Limited spatial precision**: The ability to localize sound is much less accurate than the angular precision of our visual senses. This must be taken into account when representing spatially indexed data. Also, the correct interpretation of spatial clues is reduced when the number of sounds within the display increases.
- **Lack of absolute value**: human listeners fail to conclude from a certain auditory perception to the value (e.g. from tonality to frequency). There is no intrinsic reference system like the coordinate system in plots which allows to infer the data value from its acoustic attributes.
- **Lack of perceptual orthogonality**: for most acoustic attributes there are strong couplings which influence the mutual perception. To give an example, independent upon amplitude, the perceived loudness decreases by increasing the frequency towards 10 kHz. This dependency between pitch and loudness and other perceptual couplings have to be taken into account in Auditory Display. Unfortunately, auditory perception is so little understood that it is not possible to specify many orthogonal attributes. It is possible to reduce the problem by limiting the attribute ranges. However, this will reduce also the resolution.
- **Annoyance**: sound may be disturbing if it does not provide any useful information for the task at hand. Apart from that if loudspeakers are used a sound is not limited to a single working place so that it may disturb other people. Auditory Display design has to put a special focus on this issue.
- **Interference with speech communication** can create resistance to auditory displays.
- **Absence of Persistence**: Time is an essential element in sonifications and data is usually serialized for auditory presentation. The data presented at the end of the sonification can not directly be compared with data played at the beginning. For such comparisons interactive auditory displays are necessary.

- **No printout**: It is impossible to put sonifications into printed papers without additional technical playback devices[3]. As scientific publications appear mostly in printed journals, results can not be presented using the same media. This, however, is getting less problematic with the increasing availability of the Internet.
- **User limitation**: The ability to understand and interprete sound depends on the user. While the understanding of visualizations can be easily taught, auditory perception is much more subjective.
- **No pointing in acoustic presentation**: Communication about sound is difficult for two reasons: we cannot "point into sound" as we can point on elements in a visualization, and we are not used to communicate about sonifications. Multimedia technology may help around such obstacles.
- **Open workplaces**: As we have no ear-lids we can not totally ignore sound coming from other workplaces. Sound systems that use loudspeakers may impair the users privacy.
- **Learnability**: is both a benefit and a difficulty. On the one side, humans capabilities to learn to interprete sound is addressed with sonification, on the other hand it is necessary to invest some effort until one has learned to interprete an auditory display correctly. Tutorials and reference sonifications for typical data may help new users to shorten the learning phase.
- **Cultural Bias**: Harmonic or rhythmic elements in sonification may vary among cultures. While European listeners are used to the diatonic scale, in India a quarter tone scale is used. This could influence the perception of consonance or dissonance.

3.7 Sonification Techniques

In this section, different techniques for the rendering of auditory data representations will be presented. The first technique presented is *Audification*, which simply takes the data values as a time series of sound pressure values. *Earcons* and *Auditory Icons* are discussed then as a technique to present categorical data. *Parameter Mapping* is the most common technique to render sonifications. Many researchers even use sonification synonymic to Parameter Mapping since this technique is so dominant in literature. Finally, the technique of *Model-Based Sonification* is presented which is developed in this thesis.

3.7.1 Audification

Audification is the most simple and direct auditory display technique for translating data into sound. Given a dataset of records $\mathbf{x}_i, i = 1, \ldots, N$, an audio signal is assembled by simply taking the series of data values of one variable x_{ij} as samples $s[i]$. Let us assume there are variations in the data series x_{ij}. This will translate to variations in s which are audible if (i) the variations lie in the audible frequency range from 50 Hz to about 20000 Hz and (ii) the amplitude at those frequencies exceeds the frequency dependent listening threshold (which can of course be reached by amplification). The samples are converted into analog air pressure values at a sampling rate ν_{SR}. Typical sampling rates are about 40 kHz. 40000 data values are then needed for a single

[3] DataSound (`http://www.datasound.de`) developed a technique to print sound on paper as a point pattern.

second of audification. For the perception of pitched sound with about 400 Hz, an almost periodic variation in the data must be evident with a periodicity of $\frac{\nu_{SR}}{400\text{ Hz}}$ samples.

Obviously, we need a lot of data values even for a short audification and in addition audification is limited to datasets which can be ordered in an reasonable way, like time series data. In some applications, however, data of exactly this specification is available, like in seismic measurements [Hay94] or with the analysis of dynamic systems [MR94b]. Often, the data can be brought into the audible range by resampling. In order to bring very low-frequency components to an audible frequency an up-sampling is required which shortens the duration of the audification, eventually making it too short to perceive any patterns at all. To avoid such temporal compression, time-stretching and pitch scaling techniques [Moo90] can be applied prior to playing the audification. Further operations onto the signal like shifting, scaling, normalization, dynamic range compression are possible candidates for a postprocessing of the audification in order to improve the perception of acoustic features. The given transformations can be performed by assigning a suited transfer function on the audification by $s'[n] = g(s[n])$.

3.7.2 Earcons

Earcons are non-verbal audio messages consisting of motives, which are short rhythmic sequences of pitched tones with variable timbre, pitch and amplitude [BWE94]. Their main use is to communicate symbolic messages within the computer/user interface[4]. To give an example, there might be an earcon to represent “File” and another earcon to represent “open”. These earcons could be played in a sequence in order to inform the user that a file was opened. Besides juxtaposition, other methods like inheritance and transposition exist to combine earcons.

Concerning their semantics and syntactical organization, similarities can be drawn to linguistics. In a way similar to learning words and then combining them into messages earcons can be applied [MMBG94].

MIDI sound synthesizers offer a rich repertoire of musical and percussive sounds which were often used for the creation of earcons. In auditory data display, a limited set of values of a categorical variable can be assigned to different earcons. In an auditory city map, for instance a different earcon may be used for each building type.

Earcons have been applied to enhance graphical user interfaces and to help orientation in tree-like hierarchies [BRK96].

In the context of data sonification earcons can be used to annotate certain events of interest, similar to reference lines in a graph. Thomas used earcons in combination with a parameter mapping sonification of stock market data to signal “buy” or “sell” indications [Tho01].

3.7.3 Auditory Icons

Auditory Icons are everyday sounds that convey information about events in the computer or in remote environments by analogy with everyday sound-producing events [Gav94]. Auditory Icons represent discrete items or events. A caricature of the thing or event being represented is symbolically and metaphorically presented in sound. The first application of Auditory Icons was to produce sound effects for interaction with icons on the desktop, e.g. selecting a file produces the sound of tapping an object. Similar to visual icons, auditory icons rely on the analogy between the everyday world and the model world. The more direct this analogy is, the easier auditory

[4] The continuum of analogic-symbolic is presented in Section 3.8

icons are understood. Thus properly designed auditory displays allow people to make use of their existing everyday listening skills.

The SonicFinder [Gav89] was one of the first applications to also use auditory icons to present additive information to the listener, e.g. by using the file size, file type or dragging location to determine acoustic properties of the auditory icon. Enhancing auditory icons in this way leads to Parameterized Auditory Icons [Gav94].

The SoundShark [GS90] was an application in which auditory icons were used to provide a feedback on interactions with objects. The scenario is a virtual physics laboratory for distance education. It provides for its users a model world which was much larger than the screen. Gaver applied auditory icons to assist navigation: auditory landmark objects simply kept playing a repetitive sound to assist orientation.

In the ARKola experiment the performance of subjects on controlling a complex system (in this case a simulated soft-drink plant) was studied. The production consisted of an assembly line of nine machines. Production was in one case monitored visually, in the other case by the aid of an auditory display consisting of up to 12 sounds playing simultaneously. Six pairs of participants were asked to run the plant with the aim of making as much “money” as they could during an 1 hour session. They came to the result that sound was effective in two areas:

- sound helped to keep track of the many ongoing processes. Without sound people often overlooked broken machines. Furthermore people felt that the soundscape merged into an auditory texture which they recognized as typical for normal operation.
- Usually the two participants shared responsibility for a part of the machines. With sound, however, each participant was able to hear directly the status of the remote part and that seems to enhance collaboration between the partners.

The main difference between auditory icons and earcons lies in the way how listeners can associate a meaning to the sound. While auditory icons are easily understood by making a metaphorical association to the interacting object, the meaning of earcons has to be learned. Auditory icons build on our everyday listening skills, earcons make use of musical or abstract listening skills.

Auditory icons are seldom used to present arbitrary data. They are in most cases applied to signal an event. However, in combination with other display techniques like Parameter Mapping Sonifications, auditory icons can be used to assist orientation in the display. For example auditory tick marks can supply the analogue to axis tick marks in a plot and such markers can be provided by auditory icons like a metronome click sound (see p. 139).

3.7.4 Parameter Mapping Sonification

While the previous sonification methods are either limited to large amounts of data (audification) or discrete signals (earcons and auditory icons), the parameter mapping technique is a versatile display. The concept of mapping is inspired by the technique of data plotting: in scatter plots, graphical elements (symbols) are added to the display area and the symbol attributes (x, y-coordinate, color, symbol type or size) may be driven by variable values of the represented record. Analogous, the auditory display area is given by the duration of the sonification and auditory elements (events) are superimposed. Whereas in visualizations only a few attributes are used to determine a symbol, sonic events can have many different acoustic attributes like onset, duration, envelope, volume, pitch, pitch variation, timbre, amplitude modulations, roughness. More complex events may even use rhythmical and harmonic features as attributes. To formalize the Parameter Mapping let us assume a given d-dimensional dataset $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$. An acoustic event in the sonification is described by a signal generation function $\mathbf{f} : \mathbb{R}^{m+1} \rightarrow \mathbb{R}^q$ which computes a q-channel sound signal $\mathbf{s}(t) = \mathbf{f}(\mathbf{p}, t)$ which is a function of time. $\mathbf{p}$ is an m-dimensional vector of acoustic attributes which are parameters of the signal generator. For stereo sonifications, q takes the value 2. A Parameter Mapping Sonification is then computed by

$$\mathbf{s}(t) = \sum_{i=1}^{N} \mathbf{f}(\mathbf{g}(\mathbf{x}_i), t) \ , \tag{3.1}$$

where $\mathbf{g} : \mathbb{R}^d \rightarrow \mathbb{R}^m$ is the parameter mapping function. Let $\mathbf{x} = (x_1, \dots, x_d)^{\mathrm{T}}$ denote a data point. Simple parameter mappings do not mix between dimensions but map values of an acoustic attribute p_i from a single data variable x_j. As in scatter plots where all data points of a graph may have the same color, acoustic attributes may be assigned a constant value. Such a mapping can be written as

$$\begin{aligned} p_1 &= h_1(x_{k_1}) \\ p_2 &= h_2(x_{k_2}) \\ &\dots \\ p_m &= h_m(x_{k_m}) \end{aligned}$$

The functions $h_j(\cdot)$ provide a mapping of data values to attribute values. Usually monotonous functions or constant values are used. Figure 3.2 shows some frequently applied mapping functions. The linear mapping with a clipping to min/max values in the attribute domain is used so frequently that a new notation shall be introduced for it:

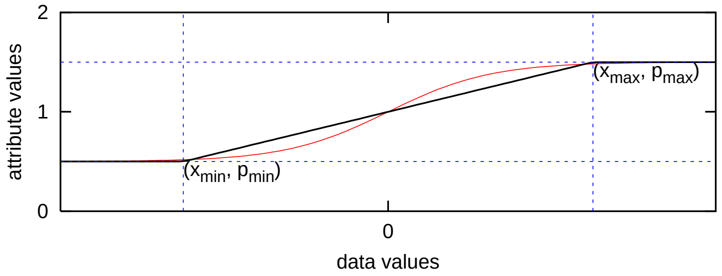


Figure 3.2: Typical transfer functions used for Parameter Mapping. The piecewise linear transfer function (black line) is described by the map() function in equation 3.3.

$$p(x) = \mathrm{map}(x, [x_{min}, x_{max}], [p_{min}, p_{max}]) \tag{3.2}$$

$$= \begin{cases} p_{min} & : \quad x \leq x_{min} \\ p_{min} + \frac{p_{max} - p_{min}}{x_{max} - x_{min}}(x - x_{min}) & : \quad x_{min} < x < x_{max} \\ p_{max} & : \quad x \geq x_{max} \end{cases} \tag{3.3}$$

provides the mapping shown in Figure 3.2.

The Parameter Mapping Sonification technique as presented here is also sometimes referred to as *sonic scatter plots* [MR94a] or nth order parameter mapping [Sca94].

While this functional description is suited for implementing Parameter Mapping Sonifications, it is recommendable to use a more readable textual representation for interpretation of the sonification. Each component of the attribute vector is given a meaningful name (i.e. pitch, onset, amplitude) and the variable names of the dataset are used and thus a mapping is written for example as:

$$p_0(x) = \text{map}(x_3, [a,b], [d,e]) \quad \Leftrightarrow \quad \text{onset}[d,e] \leftarrow \text{attribute}[a,b]$$

A complete parameter mapping can thus be represented by an assignment table, here shown in an example for the 4-dimensional Iris dataset and a sound generator with 4 attributes:

$$\begin{aligned} \text{onset}[0,2] &\leftarrow \text{petal\_length}[-,-] \\ \text{pitch}[7,10] &\leftarrow \text{petal\_width}[3,5] \\ \text{amplitude}[60,90] &\leftarrow \text{sepal\_length}[-,-] \\ \text{duration} &\leftarrow 0.5 \end{aligned}$$

The ‘–’ for the limits indicates that the minimal, resp. maximal data values are computed from the available dataset.

Parameter Mapping can be extended in several ways. One useful extension consists of *auditory beacons* developed by Kramer [Kra92]. Auditory Beacons provide references for fast comparison of data vectors. Static Beacons represent single data vectors in form of a short sound phrase rendering for this data point. They are especially suited for comparing data in a monitoring situation with some reference situations, like (Beacon 1:) the system works normally or (Beacon 2:) a certain mistake is evident. Dynamic beacons are Parameter Mapping Sonifications for a limited subset of the data set at hand. They may be used to compare different evolutions of systems in time, e.g. the cycle of different stock market crashes. Auditory Beacons consist of a data part and a mapping part. The mapping part specifies a reference mapping which is used for the sonification of the data specified in the data part. The combination of mapping part and data part is required to compute a reference sonification for comparisons.

The main advantage of Parameter Mapping for data presentation lies in its limited computational complexity. Sonifications are implemented straight forward by using simple algorithms. Since many instrument sounds can be synthesized with efficient algorithms, sonifications can be rendered almost in real-time. The time complexity is linear with the number of data points.

Parameter Mapping Sonification faces the following difficulties:

- **Perceptual Interactions**: the perceptual quality of some acoustic attribute depends on other attributes. For example while decreasing the attribute duration the frequency is more and more undefined. Increasing the amplitude while keeping the frequency constant results in an increase of pitch, so that loudness and pitch interact. The same problem is sometimes also referred to as **Lack of Orthogonality**.

- **Nonlinearities**: Variations of the attribute values lead to variations of perceptual qualities. But the dependency is often nonlinear and in most cases unknown. E.g. the perceived pitch of a tone depends on its main frequency. But doubling the frequency does not result in a doubled perception of pitch. Here, the mel scale provides the dependencies [ZF99]. For most attributes the nonlinear dependency is unknown. Keeping value ranges small reduces the problem but also the perceptual resolution.

- **Unbalanced Attributes**: Some acoustic attributes determine the perception stronger than others. The perception of pitch dominates over the perception of brightness or attack time.

It is unclear how to select attribute ranges to get a better balanced display. It remains the need for a subjective tuning of the ranges.

- **Different Perceptual Resolution**: In different perceptual qualities human listeners show different resolutions. This implies that a different number of perceptual values can be discerned.

- **No Unique Mapping**: There is no unique mapping from data to acoustic attributes. Thus there is no canonical way to determine a mapping. Therefore a manually performed assignment is necessary and the mapping is required in order to infer from perceived qualities to data relations.

- **No unique polarity** It is unclear which polarity associates data variations to perceptual variations. Rising temperature could be both represented by rising or falling pitch.

- **Learning / Interpretation**: Sonification depends on the chosen mapping. Each mapping sounds different. This makes learning and adapting to these sonifications difficult. Unless the mapping is known by heart, it is necessary to have a mapping table help to interprete the sounds meaning.

- **Limited Dimensionality**: The display dimensionality is limited to the number of acoustic attributes. The technique is thus not able to represent arbitrary-dimensional data.

- **Limited Interaction**: Parameter Mapping sonifications are rendered and played in separated steps. Thus these sonifications intrinsically do not offer intuitive interactions except maybe start, stop or pausing the playback.

Understanding the interaction of perceptual dimensions and developing strategies to overcome the problems is pursued in current Auditory Display research [ICA]. Psychophysical experiments are conducted that provide some guidelines [WK96].

3.7.5 Model-Based Sonification

Model-Based Sonification is a technique for the rendering of sonifications. The framework is developed within this thesis and presented in detail in Chapter 7.

To give a brief summary, sonification models aim at providing a setup of a dynamical system which is parameterized from the dataset. The model further provides a dynamics that determines the elements' behavior in time. Furthermore, some interaction modes are specified so that the user of a sonification model is able to interact with the model. The sonification is the reaction of the data-driven model to the actions of the user.

The main advantage sonification models have, is that they can be designed in such a way that they work on data of arbitrary dimensionality. Knowledge of the model further provides the key to an interpretation of the sound with respect to the data. In contrast to Parameter Mapping, here no mapping has to be specified. Sonification models only have few parameters and these parameters are related to physical properties. Their function for the model is therefore intuitively grasped. Sonification Models can be designed to fit a task at hand and offer intuitive ways of interacting with the auditory display.

The main disadvantage is that these models may be computationally much more expensive than Parameter Mappings and that a model design is required. Hopefully after some evolution, appropriate models for many analysis tasks will exist.

3.8 Categorizations of Auditory Display

In this section,two approaches to categorize sonifications are presented. The first approach bases on a classification of the directness of the association from data to sound and orders sonification techniques along a continuum between analogic and symbolic. The second approach uses a taxonomy from linguistics, more precisely from semiotics to explain and group sonification techniques.

3.8.1 The Analogic-Symbolic continuum

This section reports a discussion from Kramer [Kra94b] who applies ideas of Sloman [Slo85] about analogical representations to Auditory Display.

A **symbolic** representation associates the thing being represented categorically. The information being represented is clustered in categories and the relationship between the representation does not reflect intrinsic relationships between the elements being represented. Words are typical examples for symbolic representation. To give an example, relations between three perceptions of temperature (e.g. cold - warm - hot) are not reflected by relationships in the used words: a different assignment would work just as well if we agreed upon it.

In an **analogic** representation an immediate and intrinsic correspondence between the represented thing and the representation is given. Relations within the represented thing are mapped to relations in the representation which show the same qualitative structure, i.e. small changes of the represented thing map to small changes of the representation. However, the representation may be a simplification of the thing represented. A typical example of an analogic representation is a thermometer. The height of the thermometer column analogically represents the temperature.

Analogic and symbolic are isolated points which span a continuum of representation techniques between them. Along this scale (see Figure 3.3), the sonification techniques presented in Section 3.7 are now ordered into:

- Language: is found on the symbolic end of the scale. This however only regards the word meaning and not any meta-information given by prosody.

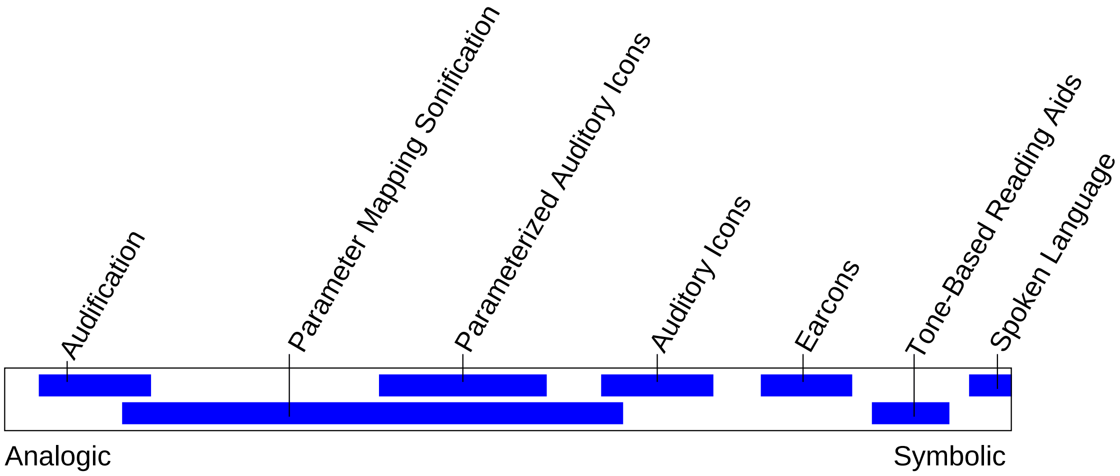


Figure 3.3: The Analogic/Symbolic Continuum

- Earcons: distinct earcons are assigned to distinct entities or messages. They are similar to language in character. However, such earcons may also have an analogic aspect (e.g. denoting a buffer overrun by an uprising chain of tones and a buffer underrun by a falling chain of tones).

- Auditory Icons: They are an acoustic caricature of the thing or event represented. The presentation is symbolic as with earcons, but here the metaphorical association helps to interprete the sound. Gavers Parameterized Auditory Icons [Gav94] show that they may also have a strong analogical component. Determining acoustical properties of the sound by available data (e.g. the loudness of a tap sound to the size of an opened file) analogically represents data at the same time. So Auditory Icons are found in the middle between symbolic and analogic.

- Parameter Mapping: The mapping from data values to acoustic attributes is analogic if small variations in the attributes result in small changes of the sound. This is the case for most attributes like frequency, amplitude, attack time, duration, brightness and so on. Parameter Mapping Sonifications may also have a symbolic component, e.g. if the value of a nominal variable is mapped to the instrument class. Such categorical association from sound to data is symbolic. To sum it up, Parameter Mappings are closer to the analogic end of the scale.

- Audification: This is without doubt an analogic data representation.

- Model-Based Sonification: The model-dynamics associates interaction sounds to a complete dataset. Similar data distributions will in most cases yield to similar sounds. However, the model may contain critical borders which yield to qualitative changes of the sound even on smooth changes in the dataset. Thus Model-Based Sonification is difficult to locate on this scale but may be found closer to the analogic side.

The human's capabilities to detect patterns in sonifications or other sounds have an important impact on the analogic/symbolic nature of the display. By detecting an acoustic pattern and associating it with some structure in the underlying data, our brains create a new category which symbolically associates the detected structure with the acoustic signature. The auditory representation becomes an auditory gestalt which is immediately recognized and associated with a newly created category. In other words, the listener learns “the language” of the display. This learning process is the actual aim of sonification for Exploratory Data Analysis. It goes along with a shift from an analogic to a symbolic use of the display.

3.8.2 Semiotic Categorizations of Auditory Display

The categorization of auditory display according to semiotics bases on work of Blattner, Papp and Glinert [MMBG94] and the dissertation of Barrass [Bar97]. Semiotics is the theory of signs and their meaning and it can be used to analyze communication media. A sign is anything from which meaning may be generated. It has two parts: the signifier and the signified. Syntax is the formal composition of elements in a language. Pragmatics studies how the signifiers are formed, e.g. if there is a limited lexicon of signs and how signs are distinguishable and memorizable. The lexical level of auditory messages is concerned with the acoustic attributes of sound. Finally Semantics is concerned with how meaning is associated with the signifier. The three most important association types are symbolic, indexical and iconic association. Symbols do not have to resemble the signified and the association must be learned. Indexical signs use a causal association (e.g. the barking sound of a dog indexically represents a dog). Finally, iconic representations resemble the signified[5].

Auditory display were classified along this taxonomy by Blattner et al [MMBG94]. Earcons focus on syntactic organization of acoustic material to communicate messages. The sounds are symbols for the signified. Auditory Icons are an example of the semantic approach: the meaning is associated to the sound by iconic association. Parameter Mapping Sonification is a lexical approach. The signs are created from the data. Barrass discusses further categorizations of Auditory Display, that emphasize certain issues like the perceptual approach, task-oriented approach connotation approach and device approach which provide further related perspectives on the classification or understanding of Auditory Display.

This short discussion shows that different ways exist how to classify auditory display techniques like by application, synthesis technique, semiotics or directness. The aim of this overview was to provide pointers to related work and to show different perspectives which might help to find inspirations for the development of sonifications, or maybe to understand sonifications better.

[5] For a more detailed discussion, see [Bar97].

Chapter 4

Auditory Perception

In this chapter different aspects of auditory perception are discussed. Analyzing what, how and why we hear provides limits for the design of auditory display and helps to create sonifications that are easy to understand. This can be achieved for instance by organizing the sound in a way similar to our environmental soundscape, so that the listener can profit from the listening skills he acquired by everyday listening experiences and from the evolutionary optimization of the auditory system. The discussion will start with presenting the human ear and how it works. This allows to understand the basic physical limitations of the human listening system. After this, the topic of psychoacoustics will be addressed. It contributes important knowledge about the functional mapping from stimuli to sensations.

Sound processing is done by the auditory brain. Here, the sound is segregated into different auditory streams and occluded or masked parts of the sound signal are reconstructed. Such phenomena are topics of Auditory Scene Analysis [Bre90]. Research in this field comes to the result that the auditory brain has found valuable solutions for stream segregation, and this justifies the usage of multi-stream auditory display, given that some basic rules are followed.

4.1 The Sense of Hearing

Sounds are vibrations of pressure in a medium, usually generated by a vibrating object. If the medium is in contact with the eardrums, the vibration can cause acoustic sensations. Normally, the medium is air. Vibrations of air pressure are caused by physical events like wind, a person speaking, musical instruments or any other contact with a vibrating object[1]. Depending on the structure and strength of the vibration we talk about noise, sound, tone, speech or music. The mechanisms of sound generation will be discussed in Chapter 6.

In the air, sound is transported by longitudinal waves which propagate with a velocity of about 340 m/s. Traveling sound waves are reflected and bundled by the outer ear (or pinna) through the outer ear canal to the eardrum shown in Figure 4.1. The outer ear and also the head and the shoulders have an important effect on the level of sound pressure in front of the ear drum. Such signal distortions are used by the auditory system to localize the sound source. The acoustic gain (or attenuation) of a sound through reflections depends both on the direction of the source and the sound frequency. The auditory system uses differences in timing, level and spectral profiles between sound signals arriving at the left and right ear to conclude from this information the

[1] Thermal motion of air molecules also causes stochastic air pressure variations but they do not exceed the perceptual threshold of our sensory system.

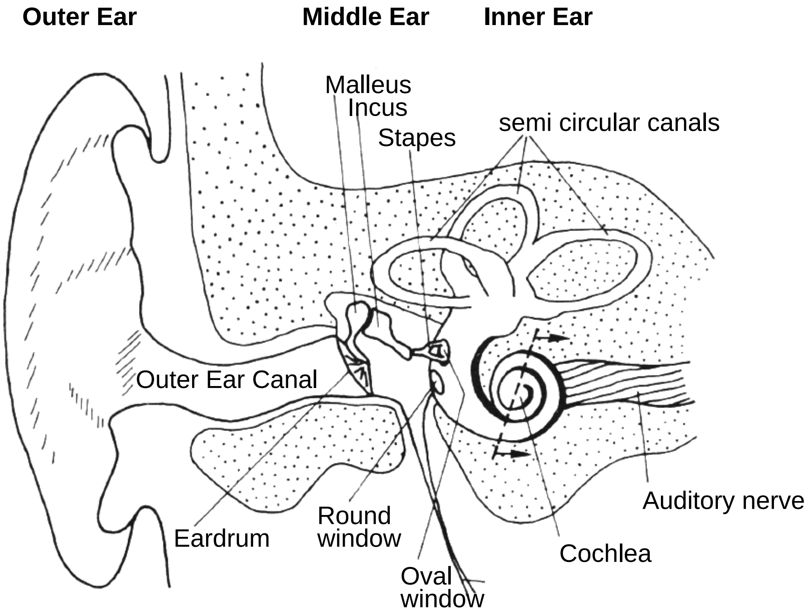


Figure 4.1: The outer, middle and inner ears. Modified from [ZF99].

localization of the sound source [Bla74].

The eardrum vibrates in response to sound. In order to cause the sensation of sound, these vibrations have to be passed on to the sensory cells whose nerve terminals are able to encode mechanical stimuli into electrical action potentials. This transformation from air pressure waves in the outer ear to fluid waves in the inner ear is done by the middle ear. As the impedance is much higher in fluid than in air, it is necessary to match the impedance, otherwise most of the wave energy could not pass the frontier. Impedance matching is achieved by two effects: firstly, the size of the eardrum is about 17 times that of the oval window, which is the bridge to the inner ear. By this, the pressure is amplified. Secondly, three little bones, the malleus, incus and stapes amplify the vibration by a leverage effect. Both factors amplify the vibrations by an factor of about 20.

The cochlea is a snail-shell-like structure. Enrolled it looks like a cylinder of 35 mm length and 2 mm diameter. The function of the cochlea is to convert the vibration of sound into nerve impulses in the auditory nerve. Sound vibrations arriving at the oval window lead to fluid waves that travel from the oval window to the apical end. These waves lead to vibrations of the Reissner membrane resulting in a relative motion of the organ of corti. Its most important constituents are the inner and outer haircells. These nerve fibers have stereocilies whose potential changes depending on their deviation from the equilibrium position. The potential changes are passed on to the nerve fibers and may at this point cause neuronal pulses. Along the basilar membrane there are about 3500 inner haircells and about 12000 outer haircells. Although there are much more outer than inner haircells, more than 90 % of the auditory nerves receive signals from the inner haircells, because their signals diverge into several nerve fibers whereas the signals of many outer haircells converge into single nerve fibers.

To understand hearing, it is important to understand the way the basilar membrane responds to any vibrations of the stapes. Let's assume that a sinusoid vibration of a given frequency arrives at the oval window. As a result a traveling wave propagates along the basilar membrane whose envelope shows a maximum at a frequency dependent position as shown in Figure 4.2. As a consequence, haircells along the basilar membrane respond selective to specific frequencies. From the shape of the envelope conclusions on masking effects can be made which shall not be discussed here [Coo99].

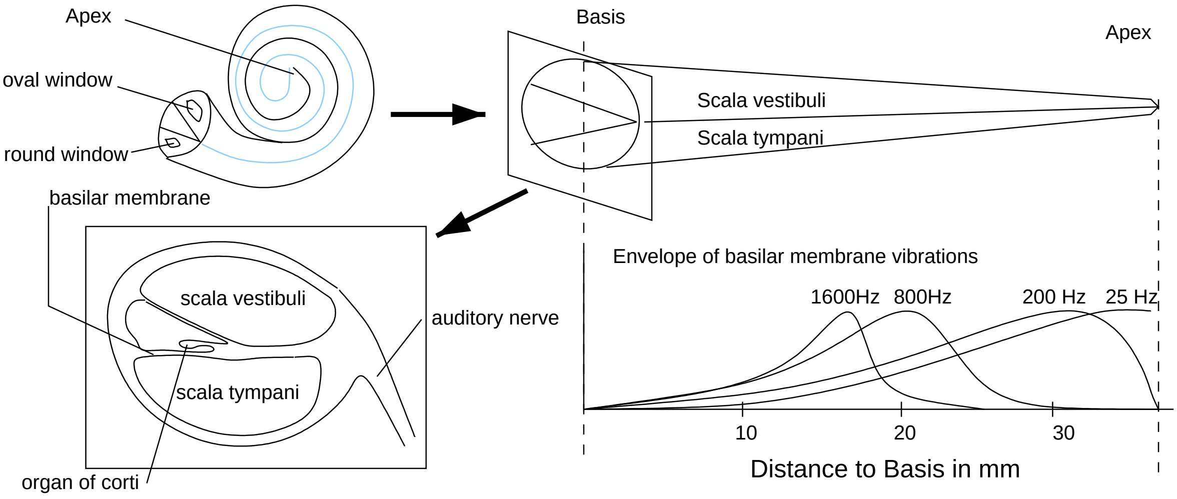


Figure 4.2: Illustration of the cochlea. Along the basilar membrane, sound waves force frequency dependent spatially varying envelope profiles as shown on the right for some frequencies as measured by Békésy.

For auditory display we can learn from the organization of the ear that there are certain physical limits for using sound waves to convey information:

- In the ear, sounds are decomposed into their frequency parts.
- The amplitudes have to be within certain ranges in order to be heard (so that they cause nerve pulse) and not to damage the ear (physical limits).
- The outer ear is of crucial importance for sound source localization. In order to use localization clues within auditory display using earphones (headphones which are inserted into the auditory canal) the reflections at the outer ear and proper signal timing need to be modeled.
- The frequency range is limited to about 50 Hz – 20000 Hz.
- Assuming there are 4000 haircells evenly distributed along the basilar membrane, and assuming a frequency range of 20000 Hz, the frequency resolution cannot be better than about 5 Hz. As the maxima of the envelopes of the basilar membrane are closer together at high frequencies, frequency resolution is expected to decrease with increasing frequency.

4.2 Psychoacoustics

Psychoacoustics assesses the relation between stimuli and the hearing sensations they cause. The stimuli are described by physical properties of the sound signal and can be measured and controlled exactly. Chapter 6 focuses on the description and generation of sounds or stimuli with specific properties. Examples for stimuli are sound pressure level, frequency and duration. More complex sounds have to be described further by their temporal and spectral evolution. Hearing sensations are caused by a stimulus if it exceeds a perceptual threshold. Typical hearing sensations are pitch, loudness, subjective duration. These sensations cannot be measured as easily as the stimuli because they depend on the listener. Psychophysical experiments are required to determine how stimuli and hearing sensations are related.

Human listeners are able to pay attention separately to different hearing sensations. They can for instance compare the loudness of two tones whose frequency differs. This is an important prerequisite for psychophysical experiments [ZF99]. One goal of psychoacoustics is to create sensation magnitudes and to determine their functional dependencies on the stimuli. This is made difficult by the fact that several stimuli may influence a single hearing sensation. For instance, although the perceived pitch depends mainly on the frequency, pressure level also has a small effect on pitch perception.

Interesting observables are the **absolute threshold**, which is the stimulus magnitude for which the corresponding sensation is audible for 50 % of the listeners and the **difference threshold**, which is the stimulus increment by which 50 % of the listeners have a difference in sensation.

This section will present some psychoacoustic findings for the most important hearing sensations: loudness, pitch and timbre. The associated magnitude scales ’sone’ and ’mel’ are introduced and consequences of psychoacoustic results for auditory display will be summarized.

4.2.1 Perception of Loudness

Sound perception bases on the excitation of haircells in the inner ear. The more the amplitude of an input sound increases, the more haircells are excited. The intensity of this excitation is perceived as loudness. Given a sound by its pressure[2] $p(t)$, the sound intensity I is given by

$$I = \frac{\tilde{p}^2}{Z_0} \tag{4.1}$$

where $Z_0 = 415\,\mathrm{Pa\,s\,m^{-1}}$ is the impedance of air. $\tilde{p}$ is the effective sound pressure. For pure sine waves it is related to the peak amplitude $\hat{p}$ by $\tilde{p} = \frac{1}{\sqrt{2}}\hat{p}$. The intensity measures the energy that passes a unit area perpendicular to sound propagation per second. According to the Weber-Fechner law, loudness is proportional to the logarithm of sound intensity I. Thus, for measuring loudness usually the sound pressure level

$$L = 20\log_{10}\frac{p}{p_0}\mathrm{dB} = 10\log_{10}\frac{I}{I_0}\mathrm{dB} \;\text{ with }\; I_0 = 10^{-12}\frac{\mathrm{W}}{\mathrm{m}^2} \;\;,\;\; p_0 = 20\mu\mathrm{Pa}\;. \tag{4.2}$$

is used which compares the sound intensity to a reference value I_0 on a logarithmic scale. A sound wave of frequency 1 kHz and intensity I_0 just exceeds the threshold in quiet. Figure 4.3 shows the threshold in quiet (the dashed line) as a function of the frequency. Obviously, perception of loudness depends very much on the frequency. The line at the top shows the limit of damage risk. Obviously our ear is able to cover a dynamic range of about 120 dB. To compare loudness, the unit phon is used, which measures the subjective loudness level for sound compared with a 1 kHz sine tone. For pure tones of 1 kHz, the phon scale coincides with the sound pressure level L measured in dB. A sound has the loudness of x phon if the perceived loudness level is the same as the perceived loudness level of a 1 kHz tone of x dB. Figure 4.3 shows a number of

[2] This is the deviation from constant barometric air pressure.

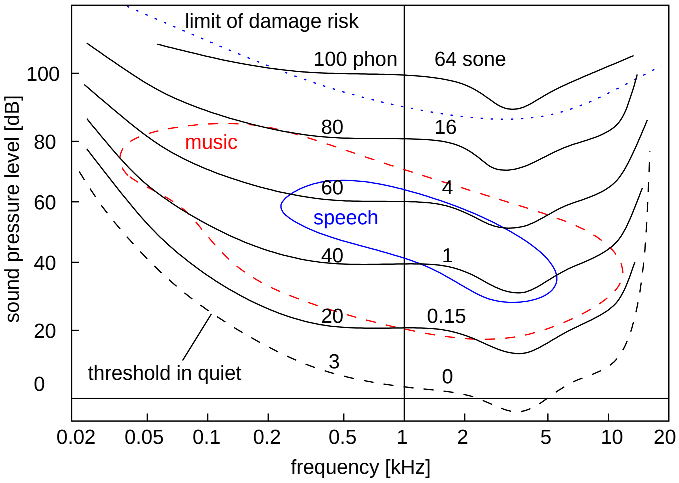


Figure 4.3: Hearing area. Threshold in quiet and isophones are shown. Typical regions for music and speech are outlined. Modified from [ZF99].

curves of equal loudness levels, which are called isophones. However, the phon scale does not provide information about the quantitative relations between the loudness of two tones. Therefore the sone scale is used, which bases on experiments where loudness ratios are adjusted by human subjects. As a gauging, the loudness of a pure 1 kHz tone of 40 dB is set to 1 sone. Sound with 2 (resp. 4) sone is thus perceived as twice (resp. four times) as loud. Figure 4.3 shows also some sone/phon pairs.

These magnitudes are defined only for pure tones. However, sound usually shows a much more complex spectrum. It is much more difficult to find a definition of loudness for frequency mixtures, since the ear integrates local signals on the basilar membrane with the result that a doubled loudness of two pure tones is only perceived if their frequencies are within a certain range, otherwise not their loudness but their intensities are added. The range itself is called frequency group. Its width Δf_G depends on the frequency as shown in Figure 4.4. The loudness of a sound also depends on its duration: with increasing duration of a sound the loudness stabilizes after about 300 ms [ZF99].

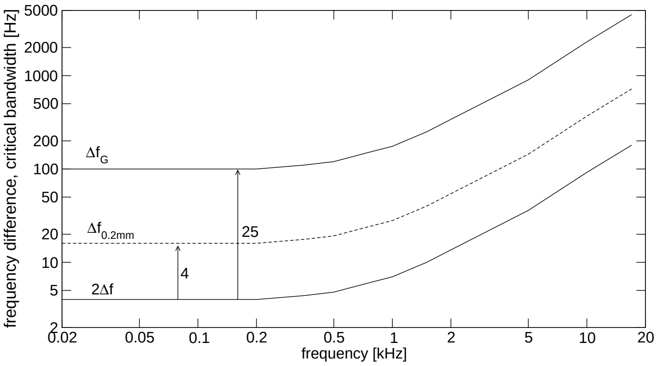


Figure 4.4: Frequency group width Δf_G, frequency step $2\Delta f$ and frequency $\Delta f_{0.2\mathrm{mm}}$, which is required to shift the envelope maximum on the basilar membrane by 0.2 mm, as a function of frequency. Modified from [ZF99].

Resolution of Loudness Perception

Throughout the whole dynamic range the ear is capable of perceiving level changes of about 1 dB [ZF99]. Experimentally, this is measured by presenting amplitude modulated tones to human subjects, whose task it is to signal when they perceive the modulation. Usually a sine function of $f_m = 4$ Hz is used. For a given degree of modulation m the sound signal is

$$s(t) = \hat{s}\sin(2\pi f_0 t)(1 + m\sin(2\pi f_m t)) \ . \tag{4.3}$$

Thus the sound level L varies by

$$\Delta L = 10\log_{10}\left(\frac{I_{max}}{I_{min}}\right) \text{ dB} = 20\log_{10}\left(\frac{1+m}{1-m}\right) \text{ dB} \ . \tag{4.4}$$

Figure 4.5 shows the perceptional threshold for amplitude differences for $f_0 = 1$ kHz. Obviously, our resolution is increasing the louder the sound becomes. For white noise, the degree of modulation is rather constant at level differences of 0.75 dB, corresponding to $m = 4\%$. Throughout the whole spectrum, loudness sensitivity resembles the function shown for the 1 kHz tone.

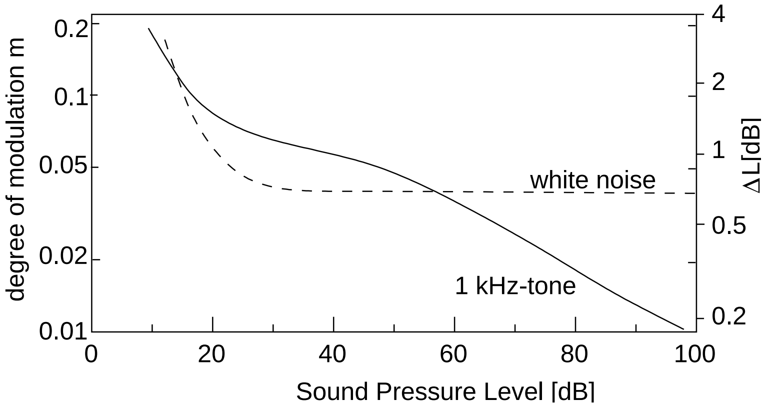


Figure 4.5: Resolution of loudness perception as a function of the sound level (modified from [ZF99]).

For auditory display, the results on loudness perception indicate that rather loudness than amplitude should be used as acoustic attributes, which means to determine a frequency dependent level offset. However, it is easier to restrict the frequencies to a range where the isophones are rather horizontal. This is given in the range from 400 Hz to 4000 Hz.

4.2.2 Perception of Pitch

As described in Section 4.1, the ears perform a frequency decomposition. Thus different haircells are responsible for different frequency parts of a sound signal. The frequency of sound is (for pure tones) perceived as pitch. As shown in Figure 4.3, tones between 50 Hz and 20000 Hz can be perceived. In contrast to the perception of loudness, which is an intensity perception, pitch is a positional perception [Zwi82]. This implies, that a mixture of differently pitched tones can be resolved even if the tones agree on all other perceptional qualities. The property of pitch being a positional perception has consequences on the possible methods for measuring perceptional functions: positional perception functions can be either determined by the stimulus steps, or from the measurements of pitch ratio.

In the musically relevant range up to 1500 Hz, the perceived pitch is doubled with a doubling of the frequency. This stops to hold true for frequencies beyond 1500 Hz. Human subjects perceive a tone of 8 kHz as twice as high than a tone of 1.3 kHz. From such psychoacoustic measurements a scale of ratio pitch $rp(f)$ can be derived, such that $rp(f_2)/rp(f_1)$ is the perceived pitch ratio for presented frequencies f_1, f_2. Ratio pitch was assigned the unit "mel" as it is related to our sensations of melody [ZF99]. The function $rp(f)$ is shown in Figure 4.6. In the musically relevant part (100 Hz – 1500 Hz), $rp(f)$ can be linearly approximated. The perception of pitch is furthermore dependent on the sound pressure level [ZF99]. But this effect is rather weak so that it is mostly neglected.

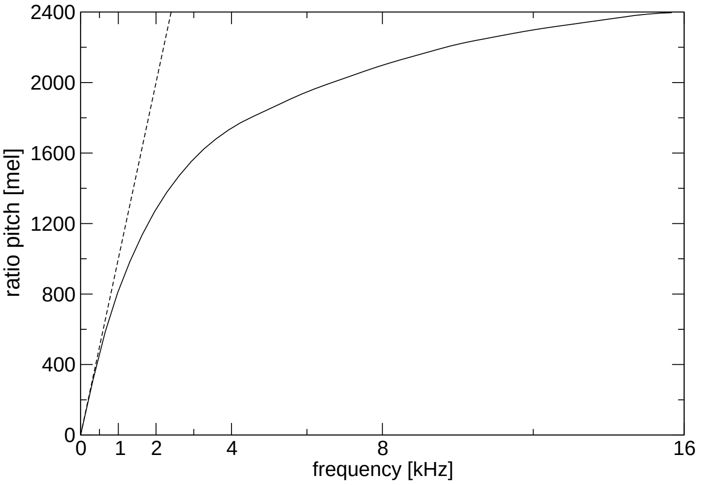


Figure 4.6: Ratio pitch as a function of frequency. Modified from [ZF99]

.

For sounds more complex than pure sine, which almost any musical instrument normally generates, the perceived pitch may differ from the lowest frequency. The phenomenon of virtual tonality can only be mentioned here: if a periodic signal is presented which has no energy in its fundamental vibration, nonetheless the perceived pitch corresponds to this frequency. This exemplifies that the perceptual functions are much too complicated to be extrapolated from the results on pure sine tones.

Resolution of Pitch Perception

The resolution of pitch is a frequency dependent function $\Delta f(f_0)$ which means that two tones differing by more than Δf can just distinguished by their perceived pitch. Analogous to the method of measuring the resolution of loudness perception, it is measured by presenting modulated sine tones to subjects. For this, frequency modulation is used. Figure 4.4 shows $2\Delta f(f_0)$ for sine tones with a constant loudness of 60 phon. For frequencies higher than 500 Hz, the resolution is almost proportional to the frequency: Throughout this range, frequency changes of 0.7% can just be perceived.

For auditory display design, the results on pitch perception indicate that the frequency variable is not suited to linearly map data as frequency resolution gets smaller with rising frequency. Using a relation frequency = exp(const × data) circumvents this. Also, if pitches should be compared, it is recommended to restrict the frequency range to about 2000 Hz. Below that, the mel scale is almost linear to frequency. In this work, pitch is used as acoustic attribute related to frequency by $f = f_0 2^p$, so that a shift of 1 octave is a pitch shift of 1. $f_0 \approx 1.0919$ is used so that $f(8.75) = 440$ Hz.

4.2.3 Perception of Timbre

In contrast to pitch and loudness, timbre is a multidimensional perceptual quality. It is defined by exclusion: when two sounds have the same pitch and loudness, timbre is the property that makes them distinguishable [Ass60]. Timbre is related to sensations like roughness, harshness, sonority, brilliance, hardness. However, it is difficult to associate such descriptions with properties of the stimuli. In music, timbre is described by the instrument type, e.g. as timbre of a clarinet or a violin.

Psychoacoustics only takes a few aspects of timbre into account like roughness or brightness [ZF99]. In general two types of signal properties which influence the perceived timbre can be discerned: (i) spectral composition of a sound and (ii) temporal evolution of a sound.

Property (i) was first used by Helmholtz to describe timbre for musical instrument [vH54]. In his opinion the Fourier series coefficients, resp. the harmonics of a periodic signal give perceptual clues that determine timbre. Thus the timbre description is at least as high-dimensional as the number of utilized harmonics. However, this timbre representation neglects the temporal evolution of energy in the harmonics, the relevance of this for understanding timbre was shown by Risset [RM69].

Property (ii) is known as amplitude envelope of the sound. For its description, mainly the attack time and decay time are specified. Sounds of musical instruments differ significantly in their envelopes. However, instrument sounds do have acoustic properties that cannot be uniquely associated with one of these types. These are low-frequency modulations like vibrato, tremolo or nonlinear distortions which both affect level and spectrum.

The advances in computational sound synthesis now allow to control every aspect of sounds with the highest precision. This is the prerequisite for a systematic investigation of timbre and its perception.

Experimentally, timbre is approached by doing dissimilarity tests. Subjects are asked to judge the dissimilarity of sound-pairs drawn from a database. Projection techniques are applied to find a suited low-dimensional representation of the sounds fitting to the collected dissimilarity scores. The dimensions obtained are regarded highly relevant for explaining timbre and the sounds along the axes are investigated to find suited timbre attributes[3] However, most studies on timbre focus on the sounds of musical instruments. They do not consider the important class of everyday sounds.

For auditory display this has the consequence that current timbre research gives only limited guidelines for the selection of acoustic attributes. Brightness or roughness may be used as timbre attributes in auditory display. Also, timbre may be used to represent categorical variables, e.g. by using a set of very distinct timbres for the representation of different variable values. Controlling the temporal evolution of sound from data allows to increase the display dimensionality but it is impossible to determine in advance if the mappings can be discerned or how the perceptual salience compares to other timbre parameters. The high complexity of timbre makes it so difficult to understand, and the best recommendation for a designer of an Auditory Displays is to experiment with timbral controls until a solution is found that subjectively allows to distinguish the parameters from sound.

[3] See http://www.diku.dk/research-groups/musinf/krist/perc.html

4.3 Auditory Scene Analysis

The previous sections focused on simple stimuli like pitch or the loudness of sine tones. With sonification and auditory display, however, we aim at generating complex sounds where potentially a lot of sonic events with a lot of acoustic attributes are mixed. This section addresses the question how we perceive such complex sounds. Synthetic and analytic listening will be introduced and the concept of auditory streams and some auditory group principles will be presented.

The *auditory scene* is the sound that arrives at our eardrums and which usually includes a mixture of the signals from all sound sources within our environment [Wil94]. *Auditory Scene Analysis* (ASA) aims at understanding the process of decoding the auditory scene into separate auditory streams. This takes place in our auditory system [Bre90]. If listeners for instance are interested in a single auditory stream of sound events, e.g. the melody of one instrument in a piece of music or a person talking at a cocktail party, this single stream has to be isolated from the mixture in order to make sense of it. An *auditory stream* is a perceptual object, a perceptual grouping of different parts of a sound signal that belong together. The way in which auditory streams are constituted from the auditory scene depends on the configuration and the way of listening. As an example, a single singing voice in a chorus can either form an auditory stream or can be integrated into a single stream which represents the whole chorus.

Our perceptual apparatus is excellent in performing the task of stream segregation and to do so applies a set of grouping principles. A nice visual example from Bregman [Bre90] for source demixing shall illustrate the demixing of two streams:

AI CSAITT STIOTOS

$A_I\ C_S A_I T_T\ S_T I_O T_O S$

Obviously, visual factors assist segregation into “a cat sits” and “i sit too”. Similarly, in auditory perception, a mixture of independent signal arrives at the eardrums. *Auditory Grouping* comprises all spectral and temporal processes that operate to assign parts of the signals to auditory streams. It depends on the focus of attention. The distinction between analytic and synthetic listening is made for this purpose. *Analytic perception* aims at focusing on the maximal information of one stream. Following the voice of a single instrument in an orchestra is a good example for this. *Synthetic perception* (also called holistic perception) aims at perceiving the auditory scene as a whole, e.g. following the piece of music rather than one instrument alone. Both types of perception usually interact and depend further on training of the listener.

As the visual example above shows, certain factors can help to segregate streams. Gestalt Psychology identifies features that promote the binding of signal parts together. Gestalt Principles like similarity, proximity, good continuation, belongingness, common fate and closure have been investigated, mainly for the purpose of vision research. However, they also exist in auditory perception [Wil94]. The most important gestalt principles in the auditory domain are:

- **Similarity:** components are perceived as related if they share the same attributes.
- **Proximity:** The closer components are together, the more likely their grouping is (e.g. frequency proximity).
- **Good Continuation:** a smooth transition between two separated components lets them be perceived as related. Example: A noise burst which separates a gliding tone nonetheless lets the two fragments be perceived as one stream.

- **Common Fate:** Components that share a common kind of change are more likely to be grouped together. Two tones whose frequency ratio is an integer are grouped within one auditory stream, interpreting the one tone as an overtone of the other. This is more likely to happen if the onset, movement and end of the tones are related.
- **Closure:** Incomplete forms tend to be completed. The perception of virtual pitch is an example: the pitch of the fundamental frequency is perceived in a mixture of overtones even if the fundamental frequency does not exist within the spectrum.

This list is not complete. For a profound discussion of these topics see [Bre90, Wil94]. The consequences of the existing knowledge on auditory display is summarized by Williams:

> The lack of a complete theory of human auditory perception and the intrinsic variability between human listeners makes it impossible to predict the interpretation of any given acoustic signal on the basis of existing knowledge. This implies that the development of Auditory Display techniques is necessarily experimental, requiring validation through user evaluation.

From another perspective, the conclusion can be seen in a much more optimistic way: obviously, auditory perception has evolved to solve the problem of stream segregation from environmental sound mixtures. This allows us to compose multiple auditory streams in much the same way as they are combined in our natural environment. The effortless stream segregation is exploited to help the listener to make sense of such mixtures of sounds. The processes of auditory stream segregation aim at demixing the auditory scene into distinct sources. Ideally, each physical sound source is associated with one stream, so that the listener can extract all relevant information for source identification. Arguing, that our auditory system is adapted to interprete sounds of physical object, one implication can be, that we improve the performance of our auditory system for the interpretation of auditory display, if the sounds in an Auditory Display are synthesized and mixed according to the same principles as environmental sounds, thus using physical models for sound generation.

Chapter 5

Acoustics

This chapter will pay particular attention to basic physical foundations to describe acoustic systems and their behavior. For this, some simple systems will be discussed which are important building blocks for sonification models developed in Chapter 8. Basic knowledge about the qualitative vibrational behavior of physical objects also provides ideas about how their sounds can be synthesized. This will be discussed in Chapter 6.

5.1 Introduction

Some basic physical laws are summarized here to facilitate their later usage. For the description of physical acoustic phenomena, a system must be described by its constituents and the forces that act on them. In classical physics, bodies are located in 3d space. In the context of sonification models it will be useful later to generalize to spaces of an arbitrary dimension d.

The simplest kind of body is a point mass which is fully described by its mass m and its location $\mathbf{x}$. Without forces, the motion of this body is unchanged, that is $\ddot{\mathbf{x}} = 0$. Forces cause a change in the motion of a body according to

$$\mathbf{F} = \frac{d}{dt}(m\dot{\mathbf{x}}) \; . \tag{5.1}$$

The work done by a force is the integrated scalar product between the applied force and the distance through which the body moved

$$W = \int \mathbf{F}\,\mathbf{ds} \; . \tag{5.2}$$

If the force is used to increase the velocity of a point mass, the energy is stored as kinetic energy. The kinetic energy of a point mass moving with velocity v is given by $W = mv^2/2$ where $v = \|\dot{\mathbf{x}}\|$. If the work is used to overcome forces inherent in a system like lifting a mass against gravitation, the energy is stored as potential energy V. If there are frictional forces, the work to be done to reach a position $\mathbf{y}$ starting at $\mathbf{x}$ is dependent upon the way. In ideal cases we neglect friction and then V only depends on the position of the end points of motion and a potential function V is given. A given potential V causes a force on a point mass given by

$$F = -\nabla_{\mathbf{x}} V \; . \tag{5.3}$$

If frictional forces can be neglected and the system is isolated, the total energy is conserved and given by

$$W_{tot} = V(\mathbf{x}) + W_{kin} = V + \frac{1}{2}m\dot{\mathbf{x}}^2 = \text{const.} \tag{5.4}$$

These basic relations suffice to treat simple systems like the oscillator.

5.2 The Linear Oscillator

The simplest kind of system showing vibratory motion is a point mass fastened to a massless spring of stiffness k so that it moves back and forth in one direction. As many systems can be reduced to this sort of system, it is discussed here in greater detail. To simplify the system further, we restrict the restoring force to be proportional to the deviation from the equilibrium point (which can be set to $x = 0$) so that

$$F(x) = -kx\ . \tag{5.5}$$

If x is small enough, an arbitrary $F(x)$ with $F(0) = 0$ can be approximated by the linear term of the Taylor expansion of $F(x)$. k is called the stiffness constant. Assuming that no other forces act on the system, the equation of motion is

$$m\ddot{x} = -kx \Leftrightarrow \ddot{x} = -\omega^2 x \text{ with } \omega^2 = \frac{k}{m} \tag{5.6}$$

The solution of this differential equation is given by

$$x = Ce^{-i\omega t} \text{ or as } x = a_0 \sin(\omega t + \phi) \tag{5.7}$$

using the convention that for scalar variables the real part of complex numbers is used. The complex number C cares for the initial phase and the amplitude of the oscillation. The physical implications of the solution in eq. (5.7) is that the frequency[1] ω is independent of the amplitude and determined only by the mass and the stiffness. Usually, we can describe the initial conditions of the linear oscillator by an initial displacement x_0 and an initial velocity v_0. These two conditions fully determine the evolution of the system by

$$\begin{aligned} a_0^2 &= x_0^2 + \left(\frac{v_0}{\omega}\right) && (5.8)\\ \tan(\phi) &= \frac{v_0}{\omega x_0} && (5.9)\end{aligned}$$

as can be shown by inserting $t = 0$ into eq. (5.7), considering its derivative and solving for a_0^2, resp. ϕ.

The energy is the sum of the kinetic energy and potential energy

$$W = \frac{1}{2}m\dot{x}^2 + \int_0^x kx\,dx = \frac{1}{2}m\dot{x}(t)^2 + \frac{1}{2}kx(t)^2\ . \tag{5.10}$$

Obviously, the energy is periodically converted from kinetic energy (when the mass is at the origin) to potential energy (when the mass is at a turning point).

[1] Angular frequency ω is related to frequency by $f = \omega/(2\pi)$!

5.3 Damped Oscillations

The linear oscillator discussed above did not possess any damping. This means, no energy leaves the system, not even energy in form of sound. For any system, sound radiation can be regarded as friction which leads to a diminishing of the system's energy. This resisting force depends on the velocity of the oscillator and is almost proportional to the velocity. This may be expressed as

$$F_{\text{fric}} = -R\dot{x} \ . \tag{5.11}$$

R is called resistance constant. Including the friction force into the force equation we get the following equation of motion

$$\ddot{x} + 2\gamma\dot{x} + \omega_0^2 x = 0 \ \text{ with } \gamma = R/(2m) \ \text{ and } \omega_0 = k/m \ . \tag{5.12}$$

ω_0 is the frequency that the oscillator would have without friction and is called the natural frequency of the system.

The general solution for eq. (5.12) is $x(t) = Ce^{bt}$. Inserting $x(t)$ and its derivatives into the equation of motion, we see that

$$b^2 + 2\gamma b + \omega_0^2 = 0 \ \Rightarrow \ b = -\gamma \pm \sqrt{\gamma^2 - \omega_0^2} \tag{5.13}$$

must be fulfilled. In many practical situations, k is much larger than R and thus the root in eq. (5.13) is imaginary and the solution can be written as

$$\begin{aligned} x(t) &= Ce^{-\gamma t}e^{-i\omega_f t} \ \text{ with } \omega_f = \omega_0\sqrt{1 - \left(\frac{\gamma}{\omega_0}\right)^2} \ \text{ or} && (5.14) \\ x(t) &= A_0 e^{-\gamma t}\cos(\omega_f t - \phi) && (5.15) \end{aligned}$$

This obviously is an aperiodic motion, decaying exponentially to A_0/e in time γ^{-1}. The amplitude $A(t) = A_0\exp(-\gamma t)$ is a function of time. The physical implications of the solution are, that same as in the undamped oscillator, frequency is independent of the amplitude of motion. We can see further that damping decreases the 'frequency' of the motion ω_f compared to the natural frequency.

The energy of the system, computed as the sum of kinetic and potential energy of the point mass now is a function of time

$$E(t) = \frac{1}{2}m\dot{x}^2 + \frac{1}{2}kx^2 \propto e^{-2\gamma t}e^{i\omega_f t} \ . \tag{5.16}$$

Averaging E over a single oscillation, we get approximately $E(t) \propto e^{-2\gamma t}$. The effect of friction is the same in more complicated systems: the frequencies are slightly diminished and amplitude and energy decays exponentially with time.

5.4 Forced Oscillations

A frequent situation is that an oscillatory system is connected to an external oscillatory system which drives the system under examination. For instance, a loudspeaker diaphragm vibrates due to a linkage to the output circuit of an amplifier. The feedback from the system onto the external system shall be neglected assuming a much larger energy reservoir in the external system.

An excitation of a system (e.g. plucking, striking) can be regarded as a special case following the general discussion where the system reaction of an oscillator on a periodic external force $F_{ext} = \hat{F}e^{i\omega t}$ is examined.

The equation of motion is modified by the external force to

$$\ddot{x} + 2\gamma\dot{x} + \omega_0^2 x = F_{ext}/m = ae^{i\omega t} \text{ with } a = \hat{F} \tag{5.17}$$

The general solution is guessed to be

$$x = Ce^{-\gamma t - i\omega_f t} + De^{i\omega t} \; . \tag{5.18}$$

Substitution of the solution in eq. (5.17) gives (see [MI68])

$$D = \frac{\hat{F}}{-i\omega Z_m} \text{ with } Z_m = R - iX_m \text{ and } X_m = \omega m - \frac{K}{\omega} \; . \tag{5.19}$$

Z_m is the mechanical impedance of the system and equals the ratio between the driving force and the velocity of the steady-state motion after the free-oscillation has died out. Same as in the “free” system, the free oscillation dies out with time constant γ^{-1}. The steady state amplitude $|D|$ as a function of ω shows a resonance peak when $\omega^2 = \omega_0^2 - 2\gamma^2 \approx \omega_0^2$. Replacing complex numbers by real numbers, the solution writes as

$$x = A_0 e^{-\gamma t}\cos(\omega_f t + \phi_f) + \frac{\hat{F}}{\omega|Z_m|}\sin(\omega t - \theta) \tag{5.20}$$

where $\tan(\theta) = X_m/R_m = m(\omega^2 - \omega_0^2)/(\omega R_m)$. This implies that for excitation frequencies less than the resonance frequency the phase difference between the external force and the displacement of the oscillator is close to 0. At the resonance frequency, the phase difference is just $\pi/2$. For higher frequencies, the phase converges to π.

In practically relevant situations, the external force is often a more complicated function than a simple oscillatory function. However, as the equation of motion is linear, superposition of solutions for sinusoidal driving forces yield a solution for the superposition of the forces. Given a driving force $f(t)$, the solution for the displacement $x(t)$ can be computed using its Fourier transform

$$F(\omega) = \frac{1}{2\pi}\int f(t)e^{i\omega t}\,dt \; . \tag{5.21}$$

The solution $x(t)$ is given by

$$x(t) = \int X(\omega)e^{-i\omega t}\,d\omega = \int \frac{F(\omega)}{i\omega Z(\omega)}e^{-i\omega t}\,d\omega \; . \tag{5.22}$$

A more detailed discussion is found in Morse [MI68].

5.5 Impact Sounds

Impact sounds are ubiquitous in our environment, as they are produced by any colliding physical objects. The sound depends on the material of the bodies, their shape, the contact location and the force of the impact. Listeners are able to extract valuable information from such sounds which indicates that they may also prove useful in auditory display [Gav93b]. The theory of elastic waves in solids is much too complex to be presented here in detail. However, some concepts and notions will be introduced in this section and the qualitative behavior of solids on physical excitation will be described, because these kind of sounds are used in some sonification models that will be presented later.

When a physical object is hit or struck, a deformation propagates through the body. It is reflected at the borders of the object and thus causes surface vibrations and sound emittance. The shape, size and material affects the modes of vibration which determine the characteristic spectrum of the body. The stiffness of the material determines the spectral centroid and the mechanical resistance determines the decay rate of the vibrations. The location of the impact determines the distribution of energy to the modes while the force influences the overall sound level. Nonlinear response of the body during the initial phase can cause other frequencies to contribute during the transient phenomenon.

To represent characteristic properties of object sounds, the example of a 2-dimensional rectangular membrane of length L_x along the x-axis and L_y along the y-axis under constant tension, is presented. A detailed discussion is found in the articles of Pai [vdDP98] and Morse and Ingard [MI68]. A membrane surface is described by a function $s(x, y, t)$ which represents the deviation from equilibrium. s obeys the wave equation

$$\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} - \frac{1}{c^2}\frac{\partial^2}{\partial t^2}\right) s(x, y, t) = F(x, y, t) \tag{5.23}$$

where c is the wave propagation velocity and $F(x, y, t)$ specifies an external force. Usually some boundary conditions are given which restrict the possible deformations s, e.g. that $s = 0$ on the edges. The initial state of the surface needs to be defined. For this, it suffices to determine $s(x, y, t)$ and the initial velocity $\dot{s}(x, y, t)$. In rest, both are 0. For the force-free system where $F(x, y, t) = 0$, the solution is

$$\begin{aligned} s(x, y, t) &= \sum_{m,n} a_{mn}\psi_{mn}(x, y)\cos(\omega_{mn}t - \Phi_{mn}) \\ \text{with } \psi_{mn}(x, y) &= \sin\frac{\pi m x}{L_x}\sin\frac{\pi n y}{L_y} \\ \text{and } \omega_{mn} &= \pi K\sqrt{\left(\frac{m}{L_x}\right)^2 + \left(\frac{n}{L_y}\right)^2}, \end{aligned}$$

K depending on the tension. The functions $\psi_{mn}()$ are called *normal modes* and ω_{mn} the mode frequencies. Any initial state $s_0(x, y, t)$ can be expressed as a superposition of the normal modes. The coefficients a_{mn} are easily computed by projecting $s_0(x, y, t)$ onto the normal modes. So far the vibration is undamped. A friction term can be introduced which causes the amplitudes to decay exponentially [MI68].

This special case shows characteristic properties which are also valid for more complicated systems. In summary one can say that the sound of vibrating solid bodies can be modeled as a superposition of decaying sines. Excitation of an object (e.g. by a collision with another object, or by being struck) modifies the displacement. It is possible to compute the amplitude of the normal modes from the displacement function. Increasing tension or decreasing size increases the mode frequencies, and increasing damping reduces the decay time. For a more detailed introduction, the reader is referred to the excellent textbook of Morse and Ingard [MI68].

Chapter 6

Sound Computing

In this chapter I will give a brief outline of techniques adopted to synthesize, transform and reproduce sound in a computer. Sound synthesis is an integral part of any sonification. This section summarizes the most commonly applied synthesis algorithms and their parameters. The chapter starts with a short introduction to different representations for audio signals. A number of sound synthesis algorithms will be presented in Section 6.2. This will be followed by some remarks on sound post processing which for instance is required to simulate room acoustics or to control the spatialization of sound sources.

6.1 Sound Representation

Sound is nothing else but air pressure variations. To record any sound it is sufficient to measure the time-variant air pressure function $s(t)$. As described in Section 4.1, variations of $s(t)$ in the frequency range between 50 Hz and 20000 Hz are audible for human ears. For storage within current computers, a quantized representation of the sound as a vector is used. Quantization is performed both on the time axis and on the values. To digitally record sounds, the analog signal $s(t)$ is sampled at equally spaced points in time. The sampling rate ν_{SR} determines the upper limit for recorded frequencies to $\nu_{Nq} = \nu_{SR}/2$, the Nyquist frequency. Therefore it is necessary to remove any frequency parts above ν_{Nq} with filters before recording or resampling a signal. Value quantization is done due to the bit resolution of the data format. The signal distortion due to value quantization is audible as quantization noise. The signal/noise (S/N) ratio of quantization noise in decibel is direct proportional to the number of bits per value. For compact disk audio recordings, a sampling rate of 44100 Hz and a sample data format of 16 bit integers is used, yielding a S/N ratio of about 96 dB.

Humans have two ears, so that at least two audio channels (stereo) are required to imitate arbitrary auditory scenes. However, since real sound is manipulated through reflections at the outer ear and by this also includes spatial clues about the sound source, special recording techniques (using microphones within the ears of an dummy head) are required to catch these clues. For a standard stereo audio amplification system either using two speakers or headphones, the control over the perceived spatial location of the virtual sound source is limited. Using headphones, spatial clues like the interaural time delay and the interaural level difference must be included into audio signals to allow source localization. In a loudspeaker setup, intensity panning is possible which allows to position the virtual sound source within the triangle spanned by the two speakers and the listener. This is achieved by a proper distribution of a mono audio signal into two speakers.

In this thesis a sound signal is represented by a series of sample frames $\{\mathbf{s}_1, \mathbf{s}_2, \dots\}$. A sample frame $\mathbf{s}_i$ is a vector containing the sample values for all available channels. Such a multichannel audio signal is stored frame by frame in memory in standard C float format.

Sound is mathematically described by a continuous function $s(t)$. Then the time-discrete signal is computed by $s[n] = s(n/\nu_{SR})$. The time series representation is the most direct representation of the sound and usually the format required for sound output. For manipulations and inspection, other representations are sometimes preferred. In this work the spectrum and the short time Fourier spectrum (STFT) are used.

The spectrum is computed by a Fourier transform, given by

$$\tilde{s}[k] = \sum_{n=0}^{N} s[n] \exp\left(\frac{2\pi ikn}{N}\right) \tag{6.1}$$

where N is the number of all samples available [LO88]. Assuming $s[n]$ to be sampled at a sampling rate ν_{SR}, then $|\tilde{s}[k]|$ represents the Fourier amplitude for frequency $f_k = k\nu_{SR}/(2N)$. The spectrum is suited for all kinds of signal filtering. For instance, a low-pass filter can be simply realized by setting the upper elements of $\tilde{s}[k]$ to zero. The temporal evolution of the sound, however, is hidden in the phases of the complex numbers $\tilde{s}[k]$ and cannot be controlled easily using this representation. Practically, the spectrum is computed by using a Fast Fourier Transform (FFT) algorithm.

The discrete short-time Fourier transform (STFT) combines a time-domain and frequency-domain representation into a single framework. The STFT consists of a separate Fourier transform for equally spaced time points. The data used for the Fourier transforms are windowed segments of the time series. For both discrete time and frequency, the discrete STFT is defined as

$$S[n,k] = \sum_{m=-\infty}^{\infty} s[m]w[n-m] \exp(-i2\pi km/N) \tag{6.2}$$

where $k < N$ denotes the frequency channel and $w[\cdot]$ is an analysis window. Usually some decimation is done, only storing the sequence $S[Ln,k]$ for all n. The decimation factor L is the step width between successive analysis windows. For analysis windows of width N_w, a decimation factor of less than $L = N_w/2$ allows complete reconstruction of the original signal from the STFT representation [LO88].

The discrete inverse STFT (also called Short-Time Fourier Synthesis)

$$s[n] = \frac{1}{W_0} \sum_{p=-\infty}^{\infty} \left[\frac{1}{N} \sum_{k=0}^{N-1} S[p,k] \exp(i2\pi kn/N) \right] \tag{6.3}$$

with $W_0 = \sum_{n=-\infty}^{\infty} w[n]$ can be used to reconstruct the time series from the data. Using a decimation factor L, it can be shown that $s[n]$ can be reconstructed from

$$s[n] = \frac{L}{W_0} \sum_{p=-\infty}^{\infty} \left[\frac{1}{N} \sum_{k=0}^{N-1} S[pL,k] \exp(i2\pi kn/N) \right] \tag{6.4}$$

presumed that the analysis window satisfies $\sum_p w(pL-n) = W(0)/L$ for all n. The synthesis technique is called Overlap-Add (OLA), since reconstructed signal pieces are superimposed at their corresponding window center positions. For details on STFT, see [LO88].

6.2 Sound Synthesis

Sound Synthesis is such an extensive field that only the most important algorithms can be tackled. The different techniques are briefly summarized and pointers into literature are given.

6.2.1 Additive Synthesis

The oldest sound synthesis technique is that of additive synthesis. The sound signal is computed by

$$s(t) = \sum_i a_i(t) \sin(2\pi f_i t + \phi_i) \ . \tag{6.5}$$

Sine waves with frequency f_i are multiplied by an amplitude envelope $a_i(t)$ and superimposed and stored into $s(t)$. The sounds of many musical instruments and physical objects show an almost periodic waveform where the total amplitude changes only very slowly in comparison to the oscillation time. The spectrum of periodic functions consists of frequencies which are integer multiples of a fundamental frequency. Such sounds can be approximated by additive synthesis using

$$s(t) = a(t) \sum_i a_i \sin(2\pi i f_0 t + \phi_i) \ . \tag{6.6}$$

In this synthesis model, amplitude envelope $a(t)$ and timbre are separated. The vector $\mathbf{a} = (a_1, \dots, a_n)$ shall be denoted as timbre vector. A timbre can be composed by superposition of some timbre vectors, e.g. a, b by

$$s(t) = a(t) \sum_i (\lambda_a a_i + \lambda_b b_i) \sin(2\pi i f_0 t + \phi_i) \ . \tag{6.7}$$

which controls the timbre space with only few parameters. Such parameters $\lambda_a, \ \lambda_b$ may be used for instance as acoustic attributes in Parameter Mapping Sonifications.

Using a time-variant function $f_i(t)$ for the frequency allows continuous pitch changes which for instance occur in a trombone or if a violin is played with vibrato. In the latter case, f_i oscillates around a center value with a vibrato frequency much smaller than f_i. The amplitude Δf_i and the vibrato frequency both are suitable parameters for sonification.

The special case of a single oscillator using a time-variant frequency is called time-variant oscillator. The sound signal is given by

$$s(t) = a(t) \sin(f(t)t + \phi) \ . \tag{6.8}$$

Time-variant oscillators are used in several sonifications, e.g. in EEG data sonifications in Section 10.2. Using a superposition of n such time-variant oscillators results in a sound mixture, which is usually separated into different auditory streams by the auditory system unless both the amplitude and frequency functions share a common fate.

Additive synthesis models are well understood and efficient to implement in a computer. However, to mimic realistic sounds, complicated trajectories in parameter space have to be conducted. A special problem for this type of synthesis is the modeling of noisy sounds, which frequently occur during the attack phase of many instruments. This is the case, because a frequency continuum is needed to represent noise and thus many oscillators are required. Here, other synthesis techniques are superior and may be used to fill this gap by hybrid synthesis algorithms.

So far, the sine function was used as the building block for the signal. Computation is avoided in practical implementations, and replaced by a table lookup and interpolation. The table holds one period of a periodic signal, called waveform. The table-lookup oscillator in this sense is the “workhorse” of the whole field of computer music [Moo90] and exists in many commercial synthesizers. As a natural generalization, additive synthesis can be performed with arbitrary waveforms using table-lookup synthesis.

For the implementation of sound synthesis, the synthesized frequencies have to be considered carefully, since sampling the sine function with frequencies f higher than the Nyquist frequency ν_{nq} results in frequency aliasing, also called fold-over: a frequency gets audible at $2\nu_{NQ} - f$, as a result of temporal quantization. Thus, the maximal frequency in the table determines an upper limit of the usable frequencies.

6.2.2 Sound Sampling

Sound Sampling may be regarded as a technique for sound composition rather than for synthesis. Recorded and digitized sound pieces, e.g. single tones of an instrument, are stored in a sound vector and reproduced by using table-lookup in combination with interpolation and other techniques for sound manipulation [Mas98]. Thus the sound is not completely synthesized but only played back. Contrary to table look-up synthesis, it is possible for the stored sample to contain much larger sound-pieces like whole spoken words or a drum beat inclusive its decay. This technique is presented here, as it is the mostly applied technique for the integration of auditory icons, and also because it is used for various sonifications in this thesis.

A disadvantage of sampled sounds is the fact that their pitch is determined by the recorded sound. Within a small range of about 30%, the pitch can be modified by changing the step increment for table lookup without reducing the quality, but increasing the pitch causes the envelope to be compressed. In a similar way, smaller step increments lengthen the sound. Therefore, many sound synthesizers use hybrid approaches, applying sampled sounds for the attack phase and superimposing table-lookup oscillator signals for the stationary part of an instrument tone. This allows a more flexible control over amplitude envelope and pitch.

6.2.3 FM-Synthesis

Sound Synthesis by Frequency Modulation became popular in the 80ies. It was introduced by Chowning [Cho73] for the purpose of sound synthesis. Frequency modulation means computing a sound signal by

$$s(t) = a(t)\sin(2\pi f_c t + I(t)\sin(2\pi f_m t)) \; . \tag{6.9}$$

This was mentioned above in Section 6.2.1 as a means of realizing vibrato using $f_m \ll f_c$. In contrast, for FM synthesis, the frequency of the modulator f_m is in the same order of magnitude as the carrier frequency f_c, usually taken a fixed multiple $f_m = \gamma f_c$. For a constant timbre, both the ratio $\gamma = f_m/f_c$ and the modulation index $I(t)$ are kept constant. Frequency analysis of the signal $s(t)$ reveals that frequency modulation basically “stoles” energy at the carrier frequency, spreading it to the sideband components at frequency $f^k_{1,2} = f_c \pm kf_m$ (for the kth sideband) whose amplitudes are determined by Bessel functions of the first kind and kth order $J_k(I)$ the argument of which is the modulation index. The larger the value of k, the higher the modulation index must be for that sideband to have significant amplitude. An important remark has to be made on “negative sideband frequencies” resulting for higher orders. Such frequency components are “reflected at the origin” which means that the phase is shifted by π and thus the sign changes. Details on FM-synthesis are found in [RS85]. In most applications, integers or small integer fractions are used for γ. Using $\gamma = 1$ results in sounds containing frequencies that are integer multiples of the carrier frequency. Sounds with such a spectrum occur in many musical instruments where higher harmonics (called overtones) arise from the vibration properties of solids. FM synthesis thus delivers a very simple model for the direct control of timbre complexity in terms of sound brightness through a single scalar parameter I. Parameters whose change implies the change of complex auditory attributes are very well suited for Parameter Mapping Sonifications. Specific modulation functions $I(t)$ allow to mimic several instruments, e.g. an electric piano with linearly decaying $I(t)$ or brass instruments with $I(t)$ starting at 0 and rising to a final value I_{max} within a constant attack time [RS85]. Several timbre classes can be realized by using other modulation ratios $\gamma = 2, 3, 1.414$, etc. The latter produces bell-like sounds, with $\gamma = 2$ the sound of organ pipe can be synthesized.

6.2.4 Nonlinear Synthesis

The previous synthesis techniques possessed a separate control of amplitude envelope and timbre. However, in most real instruments these two attributes are closely connected. When a whistle is blown stronger this results in a change of both timbre and the amplitude of the sound. Such connected behavior can be achieved with *Nonlinear Synthesis* which composes a sound from

$$s(t) = g(f(t)) \ , \tag{6.10}$$

using a nonlinear transfer function $g(\cdot)$ to manipulate the instantaneous amplitude of a source signal $f(t)$. For the input $f(t)$ any of the other sound synthesis algorithms can be used. To give an example, consider an exponentially decaying sine given by $f(t) = \exp(-t/\tau)\sin(2\pi\nu t)$. Without nonlinear distortion, the energy is concentrated at frequency ν. Using a nonlinear transfer function $g(x) = \arctan(2x)$, the peaks of the sine wave are softly clipped for high amplitudes of f. The transformed signal contains additional harmonics. With decaying amplitude, these harmonics experience a faster decrease in energy than the fundamental frequency. This leads to timbre evolution which is typical for many percussive systems. Given an input signal $f(t) = \cos(2\pi\nu t)$, transfer functions exist where the output signal is $\cos(2\pi k\nu t)$ for any $k \in \mathbb{N}$. These functions are the Chebychev polynomials, and representation of a transfer function as a superposition of different Chebychev polynomials allows us to predict the output spectrum for any input signal [Roa85b].

It is possible to extend Nonlinear Synthesis in many different ways like applying $g()$ to a mixture of sounds, or controlling the shape of $g()$ as a function of time with some parameter θ, e.g. the mixing coefficients for the polynomials. These offer rich possibilities for sound control and these might also be exploited in sonification applications.

6.2.5 Granular Synthesis

Granular Synthesis of sound composes a larger acoustic event from the superposition of thousands of very short sonic “grains”. Provided a massive amount of control data is specified, it is well suited for generating complex time-variant spectra. A physical motivation and example for Granular Synthesis can be given by the human voice where the vocal folds essentially produce a sequence of short pulses (grains). Their rate determines the fundamental pitch of the voice. Each of these excitations is manipulated by a linear filter or resonator given by the vocal tract (throat, tongue, etc.).

According to Gabor's acoustic theory [Gab47], a granular or quantum representation could describe any sound. The control parameters of each grain determine the spectral content whereas the temporal organization is controlled by the grain composition. Thus Granular Synthesis can be seen as a compromise between Fourier and time-domain synthesis techniques. In contrast to STFT synthesis (see Section 6.1) which obtains a complex sound by overlap-add superposition of equally spaced signal pieces in time, the grains of Granular Synthesis do not necessarily have to be located on a regular time grid. Stochastic grain distribution may be used where appropriate, e.g. to model rubbing or scraping sounds.

The resulting sound signal can be written as

$$s(t) = \sum_i a_i g(t - t_i, \theta_i) \tag{6.11}$$

where g is the time-domain representation of a grain, whose shape may be a function of further parameters θ. Typical grain durations are about 20–50 ms. Smooth amplitudes are used for the attack as well as the decay phase of the grains envelopes. Any of the described sound synthesis algorithms may be applied for the representation. For example, for the synthesis of vocal-tract sounds, subtractive synthesis (see below) would be adequate. The parameters θ are then filter frequencies and filter bandwidth. Granular Synthesis is a suitable technique for many sonification problems, since the sound can easily be computed, the model shows a high temporal resolution and allows flexible control over the time-variant spectral content. For further details see the tutorial on Granular Synthesis in [Roa85a].

6.2.6 Subtractive Synthesis

Subtractive Synthesis aims at shaping the spectral content of sound by filtering out undesired parts from a spectrally broadband input signal. A desired waveform is produced by applying time-variant filter to an input signal which is also called excitation source sound. This is quite complementary to additive synthesis, which builds up a complex sound by adding parts which are spectrally simple. Whereas additive models have difficulties in producing noisy sounds, noise can be easily introduced in Subtractive Synthesis by using noisy excitation signals. Whispering is an example of a noisy excitation of a resonator. In Subtractive Synthesis, the two main parts are the complex source sound and the filter or resonator system. Both components usually have time-varying parameters. Such distinction can be found in a lot of physical systems and so Subtractive Synthesis can be seen as a special case of Physical Models which will be discussed below.

A linear digital filter transforms an input signal $x(t)$ to an output $y(t)$. It can be characterized by a linear system

$$y[n] = \sum_{i=0}^{N} a_i x[n-i] + \sum_{i=1}^{M} b_i y[n-i]\ , \tag{6.12}$$

which is time-invariant if the parameters a_i, b_i do not change with time. Filters with $M = 0$ are called finite impulse response (FIR) filters because with ending input ($x[n] = 0 \ \forall \ n > n_0$) the output goes to 0 after a finite number of N samples. In case that a previous system output is fed back into the system, an infinite impulse response (IIR) filter is obtained. Such a filter may show oscillatory or even unstable behavior.

Alternatively, a linear system can be described by its transfer function [OW89] $H(z) = Y(z)/X(z)$, which is a complex valued function obtained by applying the z-transform to eq. (6.12). In general, $H(z)$ is given by a fraction of polynomials in z and can thus be characterized by its zeros and poles. The poles and zeros are directly linked to resonances and may be of particular use as parameters for sonifications. A detailed discussion of Subtractive Synthesis and linear systems is given by Moore [Moo90] and Oppenheim [OW89].

6.2.7 Physical Models for Sound Synthesis

The aim of Physical Modeling is to simulate the essential constituents of physical instruments in order to get similar means of sound control in models as exist in their real counterparts. Consider for example an acoustic guitar where the plucking position determines the resulting string sound. Implementing such "high-level" controls such as plucking position in other synthesis models would demand extensive parameter modifications. In contrast in a physical model of the string, the plucking position can be controlled directly.

The limited computation power makes it still necessary to reduce the complexity of the model for simulation. Three different alternatives for physical models will be presented in brief.

The first type of physical models is called *Spring-Mesh Models* [Smi97]. Here, acoustic systems are represented by a mesh or chain of point masses which are connected by elementary springs. Dissipation or other kinds of energy loss are modeled by introducing dashpot elements (mechanical resistors). The configuration of such a system is given by the position and velocity of all its masses. The modeling of a system is rather easy and intuitive but the computation, even of short sound pieces is costly, since the whole configuration must be updated at the audio sampling rate, e.g. 44100 times per second. The number of operations per sample is vast, even for tiny systems.

A more efficient alternative is offered by Modal Synthesis [Adr91], which analyzes the spectral properties of acoustic systems in terms of their modes of vibration (see Section 5.5). The normal modes, mode frequencies and the decay times are obtained from a mathematical analysis of the system. These can be used to determine the synthesis parameters for an additive model (see Section 6.2.1) dependent on the system excitation. Modal analysis thus provides a way to compute additive synthesis parameters from given physical parameters of a system under study. However, the more complex a system becomes, the more difficult is its mathematical treatment.

A rather new approach for Physical Modeling are Digital Waveguides (DWG) [Smi92, Smi98]. This technique is suitable to model traveling waves as sound waves in the air, and can be applied to model many physical system including the vocal tract, wind instruments or string instruments. The basic elements of Digital Waveguides are delay lines, linear filters and scattering junctions. Scattering junctions transmit and reflect parts of a wave. In addition, nonlinear interaction elements are used to model system components like the mouthpiece of a clarinet. These are modeled by nonlinear junctions [Smi98]. Waveguide sounds are efficiently computed while maintaining the ease of physically based controls.

6.3 Sound Transformations

This section is concerned with techniques to model and describe the signal modifications of a sound source (which might be computed by any of the synthesis techniques presented beforehand) until it reaches the listener's eardrums. This transformation includes effects due to reflections in the listening room, signal distortion and damping due to the medium or objects in between the sound source and the eardrum. The head and the shoulders for instance cause a direction depending damping of high frequency components. Such sound postprocessing is relevant for sonification as spatial effects may be applied to facilitate the separation of auditory streams. A reverberant sound simply is also more pleasant and natural than a dry sound.

The most frequent modifications performed on sound signals are normalization, mixing and spatialization. Sounds are mixed - within the limits of the medium's linearity - by simply superimposing their waveforms

$$s^{l,r}[n] = \sum_i s_i^{l,r}[n - n_i^0] \tag{6.13}$$

where n_i^0 is the onset of sound s_i, here written for stereo audio channels l, r.

Normalization modifies the amplitude of the source sound. This is done by

$$s[n] \leftarrow gs[n] \tag{6.14}$$

with a suited gain g so that $\max_n(|s[n]|)$ reaches a normalization limit s_{max}.

Spatialization aims at generating a stereo sound vector from a mono sound source so that the perceived location of the source is characterized by (d, θ, ϕ) at distance d from the center of the head, at an azimuth θ (θ=0, (resp.π) denotes in front (resp. right) of the listener), at an elevation of ϕ ($\phi = 0$ is the horizontal plane). Sound spatialization is a difficult task and still subject to research [Huo99]. The sound signals arriving at the ears differ in timing, level and spectral profile. The auditory system makes use of these clues to infer the location of the sound source. Interaural time difference (ITD) as well as interaural level difference (ILD) are easily computed by considering the traveling time and attenuation on propagation from the source to the ears. However, the signal distortion by reflections and shadowing effects are not as easily implemented. They can be integrated by convolving the source signal with angle-dependent filters called head-related transfer functions (HRTFs) [Huo99]. The resulting sound signals represent the sound within the outer ear channel, so that earphones have to be used to reproduce the spatial effect.

Given a stereo loudspeaker setup for sound reproduction, a sound source can be localized by intensity panning. The virtual source can be located within the triangle formed by the listener and the loudspeakers. Assuming an angle θ_s between the line of sight and a speaker, a direction (d, θ) is reached by using the gains

$$g_{l,r} = \frac{d_{l,r}}{d} \cos(\theta - \theta_{l,r}) \ , \tag{6.15}$$

for the left (resp. right) loudspeaker. For later usage, the function spatial$(s(t), \theta, d))$ is introduced, which transforms the signal $s(t)$ into a stereo sound vector so that the ITD and ILD are modeled as described above. Since both ILD and ITD do not depend on the elevation, this argument is dropped. Implementation of spatial() depends on the hardware setup. In the case of two given loudspeakers an intensity panning can be done as described above. If headphones are used, modeling the ILD and ITD without using any HRTF functions gives only a rough indication about source localization.

Apart from spatialization, the acoustic effects of the listening space need to be modeled in order to obtain a convincing sound. This includes all kinds of reverberations. Echos and reverb are determined by the individual properties of the room. At this point there is no space for a detailed discussion of how to implement these effects. To be brief, both effects can be implemented by using a chain of simple filters, comb filters and all-pass filters in parallel. Delay lines allow to add signal feedback for so-called early reflections caused by walls. For up-to-date reverberators, see Gardner [Gar98]. Although reverberation reduces the contrast of an audio signal, it enhances the listening quality, making sounds more pleasant and natural. As an additional aspect, reverberation parameters (decay time, delay, high-frequency damping) are suitable parameters for the representation of global attributes of data within sonification.

Many transformations have to remain unmentioned within this section as for example time stretching or pitch scaling [Lar98] which are suitable transformations for Audification, or sound morphing, which can enhance Parameterized Auditory Icons. For these topics, Roads [Roa96] and Kahrs [KB98] may be valuable references.

Chapter 7

Model-Based Sonification

We have seen in Section 3.7.4, that Parameter Mapping Sonification, the most common sonification technique, has several drawbacks. The framework of Model-Based Sonification (MBS) will be introduced in this chapter to offer a different perspective on sonification. MBS provides guidelines for the development of new sonification techniques called sonification models. Sonification Models evade some of the problems encountered in Parameter Mapping and they feature an intuitive user interface.

For motivation we will begin with an investigation of the listening experience. Musical listening and everyday listening will be contrasted and this will emphasize the relevance of event-related sound-properties in comparison to attributes related to the sound signal. Categories for listening contexts will be introduced and a hierarchical description of sonic events will be presented. The following analysis in Section 7.2 will stress the relevance of sound as a feedback for human actions. A prototypical interaction will be studied in which different aspects like latency, annoyance, redundancy and information are identified. Interaction sounds are not annoying if they communicate useful information for performing a task at hand. Annoyance of sonification and auditory display is a very important issue because humans have no ear-lids and thus can not ignore auditory stimuli completely. In Section 7.3, the framework of Model-Based Sonification will be introduced and guidelines will be given for the construction of sonification models. Finally, different interface techniques for the interaction with sonification models and issues of sound computation for sonification models will be discussed in Section 7.4.

7.1 Listening Experiences

Humans can experience sound in at least two different ways: if they attend to its rhythmical and harmonic organization and try to perceive relations within the sound like variations of acoustic attributes like pitch or brightness of the sound, they use *musical listening*. In this type of listening experience, properties of the sound itself are attended. Psychoacoustics obtains some understanding of musical listening, as pointed out in Section 4.2. However, when we hear a sound in our environment, we usually experience it in quite a different way. We try to identify what event or process has caused the sound, where the sound source of the sound is located relative to us and if the event may endanger us. This experience of listening is called *everyday listening*. Differences between these two types of listening experiences have been investigated from a psychological perspective by Gaver [Gav93a, Gav93b].

When asking people to tell what they hear, they frequently use a description of an imagined sound source or process and only rarely a characterization of the acoustic properties of the sound. For example, “the barking of a big dog” is more often answered than “a low-frequency tone at high level with large roughness whose amplitude decays rapidly after a quarter second”. Obviously, many sounds are directly represented internally in categories of their sources. Even if sounds are used that have not been experienced before, listeners are inclined to interprete them as a caricature of real-world events using a process-oriented description. From an evolutionary perspective it is plausible that auditory stimuli are characterized in terms of their sources: rapid detection of audio signals and the identification of the sources simply provided advantages concerning the reaction to dangerous threats (like approaching predators). The discipline of *ecological acoustics* is concerned with the description of acoustic properties that convey information about objects and events [Wri00].

Apart from the identification of sound sources, everyday listening is used to extract more detailed information about the objects involved like their material, stiffness, roughness and size. Listeners perceive attributes like the size, velocity and material of a rolling ball pretty accurately [Her00]. Such source attributes are continuous variables and their variation affect the sound signal in a very complicated manner. Nonetheless, they are extracted by most listeners without particular effort. I would propose to call this *Analytical Everyday Listening*. It differs from musical listening by the fact that the listener does not attend acoustic attributes, but continuous attributes related to source properties. In comparison to everyday listening, in analytical listening a higher degree of attention is directed to the event itself. In everyday listening the appropriate *reaction* to sound events is more relevant and this does not require the listener to concentrate on a sound in detail.

The aim of Model-Based Sonification is to address the skills of Analytical Everyday Listening in the domain of Exploratory Data Analysis. Auditory Display addressing everyday listening are not new. Gaver [Gav94] used auditory icons to convey information about events in a computer. Attributes of the interacting objects (e.g. an iconic representation of a file in a GUI) were mapped to attributes of an imagined object representing the auditory icon. For instance, when a file is selected with a mouse click a tapping sound is generated and the size of the tapped object is mapped by the file size. This technique is called Parameterized Auditory Icons. Model-Based Sonification, however, provides a means of experiencing data by interaction with data-driven objects. The linkage from data to sound is fully determined by the model and more complex than the mapping used in Parameterized Auditory Icons. While in Parameterized Auditory Icons, sound synthesis and presented data variables are separable, in Sonification Models such distinction can generally not be made.

To use everyday listening skills in a better way, it is helpful to understand how sounds are organized in our environment. Gaver proposed a hierarchical description of events [Gav93b]. The basic level are elementary interactions between solids, gasses and liquids. Hybrid sounds occur e.g. when solids and liquids interact. Each group has its own set of interaction types. Solids can for instance be struck, hit, scraped, rubbed or deformed. Liquids can drip, splash or pour and gasses can explode or wind. Everyday listening to elementary sounds allows us to identify the material type, interaction type and further continuous variables. By building patterns from elementary sound more complex sounds are obtained. *Temporal patterns* involve the superposition of a sequence of elementary sounds of the same type, e.g. a bursting glass. Gaver proposes *compound events* as the next level of complexity. A pattern of several elementary events is involved here. For instance, opening a soft drink can involves a solid sound from the can followed by a gas sound from the air that fizzles out. The decomposition of complex events into basic elements helps to describe the sound and facilitates its synthesis. But the reader should be aware that such a description as sound components is oriented on musical listening, on rhythmical patterns of the composition. This means that in such a view the process which causes the interrelation between the elements is neglected. From the everyday listening perspective it seems better to describe source properties of a compound system (instead of considering the temporal pattern in foot step sounds one would consider the speed of walking.).

Parameterized Auditory Icons try to mimic a temporal pattern of events. Model-Based Sonification in contrast describes the process. Temporal patterns in the sound may occur but they are not explicitly controlled.

Let us now consider the physical processes that causes sound in more detail. A qualitative description is used to facilitate the transfer to sonification models.

- Sound in real environment is always generated by a *physical system*. Physical processes are systems that evolve in time.
- Energy transfer from a physical process to the listener is done by propagating sound waves. These are caused by a physical contact of the system with a medium (usually air).
- Physical systems in an equilibrium state do not move and thus do not cause sound.
- Sound radiation and other dissipation reduces the energy of the system until it finally reaches a state of equilibrium.
- Excitation of a system usually increases the system's energy and starts or changes a physical process.
- Physical laws describe the dynamics of the system. The laws are invariant.

In this list, 5 elements can be discerned: the system (setup), the dynamics, the excitation, the radiation and the listener. The sound is closely related to the setup and dynamics of the sound generating systems. The configuration of a system determines the range of possible sounds whereas the dynamics (physical laws) never change. The dynamic evolution of a system connects complex properties of a system to the sound. In other words, system properties are indirectly or holistically encoded into the signal. But obviously, auditory perception uncovers such complex properties (like stiffness, size of an object) from the sound, although we still do not know how this functions exactly. In Model-Based Sonification, one intends to use this particular strength of auditory perception.

The basis for this is the humans capacity to learn. Humans are able to excite physical systems and interprete the acoustic output w.r.t. their excitation and to the object they interacted with. From such acoustic experiences and their relation to information obtained from other modalities, listeners learn the relation from material properties to acoustic properties. The same kind of learning is necessary with Sonification Models.

7.2 Sound and Interaction

Humans are embedded in a sound medium which transports sound waves, and they have various possibilities to manipulate their acoustic sensations actively.

- They can move their head and thus increase perceptual resolution (e.g. for localizing sound sources)
- They can interact with physical systems and thus cause or modify their sound environment.

- They can produce sound by themselves in various ways (speaking, breathing, clapping hand, finger flipping).

This section focuses on the second point, the interaction with acoustic systems. The focus is on interaction with solid objects as most of the objects that we can take in our hands are of this type. The energy transfer between two solids (e.g. a hand and an object) is mediated by a physical contact of both objects. The most frequent types of interaction are

- Plucking: deformation of a system by pulling a part against a restoring force. Termination of the contact sets the system free. Plucking a guitar string or bending a metal bar is an example. The interaction is short-lived.

- Hitting/striking: an impact force is transferred to the object which changes the velocity of some of its elements without affecting their position. Interaction time is in general short-lived. For example, a cup can be struck with a spoon.

- Rubbing: Rubbing is an interaction between two solids. If a hand rubs on a surface, the roughness of the surface causes a fast series of contact sounds, dependent on the rubbing speed and surface properties. The interaction is continuous.

- Scratching: a similar interaction as rubbing, but the contact force is higher which is reached by using a smaller interaction surface (e.g. a finger nail instead of the palm).

- Shaking: moving an object back and forth or rotating it back and forth in order to excite internal degrees of freedom of the object. For instance, shaking a rattle causes collisions between the contained grains. The interaction is continuous.

- Deformation: Applying a force may change the shape of an object irreversibly. For example squeezing a can of coke or to crumple up a piece of paper. Similar interactions are twisting or bending an object. These actions are continuous.

Consider the following interactions: (i) a person closes a door by pulling the door handle: the lock snaps in and causes a sound. (ii) a person tries unsuccessfully to put a key in a keyhole: failing trials cause a specific sound. The following observations can be made:

Immediate Response: sound corresponds to actions with short latency. Latencies are in the order of traveling times of sound waves from sound source to the ear, so 3 ms per meter.

Information: sound delivers helpful information for performing the task. The absense of sound would cause vagueness.

Annoyance: sounds that provide useful information for performing a task are not disturbing at all or only mildly disturbing. A pleasant sound level is not any louder than the level required to pick up the needed information.

Redundancy: sound may be redundant. At the same time the sound is heard one can see that the door is closed, haptic information is collected while positioning the key. Redundant information makes human actions more robust.

Signal: A sound can signal the completion of an action.

Subconscious Processing: The sound is usually not perceived consciously in such situations. It is part of an auditory background which contributes to facilitate human actions.

Control Loop: Actions cause an acoustic feedback which causes an update of the persons actions.

These examples show a set of aspects which have to be regarded when using sounds in human-computer interactions, especially in sonification. Often, a task may be performed by using visual tools and sound may be used to augment the display. In this case, a lot of care must be taken to make sure that such sounds are connected to human actions and that their level is low enough not to disturb the user. Small latencies are required if the sound is intended as a feedback to the user's actions. The information given by sound should not be independent of the information given in other media, because redundancies help to connect the sound with a process or action in the computer.

Some of the requirements are automatically fulfilled if actions are used as excitations of simulated objects. Model-Based Sonification uses human-computer interactions in this way as will be pointed out in the following section.

7.3 The Framework of Model-Based Sonification

Summarizing the results of the previous sections, auditory perception is able to extract meaning from sound signals which result from a temporal evolution of a physical system. The temporal evolution of a sounding system is determined by (i) the setup, (ii) the dynamics given by physical laws, (iii) the initial state of the system, (iv) the excitation which moves the system away from equilibrium. Humans are very skilled in extracting even subtle source properties from sounds which are produced by using this concept. In most cases, sounds which are produced actively are not annoying. Sounds are often not consciously interpreted and used in combination with other sensory input.

The aim of Model-Based Sonification is to translate these experiences into principles allowing to create sonifications from data. This is achieved by a Sonification Model. Analogous to sound generation in the real world, the model consists of a (a) a setup of dynamical elements, (b) a dynamics ("virtual physical laws") which describe the temporal evolution and (c) interaction interfaces. The dataset is used to parameterize the configuration of the model (also named "virtual acoustic object"). To facilitate further discussion, a simple example for a sonification model shall be given.

> **Example Model**: Assume a high-dimensional dataset is given. Perform 2d density estimation on the first two principal components of the data. Take the density function in the 2d plane as the local stiffness of a rectangular membrane under tension. The system user explores the model by striking-interaction. Striking (spatially resolved) the membrane puts kinetic energy on a surface element of the membrane. According to the given dynamics (e.g. wave equation), these excitation cause a dynamical motion of the membrane resulting in a sound. This sound is taken as the sonification and is presented to the user as a feedback of each of his excitations.

With such a model in mind, the qualitative differences between this and other sonification methods shall be contrasted now:

- In Parameter Mapping, the sound and its acoustic attributes are focused. In Model-Based Sonification, the source and its properties are focused.

- For the interpretation of Parameter Mapping Sonifications, a codebook or mapping table has to be referred to, and musical listening is needed to follow the evolution of isolated sound attributes. Interpretation of sounds from a Sonification Model only requires knowledge about the model.

- Sounds from Sonification Models do not have to make explicit use of sound attributes. The sound is less determined as in parameter mapping, since it also depends on the interactions with the model.

- Parameter Mapping Sonifications produce sounds without a natural analogue. There is no canonical way of interacting with them. They are more related to data-driven music compositions. Model-Based Sonifications build on a natural analogue. They try to meet our expectations on the sound-environment and explicitly include interaction.

7.3.1 Model Specification

This section provides more detailed and practical guidelines for constructing sonification models. The development process must address and answer the following 5 groups of questions:

Setup: What are dynamical elements of the system? What are their degrees of freedom?

Dynamics: What dynamics describes the evolution of the system in time?

Interaction: How can the user interact with the system?

Model-Sound Linking: What elements contribute to the audio signal? What elements transfer energy to the sound wave field?

Listener: If a “virtual listener” is part of the model, questions arise like: How do sound waves propagate to the virtual listener? How is the listening space structured? What is the location and orientation of the virtual listener?

Development and design of sonification models are creative processes which have to include knowledge about acoustics, sound computation and an imagination about what information is useful for a task at hand. Unfortunately it can not be known in advance if an imagined sonification model will offer sounds that are helpful or informative for the listener. A possible approach is to consider and optimize models by trial-and-error method. However, for simple models the out-coming sound may be anticipated. For further refinement, however, trial-and-error can not be avoided. A specific problem with this kind of development strategy is that the listener must become familiar with the sounds of a model. Using a standard collection of datasets which differ in those properties that shall become audible from the sonification model, may accelerate the development process.

Model Setup

In order to model a sound generating process, the first step is to define the physical elements of the sounding object. Many models, which are presented in Chapter 8 use a layout of point mass objects in an Euclidean vector space. The vector space dimension can be chosen arbitrarily. Three choices should be contrasted:

2d-models fit exactly to the dimension of the visual display. Representing data in form of a map (or a set of maps) enables the user to browse them visually, interacting with the map by using a computer mouse. Taking a model whose dimension matches that of the information displayed visually could increase the learnability of the display.

3d-models fit best to our acoustic real-world experiences as our environment possesses the same topological structure.

d**-dimensional models:** a suitable dimension is often that of the dataset itself. Embedding objects for each record of the dataset as a point into data space allows models without dimensionality reduction. Apart from that, the concept of data spaces is something scientists are familiar with so that this model can be understood easily.

Using one dynamical element for each data record is one possibility. An alternative is to use the dataset to setup some fixed elements and to introduce dynamical elements which probe the system. An example for a model using this strategy is the Particle Trajectory Sonification Model in Section 8.2. Another possibility is to use an intermediate layer: for instance, the dynamical elements could be a constant set of objects that are parameterized by a reduced representation of the data, e.g. by using vector quantization. In this case the data within the Voronoi cell of each prototype could be used to determine the acoustic properties of an object associated to a prototype.

Dynamics:

The dynamics describes the temporal evolution of the system as defined by the setup. For analogous (physical) systems the dynamics is given by the laws of physics which may be expressed in the form of differential equations. The state of a discrete system may be expressed by a point in phase space, given by the configuration and velocity in all degrees of freedom. The dynamics must allow to compute a new state $\mathbf{s}[n+1]$ of the system from the previous state $\mathbf{s}[n]$ and can for instance be given by a finite difference approximation of a differential equation.

In practical situations, a numerical simulation of a system is out of reach due to limited computation power. In this case an abstraction can be used. For instance a spherically expanding wave in model space is not computed each time step, but it is directly computed at what point in time the wave front reaches the virtual listener in model space. When such simplifications are performed, we have to take care that the algorithm tries to mimic the expected behavior as good as possible so that the acoustic output agrees with the expectations on the systems behavior. A good choice for dynamics is to take the same laws that describe real acoustic systems, since human listeners are already familiar with sounds of physical processes which are governed by such dynamics.

Interaction

The most frequent state of a sonification model is the equilibrium. In an equilibrated system the energy has reached a local minimum. Excitation drives the state away from equilibrium. This can be achieved in similar ways as exciting systems in the real world and which have been discussed in Section 7.2. Assume we have a model of the data-driven membrane introduced above. The implementation could be done by using a 2d-mesh of masses and springs where the spring stiffness is computed from the local density estimate. In this model, the dynamical elements are masses. Their degree of freedom is motion along the z-axis, which is orthogonal to the membrane surface.

The equilibrium state of this model is $z_i = 0 \;\; \forall \, i$. Exciting the membrane by striking mass i can be realized by setting the velocity of mass i to a positive value proportional to the hitting force. Plucking could be realized by setting mass i to a position $z > 0$ and update neighboring masses so that the membrane reaches a smooth shape. Rubbing would mean to update different masses in sequence depending on the rubbing trajectory.

Generally, an interaction is an algorithm for changing the system's configuration which optionally depends on the state of the system and additional parameters. The interface defines how parameters are specified using the input device. For instance some parameters may be driven by the position of the mouse pointer within a plot or from mouse clicks or graphical widgets on the screen. The closer the interface matches the user's expectations, the more easier it will be to intuitively relate the sound to the actions.

Listener

Sonification models can be grouped into two categories: (i) model spaces where the listener is "virtually" positioned and oriented, and (ii) location free models where the listeners position relative to the object has no influence on the sound signal. Models of type (i) address spatial listening and thus require at least a stereophonic sound system, even better a loudspeaker array around the listener. They may be characterized as macroscopic models, as the virtual acoustic system is large compared to the listener who is integrated into the model space. Models of type (ii) may be denoted as microscopic models. They generate mono sound sources expected to be located in one point relative to the listener. In the case of a computer workplace, there is a mismatch between the perceived source location (the loudspeaker) and the visualized source location (maybe an object on the screen) but users have only few difficulties in compensating the mismatch. Models of type (ii) are usually less expendable to compute. In Chapter 8, examples for both types of models will be presented.

7.3.2 Discussion

Model-Based Sonification offers an approach to sonification which differs from Parameter Mapping Sonifications. Therefore Model-Based Sonification is able to avoid some of the problems and disadvantages of Parameter Mapping. Important advantages of Model-Based Sonification are:

- Less Parameters. Whereas Parameter Mapping needs a complex mapping specification for high-dimensional datasets, sonification models work with a very small set of control parameters. In addition these (greatly reduced) parameters find an intuitive interpretation, since their role within the process is known. By this, they can be controlled easily.
- Persistent Structure. Whereas in Parameter Mapping the meaning of acoustic elements depends on the individual mapping, in sonification models the typical arrangement of acoustic elements remains invariant w.r.t. exchanging datasets. Thus sonification models are more likely to be learned and interpreted correctly by a user.
- Learnability. A sonification model will be used with a lot of different datasets. That allows the user to learn how to relate sound to structures in the dataset better than in Parameter Mapping Sonification where the mapping or an instrument is likely to be chosen different for every sonification.

- Flexibility. In Parameter Mapping Sonification, relations between data can not be explicitly used to determine the sound, since sound events are only driven by single records. In sonification models, such relations can easily be integrated into the model. A model can for instance use local information like density estimates to determine the acoustic features of a model component.

- Addressing Everyday Listening Skills. Perceptual interaction of acoustic attributes may cause that certain relations get inaudible in Parameter Mapping Sonifications. For example, if an event's duration is too short, attributes like modulations can not be further discerned. In Model-Based Sonification similar things may happen as well. But as sonification models do not address musical listening but everyday listening, this doesn't cause a problem. It is left to auditory perception to decide what parts of the sound are useful for discerning source properties and thus data properties. Thus, Sonification Models avoid a confrontation with this question.

- Intuitive Time Axis. In Parameter Mapping Sonifications, the meaning of time depends on the chosen mapping. In Model-Based Sonification, time matches to temporal evolution of the model and is thus intuitively related to changes or events within the process.

- Dimensionality Independence. In Parameter Mapping, due to a finite number of sound attributes, it is necessary to reduce the dimensionality in order to match the dimension with the number of acoustic attributes. Sonification models can be designed to work with arbitrary data dimensionality without dimensionality reduction. This can for instance be done by using the distances between data points to determine the behavior of system components.

- Task-Oriented Design. Sonification Models may be designed to provide information for a task at hand. For example they may be tuned in a way that sound depends on clustering, if the user is interested in data clustering.

- Intuitive Interface. Model-Based Sonification includes concepts for the manipulation and access of sonifications. Parameter Mapping Sonifications lack such controls and are rather perceived as a piece of music with few means of interaction.

- Interpretation. Knowledge of the model provides the key for interpreting sound or relations in the sound with respect to the data. For such conclusion Parameter Mapping needs the recourse to the mapping.

Some of the models presented in Chapter 8 finally apply synthesis techniques which are also used within Parameter Mapping Sonifications. Thus it may seem possible to interprete sonification models as a pre-stage for Parameter Mapping Sonification by transforming the dataset into a dataset of attribute vectors which then is used for sound synthesis. Using intermediating structures in this way is referred to as the concept of "virtual engines" in the sonification community [Kra94c], it is also called n-th order Parameter Mapping [Sca94]. In certain cases a sonification model can be implemented in such a manner. Nonetheless, Sonification Models exist which demonstrate that the class of sonification models is more extensive than what can be achieved by intermediating structures. The Particle Trajectories Sonification model presented in the Section 8.2 is a good example for this. One reason why sonification models may be realized by parameter mapping is the enormous computational complexity of simulating a physical process numerically. With the computational power currently available, it is necessary to simplify the model. Often such simplifications bring the algorithm into a form that allows to use familiar synthesis algorithms. One strategy to perform such a simplification is to separate subsystems which are set far apart so that they do not interact. In the Data Sonogram Sonification Model (see Section 8.1) all data points will be decoupled in this way. The more the dynamical elements are decoupled, the closer the rendering moves towards second order Parameter Mapping.

7.4 Sound Computation and Interaction for Sonification-Models

Sonification models aim at providing an intuitive interface for the user. This includes that the sound is computed in real-time and played with only small latency to the excitation. Unfortunately, real-time simulation is out of range for more complex sonification models involving large datasets and thus real-time sonification could not be considered in great detail in this thesis. First experiences with the real-time control of soundscapes by using hand postures as interface has been undertaken in cooperation with C. Nölker [HNR02] and the experiences are reported there.

The sonification models presented in this thesis therefore use short-lived excitations like plucking or striking. The sonifications are then computed off-line and the resulting sound vector is played back. During playback, the user has no means of interacting further with the system. However, as most sonifications last about one or two seconds the interaction loop is not interrupted for too long.

A graphical programming environment called Neo/NST [Rit00] was applied to implement the sonifications. The system offers modules for data visualization and database access and it has now been extended by modules for sonification. The user is shown a data visualization and he can use the mouse pointer to excite the sonification by clicking in the graphical display.

Compared to the expressiveness of human hands, this interface is very poor and currently ongoing research addresses more elaborate audio-haptic interfaces. This is the topic of J. Krauses diploma which the author has the pleasure to supervise [HKR02].

Chapter 8

Sonification Models

This chapter will present a number of sonification models and thus gives examples for the framework developed in Chapter 7. As pointed out before, development of sonification models is a creative process. The following perspectives might help finding some inspiration for a sonification model:

Analogous Processes: observe everyday listening situations which provide useful information to the listener. Consider the acoustic process that provides useful information in the real world and try to take an analogous process for the model.

Task-Oriented Observables: given a certain exploration task at hand (e.g. dimensionality estimation), focus on promising observables (e.g. the eigenvalue spectrum of local covariance matrices) and use them to drive object attributes (e.g. damping, stiffness of a spring) so that the sonification summarizes all the observables in acoustic form.

Listening Skills: observe human listening skills (e.g. listeners are very sensitive to detect rhythmical changes) and design the model as a process that generates sound that uses corresponding acoustic structures (e.g. rotating objects that yield to periodic, resp. rhythmical sound patterns.). Another example: human listeners are very sensitive to language sounds. Using a data-driven model of a vocal tract may potentially yield sounds which are for listeners easier to differentiate.

For the models presented in this chapter, these approaches go hand in hand. Without doubt, the developed models here are subjective choices made by the author in order to demonstrate various facets of Model-Based Sonification. In some cases during the model development process, the sonifications differ from the developer's expectations. In this case, the model can be modified to reinforce the perceptual effect. In other situations, a structure becomes audible without having expected it. In this case, the developer has the chance to learn something new from the data and detect new and unexpected structures in the data. My hope is that those sonification models which are suited best for certain EDA tasks will have better conditions for being used and that after an evolutionary phase a small set of standard sonification models will arise from the universe of models.

8.1 Data Sonograms

The Data Sonogram Sonification Model provides an acoustic display of a high-dimensional dataset in terms of a spatial representation of data [HR99]. As mentioned in Section 3.5, one function of auditory perception is providing clues for the localization of sound events in our environment. Data Sonograms are intended to exploit these listening capabilities. The model world consists of a vector space in which acoustic objects are positioned. Specifically the objects are damped oscillators and may be imagined as point masses attached to a spring. Excitation of the system causes the masses to vibrate. The location of the objects and the setup of their attributes (like spring stiffness, resistance constants, mass) are determined by the data and by local observables. The user interface consists of a 2d scatter plot of the data, in which the user can initiate a shock wave which expands spherically in data space. The shock wave transfers energy onto the objects and thus causes vibrations. The dynamic evolution of the model is used to compute the sonification.

Data Sonograms are a versatile concept that can be used for different tasks like cluster analysis, outlier detection, inspection of class borders, comparison of probability distributions or exploration of density variations. Depending on the task at hand, appropriate specializations of the model can be made concerning the setup of the objects from the dataset. Two possibilities for this will be introduced in Section 8.1.1.

8.1.1 Model Description

The Data Sonogram Sonification Model is defined for datasets given by a data matrix $\mathbf{X} \in M(N \times d, \mathbb{R})$ whose row vectors $(\mathbf{x}_i^{\mathrm{T}}, \mathbf{y}_i^{\mathrm{T}})$, $i = 1, \ldots, N$ are the data records, here divided into an input (or independent) part $\mathbf{x}_i \in \mathbb{R}^{d_{in}}$ and an output (or dependent) part $\mathbf{y}_i \in \mathbb{R}^{d_{out}}$. In unsupervised problems, $d_{out} = 0$ and in classification problems, $\mathbf{y}_i$ may contain the class label. In this model the input part is used to determine the location of objects in model space while the output part specifies acoustic properties of the objects.

Setup

Data Sonogram Sonification is based on a spatial model. The model space is an Euclidean vector space $\mathbb{R}^{d_{in}}$ in which objects O_i are located at $\mathbf{v}_i,\ \ i = 1, \ldots, N$. How the $\mathbf{v}_i$ are related to the data will be considered later. Assume that the objects may perform motion along one direction and additionally have d_{out} internal degrees of freedom (e.g. deformations). This choice is made to integrate the output part of the dataset to determine the sound. Then the objects state is determined by a $(1 + d_{out})$-dimensional vector in configuration space.

$$V_i(\boldsymbol{\xi}) = \sum_{j=0}^{d_{out}} \frac{1}{2} k_{ij} \xi_j^2 \tag{8.1}$$

as function of a point mass displacement vector $\boldsymbol{\xi}$ for object O_i. An object can thus perform independent motion along $(1 + d_{out})$ directions: one for each output dimension and an extra dimension for a data-driven observable as described below. V is a harmonic potential along each coordinate direction ξ_j.

The restoring force $-\nabla_{\xi_j} V_i = -k_{ij}\xi_j$ accelerates the mass m towards $\xi_j = 0$ with a force proportional to ξ_j and thus can be thought of as an ideal spring with stiffness k_{ij} along direction $\mathbf{e}_j$.

Dynamics

The point mass follows the equation of motion given by

$$m\ddot{\boldsymbol{\xi}}(t) = F(t) = -\nabla_{\boldsymbol{\xi}} V(\boldsymbol{\xi}(t)) - R\dot{\boldsymbol{\xi}}(t) \ , \tag{8.2}$$

which is a damped multidimensional oscillator as discussed in Section 5.3. R is called the resistance constant, m is the point mass. With V given by eq. (8.1), the system of differential equations decouples to $(1 + d_{out})$ independent differential equations

$$\ddot{\xi}_j + \frac{R}{m}\dot{\xi}_j + \omega_{0j}^2 \xi_j = 0 \ , \ \ j = 0, \ldots, d_{out} \tag{8.3}$$

with $\omega_{0j}^2 = k_j/m$. The general solution is given by

$$\xi_j(t) = A_j \exp\left((-\frac{R}{2m} + i\omega_{fj})t + \phi_{0j} \right) \ \text{with} \ \omega_{fj} = \omega_{0j}\sqrt{1 - \left(\frac{R}{2m\omega_{0j}}\right)^2} \ . \tag{8.4}$$

Along each coordinate axis of the oscillator, a damped harmonic motion with amplitude A_j, frequency ω_{fj} and initial phase ϕ_{0j} is performed. Amplitude and initial phase are given by the initial conditions of the system.

Initial State

Point masses of all objects O_i are initialized to $\boldsymbol{\xi}(0) = 0, \ \dot{\boldsymbol{\xi}}(0) = 0$. The objects are in a state of equilibrium and the point masses do not move.

Excitation and Interaction Types

From a selected point $\mathbf{v}_s$ in data space, an excitation wave is initiated. The wave front expands on spheres through model space with a constant velocity v_s. Exciting the system at time t_0, the object O_i is reached by the wave front at $t_0^i = (r/v_s) + t_0$ with $r = \|\mathbf{v}_i - \mathbf{v}_s\|$.

When the wave front of the excitation wave passes the coordinates of an object, an impact force proportional to the wave intensity $I(r)$ is given to the point mass. The impact force increases the kinetic energy of the object by $T = I(r)\sigma t_c$ where σ is the cross section and t_c the contact time.

The kinetic energy is distributed equally on all degrees of freedom by setting the velocity to $\dot{\xi}_j = v_0 \ \forall \ j$. Using the energy relation (see eq. 5.10) for the potential and kinetic energy along the jth component

$$E_j(t) = \frac{1}{2}m\dot{\xi}_j^{\,2} + \frac{1}{2}k_j\xi_j^2 \tag{8.5}$$

we derive the amplitude A_j of the solution in eq. (8.4) to be $A_j = v_0/\omega_0$.

There are several alternatives of how $I(r)$ may decay during wave expansion:

- Energy conservation: $I(r) \propto r^{-(d_{in}-1)}$ satisfies $I(r)A(r) = $ const., where $A(r)$ is the surface of the d_{in}-dimensional sphere. However, with high-dimensional settings, decay is too strong so that point masses at a larger distance are excited too weakly.
- Intensity decay in $\mathbb{R}^3$: $I(r) \propto r^{-2}$. We are familiar with this relation from our real-world experience.
- No fading: $I(r) = $ const.. Although this is unphysical, this distance law can be useful in certain tasks, e.g. for hearing outliers, which usually lie far away from other data points.

Model-Sound-Linking

For the sound signal $s_i(t)$ contributed from object O_i, the sum of all components of ξ_j is taken and thus the signal is given by

$$s_i(t) = \sum_{j=0}^{d_{out}} \xi_j(t) \; . \tag{8.6}$$

Listener

The model aims at surrounding the listener with object sounds. To achieve this, a “virtual listener”, as described in Section 7.3, is introduced into the model space and characterized by the head location $\mathbf{v}_l$ and its orientation. As a basic choice, the listener is located at the excitation wave center $\mathbf{v}_s$ with the ears aligned with the x-axis of the shown scatter plot. Object sounds propagate to the listener using a r^{-2} intensity decay where $r = \|\mathbf{v}_i - \mathbf{v}_l\|$.

Sound Synthesis

For efficient sound computation, modal synthesis is applied. This means, that instead of solving the differential equations numerically, the modes of the system are superimposed as described in Section 6.2.7. The source sound for an object is given by

$$s(t) = \sum_{j=0}^{d_{out}} A_j \sin(\omega_{fj} t) \exp\left(-t/\tau\right) \tag{8.7}$$

with decay time

$$\tau = \frac{2m}{R} \; . \tag{8.8}$$

The sonification signal is a stereo sound vector $s_{L/R}(t) = (s_L(t), s_R(t))$ computed by

$$\begin{aligned} s_{L/R}(t) &= \sum_{i=1}^{N} \text{spatial}(s_i(t), \phi_i, g(\|\mathbf{v}_i - \mathbf{v}_l\|)) \\ \phi_i &= \arctan(\|\mathbf{v}_{im} - \mathbf{v}_{lm}\| / \|\mathbf{v}_{in} - \mathbf{v}_{ln}\|) \\ g(x) &= \text{map}(x, [0, \max_i \|\mathbf{v}_i - \mathbf{v}_l\|], [0.3, 1]) \end{aligned}$$

Here, the azimuth angle ϕ_i for spatialization matches the direction which is visible in the 2d scatter plot of column (m, n) of the dataset. The mapping $g(\cdot)$ is used to avoid divergences of the sound level for objects that are too close to the virtual listener. The function spatial() computes an appropriate sound spatialization for the source sound with correct ITD and ILD as described in Section 6.3.

Data-Model Assignment

The connection between dataset and model parameters depends on the task. Here, two different assignments are presented. The first is suitable for cluster analysis, the comparison of cluster variance and outlier detection. The second assignment uses data-driven observables to demonstrate their use for creating task-oriented specializations. A summary of the available parameters of the model is listed in Table 8.1.

Parameter	Description
Objects O_i, $i = 0, \dots, N,\ \ j = 0, \dots, d_{out}$	
$\mathbf{v}_i$	coordinates vector
m_i	point mass
R_i	resistance constant
k_{ij}	stiffness constants
A_{ij}	mode amplitudes
Excitation wave	
$\mathbf{v}_s$	shock wave center
v_s	excitation wave velocity
I_0	initial intensity
Listener	
$\mathbf{v}_l$	listener location

Table 8.1: Parameters of the Data Sonogram Sonification Model.

Basic Data Sonogram Parameterization

This data-model assignment is the basic connection between data and model. The mass of all objects is set to a constant value, e.g. 1. The resistance constants are adjusted for all objects equally so that the object sound decays in about 200 ms by 60 dB. This is an acceptable compromise between long and short decay times, since for longer decay times the object sounds would overlap strongly and thus reduce the temporal resolution while for shorter decay times pitch perception would become more and more difficult. The stiffness constants $k_{ij}, j = 1, \dots, d_o$ from object O_i are a mapping from the output vector $\mathbf{y}_i$ of record i given by

$$k_{ij} = \text{map}(y_{ij}, [\min_i(y_{ij}), \max_i(y_{ij})], [1, 2])\ \forall\, 1 < j \leq d_{out}, 1 \leq i \leq N \tag{8.9}$$

and the stiffness constant k_{i0} is set to a constant value. Later, k_{i0} is driven from an appropriate local feature at $\mathbf{x}_i$. The shock wave velocity v_s is chosen as a matching constant so that the wave front passes the hyper-diagonal of the cube enclosing all data within a given time T. Alternatively, the average standard deviation over dimensions

$$\sigma = \sqrt{\frac{1}{d_{in}} \text{Tr}(\hat{\mathbf{C}}(\mathbf{X}_{N \times d_{in}}))} \tag{8.10}$$

is passed in time T. Suitable values for T are between 4 s and 15 s.

Feature-Driven Data Sonograms

This data-model assignment extends the previous assignment by using local features in data space to parameterize object properties. In the following the stiffness k_{i0} is determined by a data-driven feature.

For datasets containing an output part the following three features are suggested: properties of object O_i can be determined from

- a density estimate at $\mathbf{v}_i$.
- the average distance between the K nearest neighbors of O_i.

- intrinsic dimensionality at $\mathbf{v}_i$: e.g. compute the local principal components from the K nearest neighbors of O_i. The smallest subspace dimension including 80% of the variance may be used as a feature.

In classification problems where the output is a class label, the following specialization of Data Sonograms allows to perceive properties of class borders: compute a class member histogram from the K nearest neighbors of object O_i in model space. Then compute the local class mixing entropy S by

$$S = -\sum_{\alpha=1}^{M} p_\alpha \ln p_\alpha, \tag{8.11}$$

where M is the number of distinct classes and p_α is the probability for a member of class α. The probabilities can be estimated by the number of class members in class α among the K nearest neighbors divided by K. The class entropy is relatively low for data points surrounded by members of the same class and high in regions where different classes mix. Use S to drive the stiffness k_{i0}.

8.1.2 Implementation

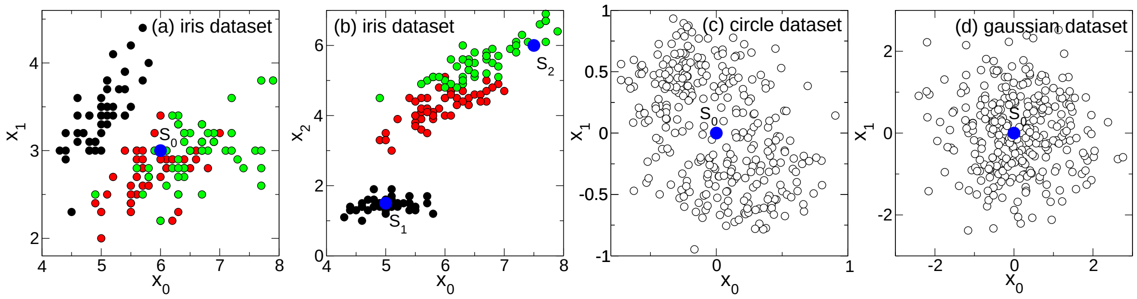
The user interface allows to specify the total duration, the sound level and the columns that are interpreted as the output part. A 2d scatter plot is shown and the user may select which components are visualized. In this scatter plot the sonification is started by a mouse click in the plot window. As pointed out before, modal synthesis is applied and the whole sonification is rendered off-line.

8.1.3 Examples

In this section, typical sounds of Data Sonograms are demonstrated for synthetic and real-world datasets. As a first example, data sonograms of the Iris dataset[1] are presented [Fis99]. The class label is the only dependent column and using the Feature-Driven Sonogram results in pitched object sounds with 3 different pitches for the 3 classes. The first sonogram (started at S_0 in plot (a) in Table 8.2), gives a first impression of the sound: each data point is presented by the high pitched marker sound and a lower pitched oscillator. Three pitches can easily be discerned. Sound examples are collected in Table 8.2. At the beginning of the sonifications, a noisy sound is played to mark the beginning of the sonogram. This is particularly useful in high-dimensional situations, where most points are at large distances of any data point, since here the delay between excitation and first return indicates how empty the space is. In the Iris examples, all low-pitched tones are grouped in the first sound example. However, by browsing the scatter plot and starting sonograms from other points, locations are easily found where the Iris plant groups are also separated in the sound. Excitation at S_1 allows the conclusion that no mixture between classes 1 and 3 occurs. Exciting the data at S_2, the gap between class 1 and the others is clearly perceived, indicating separability.

In the next example, the 10d noisy circle dataset is used with $\sigma^2 = 0.2$ (see page 9). In most 2d projections, the circle structure can not be seen. The data sonogram of this dataset started at the dataset mean shows that no data is found close to the mean. It sounds like all data points

[1] see page 10



File/Track 1:	Iris dataset: started in plot (a) at S_0 , (b) at S_1 , (b) at S_2
Track 2	10d circle dataset: started in plot (c) at S_0 (mean) (c) at S_1 (edge)
Track 3	10d Gaussian: plot (d) started from S_0
Track 4	Three clusters: Example 1
Track 4	Three clusters: invisible columns used as output variables Example 2
Description:	Data Sonogram Sound examples for synthetic datasets and the Iris dataset
Duration:	≈ 5 s

Table 8.2: Sound examples for Data Sonograms.

are sharply distributed at a distance d. For comparison, listen to the sonogram of a 10d Gaussian cluster. The scatter plot of this dataset looks almost the same but the sound is quite different. As a third sound example, the circle dataset is excited from the edge of the circle. You will notice that the sonification lasts about twice as long and sound contributions start much earlier after excitation.

The next examples show a dataset drawn from a mixture of three Gaussians. The clusters get audible as sonic bursts. Even if no output variables are given, it is sometimes practical to take some variables that are not visualized as dependent components. These variables then contribute to the timbre of the object's sound. This can be perceived in the last sound example where the different pitches within the clusters indicate the variation of the output columns in the clusters.

8.1.4 Conclusion

Data Sonograms provide a new way to perceive data by using a spatial scanning of the dataset. Since arbitrary local observables can be used to control the physical attributes of the objects, the model can be adapted to specific analysis tasks.

One advantage of the Data Sonogram Sonification Model presented here, is that it can be applied to arbitrary datasets without modifications and with the need to control and adjust only few parameters. The physical parameters and their influence on the sound of an object are easily understood and thus the interpretation of the sound w.r.t. the data becomes intuitive.

8.2 Particle Trajectories Sonification Model

The Particle Trajectories Sonification Model provides a means to render auditory presentation for high-dimensional datasets by probing a data-driven potential function with test particles. The motivation for this model was to create an auditory representation which provides information about clustering of multivariate data without relying on any other cluster analysis carried out beforehand. Instead, the aim was to let "the data speak for itself". In addition, a model was looked for, that is able to generate “rich sounds” in the sense of sound complexity without needing the direct control of limited sound synthesis algorithms. The latter objective was motivated by the fact that our listening sense is well suited to discern even subtle clues from complex sound.

The Particle Trajectory Sonification Model works as follows: a potential function is constructed from the given high-dimensional dataset by superimposing data point potential functions which are shifted to the data point coordinates. Particles with a given initial kinetic energy are injected into model space. They move around according to a given dynamics. The sonification is computed from the superposition of the kinetic particle energy as a function of time. While single particles fail to provide clustering information about the whole dataset, a bunch of test particles is able to summarize such clustering properties. In the sonification features emerge making the whole sound more than the sum of its parts.

There are different ways of using the model: (i) interactive probing allows the user to locate the starting point of the particles in a 2d scatter plot of the data. This provides information about the local properties of a region of interest. (ii) global sonification distributes the initial particle positions over the whole distribution and thus gives a summary over the distribution. (iii) multi-scale sonification computes a sequence of either global or local sonifications reducing a resolution parameter by each step. As a result the patterns change throughout the sequence so that resolution dependent structures become audible.

Particle Trajectory Sonifications are computed by numerically solving the equations of motion. This is necessary since the potential function is complicated and the system can not be divided into unconnected subsystems. Therefore, sonification rendering is extremely computational expensive so that real-time application is currently out of range. Nonetheless from the conceptual point of view the PTSM is an interesting model as to be shown in the following.

8.2.1 Model Description

The model may be used for any dataset given by a data matrix $\mathbf{X} \in M(N \times d, \mathbb{R})$ whose row vectors $\mathbf{x}_i^{\mathrm{T}},\ i = 1, \ldots, N$ are interpreted as point coordinates in a d-dimensional Euclidean model space. No additional information about the records such as class labels is used within the model.

Setup

Particle Trajectory Sonification is based on a spatial model. The model space is an Euclidean vector space $\mathbb{R}^d$ in which point masses of mass m are fixed for each data point at coordinates $\mathbf{x}_i$. The same mass is used for all data points. The masses are fixed and do not involve any degrees of freedom. Their only purpose is to contribute to a gravitational potential in data space given by

$$V(\mathbf{x}) = \sum_{i=1}^{N} \phi(\mathbf{x} - \mathbf{x}_i)\ . \tag{8.12}$$

$\phi(\mathbf{x})$ is the potential energy of a point mass m_p in the field of a point mass m in model space. Different from the gravitational law, where $\phi_{gr}(\mathbf{x}) = -Gm_pm/\|\mathbf{x}\|$ here the potential

$$\phi(\mathbf{x}) = -m_p m(\sigma) \exp\left(-\frac{\|\mathbf{x}\|^2}{2\sigma^2}\right)\ . \tag{8.13}$$

is used, which approximates a harmonic potential close to the origin and which converges to 0 for large distances. This has be chosen for two reasons:

- Divergent potential functions like the gravitational potential cause numerical instabilities during computation.
- The potential in eq. (8.13) is approximately quadratic at the origin. This leads to oscillatory motion of particles in ϕ that show only a single frequency and thus results in simple pitched tones in the sonification.

An important control parameter in $\phi(\mathbf{x})$ is the interaction length σ, which determines the range and the resolution of the potential: for σ^2 being much larger than the average variance over dimensions, V has only one global minimum close to the dataset mean which in model space is the center of mass (see Figure 8.1,a). Then the dataset is represented with a too coarse resolution (also denoted as underfitting or oversmoothing). In the other extreme, for σ smaller than the average distance between nearest neighbors of the dataset, V has a local minimum for each point mass (see Figure 8.1,d). That can be regarded as looking at the data with a too fine resolution (also denoted as overfitting). More interesting are values of σ between these extremes where clusters in the dataset are represented by local minima of V. An alternative potential is $\phi_{sh}(\mathbf{x}) = -m_p m(\sigma)\sigma/\sqrt{(\sigma^2 + \|\mathbf{x}\|^2)}$ which is a shielded gravitational law so that it has the same curvature as $\phi(\mathbf{x})$ at $\mathbf{x} = 0$. Both functions are shown in Figure 8.2.

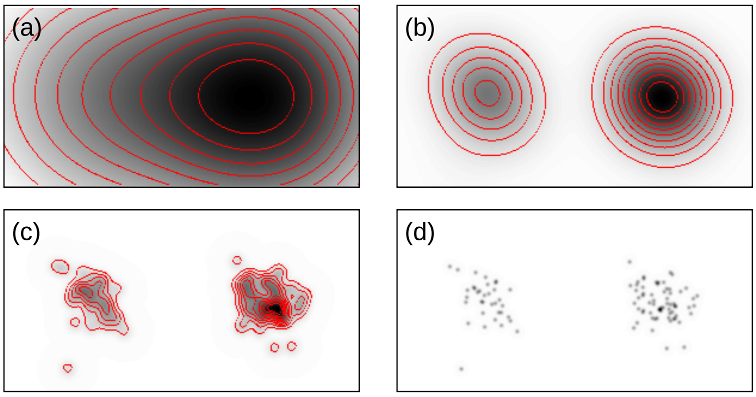


Figure 8.1: Density plots of the data potential V for a dataset drawn from a mixture of two Gaussians with different a priori probabilities. The interaction length σ is chosen depending on σ_{ds}, the average standard deviation over dimensions for the dataset. The plots show iso-density lines for $\sigma = \sigma_{ds} \cdot 10^v$ with (a) $v = 0$, (b) $v = -0.5$, (c) $v = -1.2$ and (d) $v = -1.8$.

The construction of V can be related to kernel density estimation given by

$$p(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N} K\left(\frac{\mathbf{x}-\mathbf{x}_i}{\sigma}\right) \tag{8.14}$$

with a kernel function $K(\mathbf{x})$ normalized to fulfill $\int K(\mathbf{x})d\mathbf{x} = 1$ [Sco92]. Compared to kernel density estimation with a Gaussian kernel, V shows minima instead of maxima due to the sign and does not fulfill the normalization.

Dynamical elements of the model are particles that are injected into model space. They are entirely described by their mass m_p, coordinate vector $\mathbf{x}^\alpha$ and velocity vector $\mathbf{v}^\alpha = \dot{\mathbf{x}}^\alpha$ with α being the particle index.

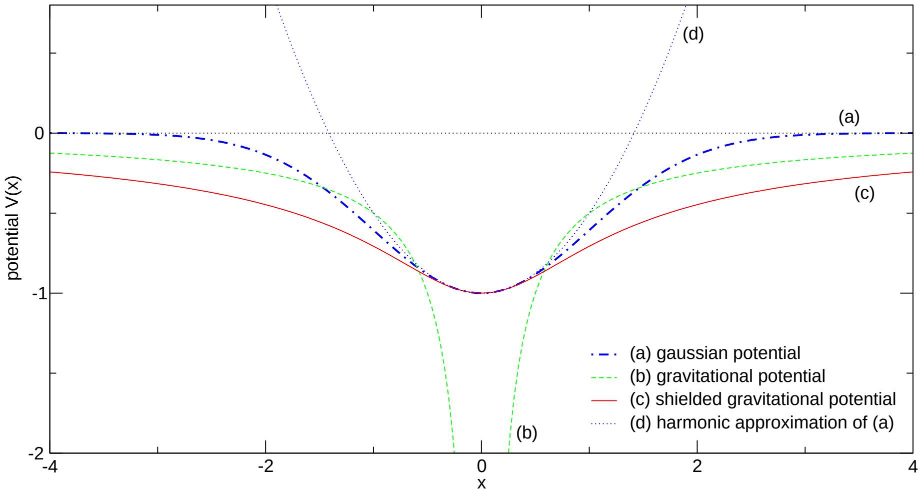


Figure 8.2: The data point potential in the one-dimensional case: (a) negative Gaussian potential with $\sigma = 1$, (b) gravitational potential (c) shielded gravitational potential $\phi_{sh}(\mathbf{x})$ and (d) quadratic approximation of (a).

Dynamics

M particles are injected into model space to probe the potential V. The test particles follow the equation of motion given by

$$m_p \ddot{\mathbf{x}}(t) = -\nabla_{\mathbf{x}} V(\mathbf{x}(t)) - R\mathbf{v}(t) \;\text{ with }\; \mathbf{v}(t) = \dot{\mathbf{x}}(t)\;, \tag{8.15}$$

which is Newton's law including a damping term as introduced in Section 5.3. R is the resistance constant, m_p the particle mass. $\mathbf{x}(t)$ and $\mathbf{v}(t)$ are computed by numerically solving the equation of motion at discrete time steps $t_n = n\,\Delta t$. The sampling period $\Delta t = 1/\nu_{SR}$ is given by the reciprocal sampling frequency used for sound representation. For a given configuration $(\mathbf{x}_n^\alpha, \mathbf{v}_n^\alpha)$ of a particle α at time step n, the configuration at time step $(n+1)$ is approximately given by

$$\mathbf{v}_{n+1}^\alpha = \mathbf{v}_n^\alpha + (-\nabla_{\mathbf{x}} V(\mathbf{x}_n^\alpha) - R\mathbf{v}_n^\alpha)\,\frac{\Delta t}{m_p} = g\mathbf{v}_n^\alpha - \frac{\nabla_{\mathbf{x}} V(\mathbf{x}_n^\alpha)}{m_p}\,\Delta t \tag{8.16}$$

$$\mathbf{x}_{n+1}^\alpha = \mathbf{x}_n^\alpha + \mathbf{v}_n^\alpha\,\Delta t\;, \tag{8.17}$$

where $g = 1 - \frac{R}{m_p}\Delta t$ is the gain per step which determines how fast the particles converge to a local minimum. The number M of particles to be injected is specified by the user. Assume that σ is chosen so that the N_c clusters in a dataset are separated by potential walls of V, then M should be large enough (e.g. $M \gg N_c$) to ensure that particles converge to all clusters. The more particles are taken, the more “stable” the sonification is on repetitions. However, as computation time scales with M, a trade-off between quality and computation time has to be found.

Initial State

The test particles with index $\alpha = 1, \ldots, M$ are positioned at randomly selected positions $\mathbf{x}_0^\alpha$ in model space. The velocity vector is initialized to a random direction so that the total particle energy

$$E_0^\alpha = V(\mathbf{x}_0^\alpha) + \frac{1}{2} m_p \|\mathbf{v}_0^\alpha\|^2 \tag{8.18}$$

is negative and the particles remain close to the dataset and are not able to escape. A simple method to achieve this is to set $\mathbf{v}_0^\alpha = v_0 \hat{\mathbf{v}}$ with a random unit vector $\hat{\mathbf{v}}$ and a velocity

$$v_0 = \sqrt{-0.05 \frac{V(\mathbf{x}_0^\alpha)}{m_p}}\ . \tag{8.19}$$

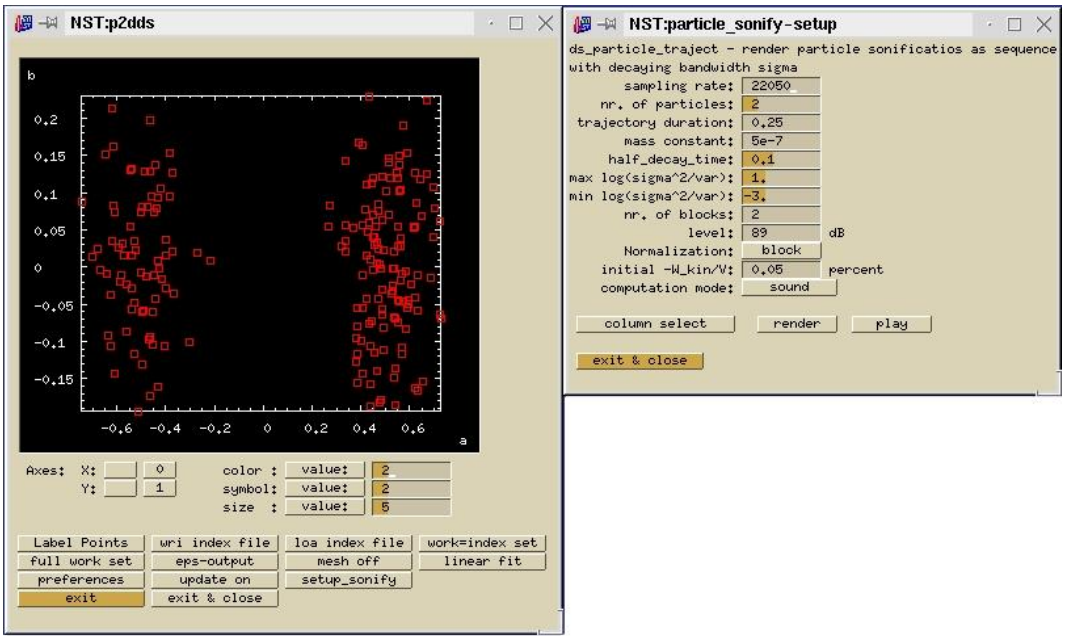
This means that the particle's energy exceeds the binding energy by 5%. Selection of the initial coordinates $\mathbf{x}_0^\alpha$ depend on the interaction types as explained below.

Excitation and Interaction Types

The system is excited by setting a set of particles into an initial state as described above. Figure 8.3 shows the GUI for invoking Particle Trajectory Sonifications. Three alternatives have been considered for this:

- **Interactive probing:** in a scatter plot visualization the user can select a position by clicking the mouse. The nearest neighbor to the mouse pointer in the plot is searched and particles are set to its coordinates. Since the initial velocities of the particles differ, they will move on different trajectories.

- **Global sonification:** the particles are set on data point coordinates of a random subset of the data.



Axes: X: 0
Y: 1
color : value: 2
symbol: value: 2
size : value: 5

ds_particle_traject - render particle sonificatios as sequence with decaying bandwidth sigma
sampling rate: 22050
nr. of particles: 2
trajectory duration: 0.25
mass constant: 5e-7
half_decay_time: 0.1
max log(sigma^2/var): 1.
min log(sigma^2/var): -3.
nr. of blocks: 2
level: 89 dB
Normalization: block
initial -W_kin/V: 0.05 percent
computation mode: sound

Figure 8.3: Screenshot of the Graphical User Interface for Particle Trajectory Sonification. Particles are injected into model space by clicking the mouse in the scatter plot or by pressing the "render" button.

Model-Sound Linking

The sound vector is computed by superimposing the contributed sound signal $s^{\alpha}[n]$ of each particle α which is given from the temporal evolution of the particles kinetic energy so that

$$s[n] = \sum_{\alpha=1}^{M} s^{\alpha}[n] = \sum_{\alpha=1}^{M} \frac{1}{2} m_p \|\mathbf{v}_n^{\alpha}\|^2 \tag{8.20}$$

As an alternative, the energy difference between consecutive values can be taken:

$$s_2[n] = \frac{1}{2}(s[n] - s[n-1]) \tag{8.21}$$

This is equivalent to applying a high-pass filter to the signal $s[n]$ in eq. (8.20) given by its z-transform $H(z) = (1-z^{-1})/2$ and a frequency response $g(f) = |H(e^{i2\pi f/\nu_{SR}})| = \sin(\pi f/\nu_{SR})$ and thus any DC-Bias is removed [OW89]. The sound signal is normalized to an appropriate sound level prior to playing it.

Listener

The sound signal is not spatialized w.r.t. the listener. The sonification is a mono sound signal. Therefore sound propagation from the particles to the listener does not need to be addressed.

Data-Model Assignment

The particle trajectories are controlled by 6 parameters which are summarized in Table 8.3.

Parameter	Description
m	data point mass
R	resistance constant
σ	interaction length
m_p	particle mass
ν_{SR}	sampling rate
W_{kin}	initial kinetic energy

Table 8.3: Parameters of the Particle Trajectory Sonification Model.

σ and W_{kin} control the qualitative behavior of the sonification. The role of σ as a resolution parameter has already been discussed in the model setup. Assume a dataset is given with only one record. The sonification should be independent of σ since in this case there is no preferred interaction length. To achieve this behavior, the curvature[2] of ϕ must be independent of σ at $\mathbf{x} = 0$. This can be accomplished by using a scaling of m according to $m(\sigma) = \hat{m}\sigma^2$. This mass scaling forces the shape of $\phi(\mathbf{x})$ to remain independent of σ. The sampling rate determines the time step $\Delta t = 1/\nu_{SR}$ and the mass $\hat{m}$ is adjusted to get a suitable oscillation frequency of particles in $\phi(\mathbf{x})$. In a one-dimensional case, the frequency is given by

$$f = \frac{1}{2\pi}\sqrt{\frac{\phi''(0)}{m_p}} = \frac{1}{2\pi}\sqrt{\hat{m}}\,, \tag{8.22}$$

[2] more precisely the Hess(ϕ)

and thus independent of m_p. The resistance constant R determines how fast the particle energy decays and thus is a parameter for choosing the decay time which is related to R by $\tau_h \approx \ln(2)\Delta t\, m_p/R$. In the user interface it is more convenient to specify τ_h.

The sonification of a single particle provides only limited information about the clustering of the dataset. Assume a dataset is given where the data points group in 3 Gaussians clusters. Assume further, that the number of data points in the three clusters is N_1, N_2 and N_3. With a suitable σ where V has 3 separated potential troughs, the sound of a particle is determined by the trough into which the particle converges. Assume this is the potential trough of cluster 1, then the sound is determined by the curvature at the minimum which is approximately $\phi''(0) \approx N_1 m_p \hat{m} \boldsymbol{I}_d$. The particle will perform an oscillatory motion in this trough of V with frequency $f \approx \frac{1}{2\pi}\sqrt{N_1 \hat{m}}$. This implies that from the pitch only clusters can be distinguished by sound if the cluster masses differ. It is furthermore possible that one cluster contributes with particle oscillations at different frequencies. This would be the case if the cluster had an elliptical shape being equivalent to different eigenvalues in the cluster sample covariance matrix.

Figure 8.4 shows a particle trajectory in a potential $V(\mathbf{x}) = \phi(\mathbf{x})$. It depicts how friction causes energy loss and that the particle converges to the origin. As expected, the frequency of the sound converges asymptotically. The particle sound can be found in sound example PTSM-Ex-1 (Track 5).

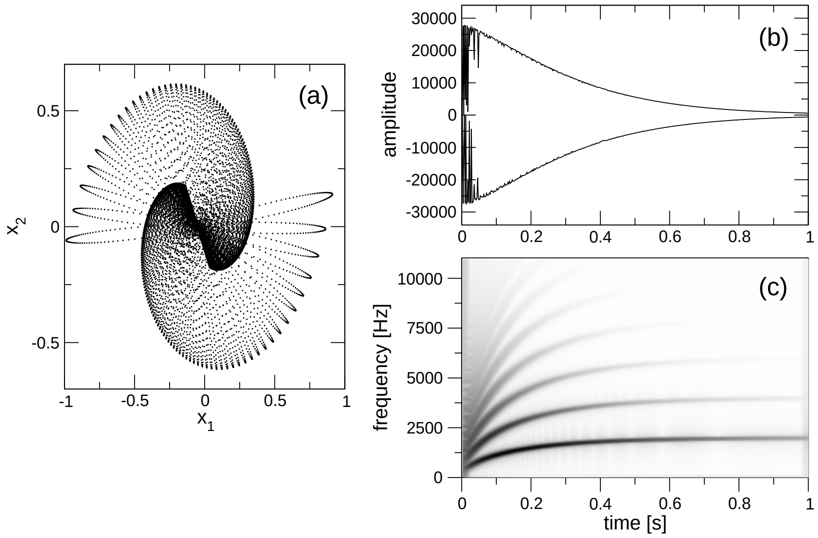


Figure 8.4: Particle Trajectory in a 2d model space for $V(\mathbf{x}) = \phi(\mathbf{x})$. (a) shows 11000 steps of the trajectory, (b) high-pass filtered kinetic energy, (c) spectrogram of the sound signal. The sound for this trajectory is provided as PTSM-Ex-1.

Figure 8.5 shows the spectrogram for a sequence of 10 particles moving in a dataset with two clusters in $\mathbb{R}^2$. The particle is started in each step with an initial energy $W_{kin} = -0.5V(\mathbf{x}_0)$. Thus it is able to move over the barriers between different potential trough in the beginning, before its energy gets so small that it converges to the local minimum of V. The pitch of the particle sound identifies the trough. The relative number of steps with identical pitch corresponds to the a priori probability of the cluster. The sound example is PTSM-Ex-2 (Track 6).

Figure 8.6 shows the spectrogram for the sonification of 25 particles injected simultaneously into model space for the same dataset as in the example given above. Now both tones overlap and

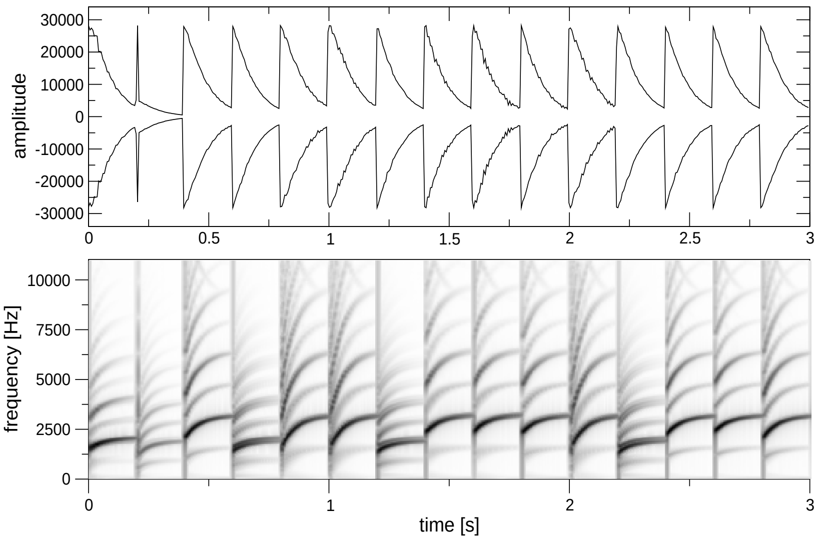


Figure 8.5: Sequence of 15 Particle Trajectory Sonifications in a dataset with 100 records sampled from a mixture of 2 Gaussians with different prior. The interaction length was adjusted to $\sigma^2 = 0.1\mathrm{Tr}(\hat{\mathbf{C}}(\mathbf{X}))/d$. Particles converging to different modes differ in their pitch. Sound Example: PTSM-Ex-2.

the loudness of a tone corresponds to the number of particles that converge to one of the clusters. The various harmonics of both tones are shown in the spectrogram. However, when listening to the sound example PTSM-Ex-3 (Track 7), it becomes difficult to discern two tones. Since they share a common onset and have a similar spectral evolution, they are grouped into one auditory stream. But if we compare the sound to a sonification where all particles converge to one of the clusters (as in sound example PTSM-Ex-4) (Track 7) we can clearly perceive a difference.

The choice of σ is of particular importance for the sounds of this model. In the sonification shown so far, σ had to be selected by the user. Several values of σ are of potential interest and it remains unclear which ones until the respective sonifications are heard. Therefore the following sonification presents a set of Particle Trajectory Sonifications in a sequence, decreasing σ by each step by multiplying it with a factor $g_s < 1$. This sequential sonification is called multi-scale sonification as several resolutions are probed. The qualitative behavior of the multi-scale sonification for a clustered d-dimensional dataset $\mathbf{X}$ with N records, is:

- for large values of σ compared to $\sqrt{\mathrm{Tr}(\hat{\mathbf{C}}(\mathbf{X}))/d}$, the potential becomes $V(\mathbf{x}) \approx N\phi(\bar{\mathbf{x}})$ and thus a scaled version of ϕ centered at the dataset mean. The sound of particle trajectories will resemble the sounds of a particle in ϕ with the only difference that the frequency will be higher.

- With decreasing σ, $V(\mathbf{x})$ changes until at some σ data clusters form separated potential troughs. Particles converge to the corresponding centers and contribute pitched sounds with a frequency characteristic for the curvature at the mode of V.

- Finally σ becomes so small that the potential has separated minima close to each data point. Again all sounds resemble the sound of a particle in ϕ.

- The frequency decreases during the sequence as fewer and fewer data points contribute to

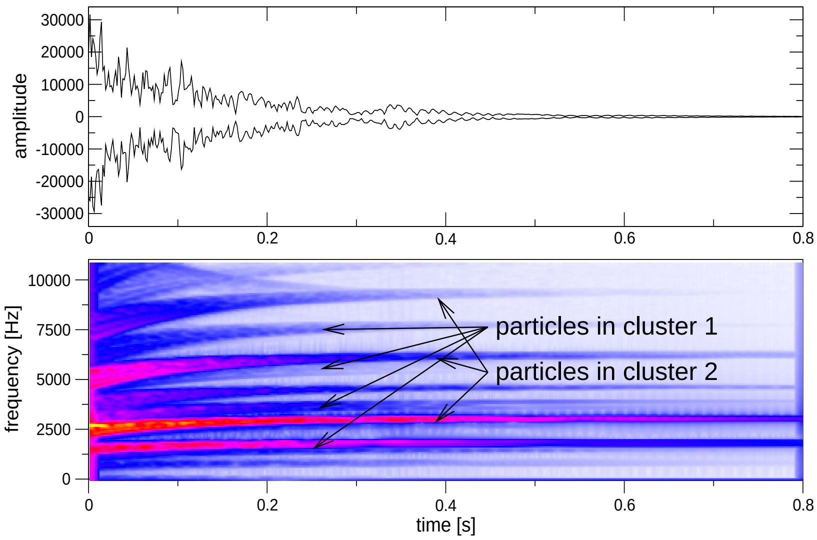


Figure 8.6: Sonification of 25 particles for a dataset with two Gaussian clusters, as described in the caption of Figure 8.5. Particles that converge to different clusters contribute with tones of different pitch and timbre.

minima of V. Roughly it will decrease from $\sqrt{N} f_0$ to f_0 with $f_0 = \sqrt{\hat{m}}$.

Figure 8.7 shows the spectrogram of a multi-scale sonification for the dataset with two clusters that as been used earlier. For the example, $N_s = 10$ σ-steps are computed using 25 particles for each step with a constant value of $\sigma_k,\ k = 1, \ldots, N_s$ given by

$$\sigma_k = \sigma_{k-1} \cdot g_s = \sigma_0 g_s^k \text{ with } \log(g_s) = \frac{1}{N_s} \log \frac{\sigma_{N_s}}{\sigma_0} \ . \tag{8.23}$$

For the example, $\sigma_0^2 = 10\sigma_{ds}$ and $\sigma_{N_s} = 0.001\sigma_{ds}$ was used with $\sigma_{ds} = \sqrt{\mathrm{Tr}(\hat{\mathbf{C}}(\mathbf{X}))/d}$.

8.2.2 Examples

In this section a number of sound examples will be presented for synthetic datasets to demonstrate the typical sounds of this model.

Three Gaussians

The first example is a multi-scale sonification for data sampled from a mixture of three Gaussians with different a priori probabilities in $\mathbb{R}^2$. A sequence length of 30 σ-steps with 50 particles per step is used and 0.25 s sound are computed per step at a sampling rate of 16 kHz. Figure 8.8 shows a spectrogram of the sound signal in plot (a). The sound example is PTSM-Ex-5 (Track 8). For comparison listen to the second example, where the data is drawn from a normal distribution (sound example PTSM-Ex-6) (Track 9). The coarse signal evolution agrees with the remarks made in the last section.

- A: all data point potentials merge in the limit of large interaction length.
- B: chaotic trajectories passing between potential trough are perceived as noisy sounds.

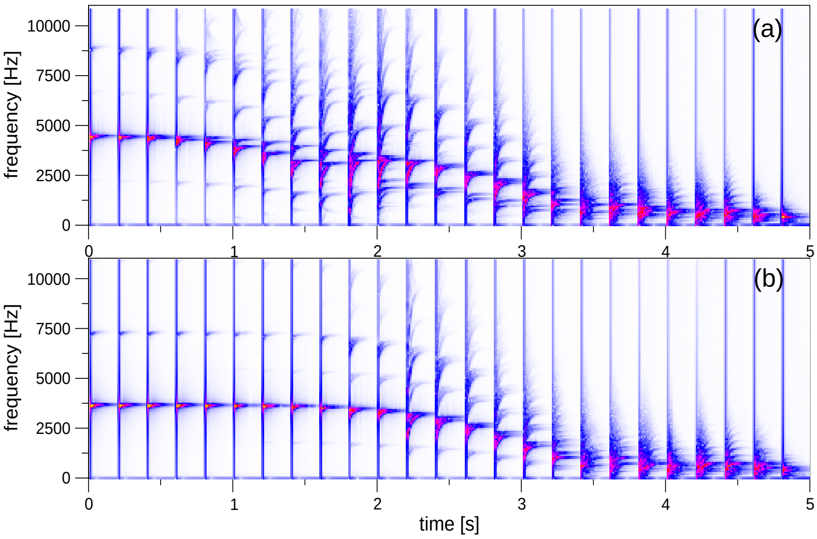


Figure 8.7: PTS σ-sequence sonification using 25 particles, 10 steps in a data potential for a dataset with two clusters. (a) shows the spectrogram for PTSM-Ex-3, (b) for PTSM-Ex-4.

- C: stable plateau where 3 separated troughs are given for the 3 clusters.
- D: members of the clusters begin to separate. Again this contributes noisy sounds.
- E: all data points are separated by potential walls.

Most obvious and helpful for detection of clustering from the sound is the plateau (C). In contrast, a sharp drop in pitch like in the second example indicates that there is no particular clustering structure in the data.

The next examples sonify some 4-dimensional datasets. The first is the Iris dataset. Here the sonification indicates that there might be a weak clustering of the data. Sound Examples: PTSM-Iris-1 (Track 10) with 20 particles per step, PTSM-Iris-2 (Track 10) with 3 particles per step. The spectrogram is shown in Figure 8.9.

The last dataset is the 4d tetrahedron cluster dataset with 5 clusters (see p. 10). In this sound a strong pitch plateau is audible, although it contains only one pitch. This indicates that the dataset contains small compact clusters (as the plateau begins early) which have spherical shape (only one pitch occurs) and in addition all clusters have the same size (the pitch plateau is left by all particles at the same time). Sound Example: PTSM-Tetra1 (Track 11) with 10 particles per step.

8.2.3 Conclusion

The Particle Trajectory Sonification Model renders a sonification by numerically integrating the equation of motion for a particle in a complicated data-driven potential. The sound therefore is complex in that it shows features which emerge from the complicated motion of particles in V. The multi-scale sonification allows to conclude from the existence of a pitch plateau to a clustering of data. The number of different pitches corresponds approximately to the number, resp. shape of the clusters. Obviously these features are also seen in a visualization of the spectrogram. However, the sounds provide a qualitative new experience to the user by using a different medium.

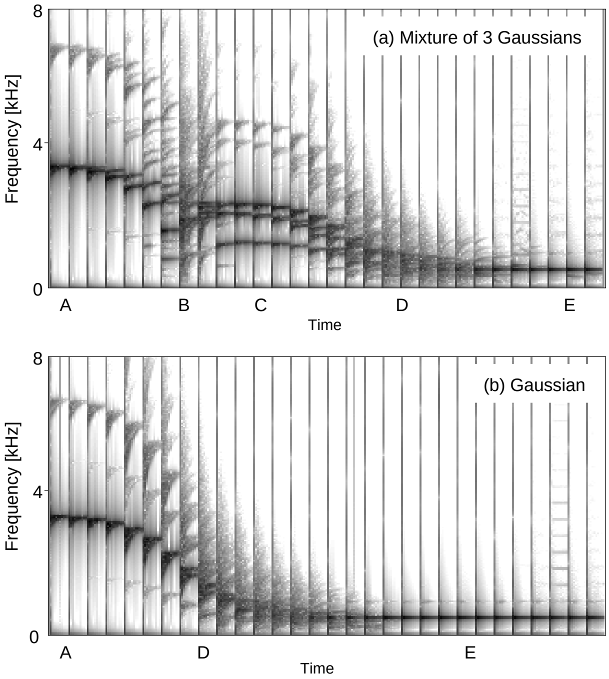


Figure 8.8: Spectrogram of the Particle Trajectory Sonification for the (a) mixture of 3 Gaussians and (b) a Gaussian distribution, as described in the text.

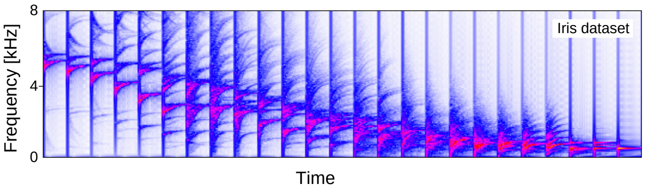


Figure 8.9: Spectrogram of the Particle Trajectory Sonification for the Iris dataset. The sonification is computed for the dataset without class label. The sound provides weak evidence for a clustering in the data.

In contrast to Parameter Mapping Sonification, the model has few parameters whose meaning is easily understood.

The main disadvantage is the computational complexity. Computation of the examples of 7 s duration took about 100 s on a 800 MHz Pentium machine. This model shows clearer than the other models in this thesis that Model-Based Sonification is a richer concept than Parameter Mapping Sonification. In Parameter Mapping each sound contribution is determined by a single data point, here all data points potentially contribute to determine one particle trajectory in a complex manner. One particle trajectory thus encodes collective properties of the whole dataset in a sound which may be very rich in terms of its complexity. Whereas the Data Sonogram Model may be implemented as a data preprocessing that generates appropriate attribute vectors for a Parameter Mapping Sonification, such a model simplification is not possible in PTSM.

Following this conceptual type of model therefore seems promising for the development of sonifications.

8.3 Sonification of McMC Simulations

The McMC Simulation Sonification Model [HHR01] allows to perceive the structure of a high-dimensional density function p by listening to an ongoing exploratory process in the domain of p. The model was developed in cooperation with M. Hansen[3] during the author's research stay at the Bell Labs.

The motivation for this model was the fact that humans routinely use their auditory senses to monitor processes in their environment. Listeners usually habituate to a stationary sound process and then are able to distinguish even subtle changes in the sound patterns. To exploit such listening capabilities for exploring high-dimensional densities or datasets, a process is defined in model space that continuously generates sound events which represent the sonification. Specifically, this model applies a Markov chain Monte Carlo (McMC) process [GRS96] to explore features of a probability density under investigation. The McMC simulation process runs a chain of states in the domain of a density p which has p as its stationary distribution. Therefore the chain visits all modes of p if it is only run long enough. The connection of McMC simulations with the Particle Trajectory Sonification Model from Section 8.2 was the basis for this model and extends the latter model to exploration of high-dimensional densities.

McMC simulation is a popular computational tool for making inferences from complex, high-dimensional probability densities p. It uses a Markov chain which is setup so that it has p as its stationary distribution. However, to be successful, the chain needs to be run long enough so that the distribution of the current draw is close to the target density. Unfortunately, very few diagnostic tools exist to monitor characteristics of the chain. Here, sonification contributes as a new tool to observe convergence properties of a McMC chain.

The McMC sonification consists of three auditory streams which provide information about the behavior of the Markov chain. Information about the McMC moves, their acceptance or rejection and specifically the nearest mode to the sampler in p are used to determine acoustic elements. The model is defined to explore a density function. However, it can also be applied to explore high-dimensional datasets. In this case a density p can be obtained by kernel density estimation [Sil86], see also eq. (8.14).

8.3.1 McMC Simulation

Traditional Monte Carlo techniques use an independent sample of data drawn from a target density p to estimate various features of p. Especially in high-dimensional settings, p might be very complex so that direct sampling from p becomes very inefficient. Here, McMC methods provide a tool for drawing samples more efficiently so that the data is well suited for Monte Carlo integration. The idea behind McMC is to generate a sequence $\{\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \dots\}$ which has the target density p as its stationary distribution. For the model presented, the Metropolis algorithm [GCSR95] is used. In the literature on statistics, most applications of McMC are associated with so-called Bayesian models. In this case, the variable $\mathbf{x}$ is a vector of parameters in a probability model and p is a *posterior distribution* for $\mathbf{x}$. The characteristics of p relate directly to the uncertainty present in the components of $\mathbf{x}$.

To implement the Metropolis algorithm, first a suitable *jumping distribution* $J(\mathbf{x}_b|\mathbf{x}_a)$ is identified, where $\mathbf{x}_a, \mathbf{x}_b \in \mathbb{R}^d$. It requires J to be symmetric, so that $J(\mathbf{x}_b|\mathbf{x}_a) = J(\mathbf{x}_a|\mathbf{x}_b)$ for all values of $\mathbf{x}_a$ and $\mathbf{x}_b$. A further requirement for J is irreducibility, that means that any state $\mathbf{x}$

[3] `http://cm.bell-labs.com/who/cocteau/index.html`

can be reached from any other state in a finite number of steps with a probability greater than 0. See [Tie94] for further requirements on J.

To move the chain from $\mathbf{x}_{t-1}$ to $\mathbf{x}_t$ a point $\mathbf{x}^*$ from the distribution $J(\mathbf{x}^*|\mathbf{x}_{t-1})$ is drawn. The *acceptance ratio* is then computed by

$$r = \frac{p(\mathbf{x}^*)}{p(\mathbf{x}_{t-1})} \; . \tag{8.24}$$

Finally, with probability $\min(r, 1)$ the proposition is accepted and thus $\mathbf{x}_t = \mathbf{x}^*$ is set; otherwise $\mathbf{x}_t = \mathbf{x}_{t-1}$ is taken. As initial state $\mathbf{x}_0$, a random point for which $p(\mathbf{x}_0) > 0$ is chosen.

Under this simple scheme, it is not difficult to show that the distribution of the samples $\{\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \dots\}$ converges to p [GCSR95]. The qualitative properties of this Markov chain depend on J. For example, suppose that J is multivariate normal so that

$$J(\mathbf{x}_a|\mathbf{x}_b) = \left(\frac{1}{2\pi\sigma^2}\right)^{d/2} \exp\left(-\frac{\|\mathbf{x}_a - \mathbf{x}_b\|^2}{2\sigma^2}\right) \; . \tag{8.25}$$

If σ^2 is small compared to the average variance of p per dimension, the probability that the chain moves between different modes of p is small; hence the chain remains near the same mode for a long time. On the other hand, if σ^2 is very large, the acceptance ratio for each proposed move tends to be small and the chain rarely leaves its current position. Therefore, while convergence is guaranteed at least theoretically for many choices of J, the jumping distribution has considerable influence on the finite-sample properties of the chain. Figure 8.10 shows the output from three runs of the Metropolis algorithm for a 2d density p. The Metropolis algorithm is perhaps the most simple technique for quickly generating a Markov chain with the correct stationary distribution. Many other schemes exist in the statistics literature that extend this approach to much more elaborate modeling contexts.

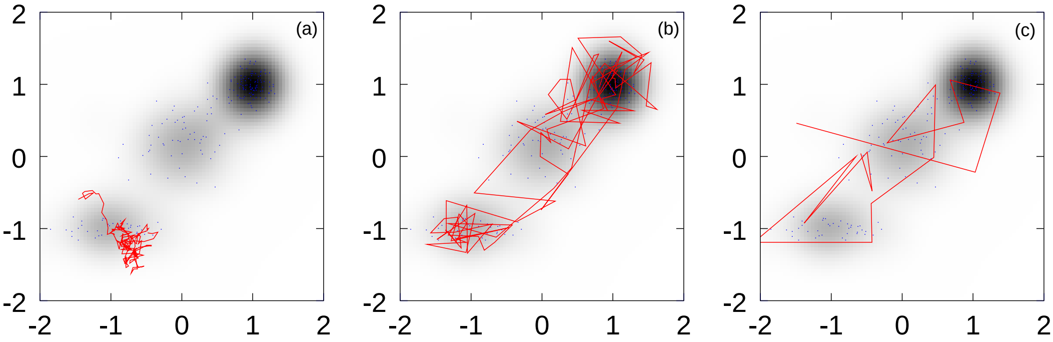


Figure 8.10: McMC random walk in a 2d density estimate for a dataset drawn from a mixture of 3 Gaussians. Grey values represent probability density, data points are plotted with blue symbols. 200 McMC steps are connected by a red line. The jumping distribution is $J(\mathbf{x}_a|\mathbf{x}_b) = g(\mathbf{x}_a; \mathbf{x}_b, \mathbf{\Sigma})$ with $\mathbf{\Sigma} = 2s^2\boldsymbol{I}_2\mathrm{Tr}(\mathbf{C}(p))$ and s^2 set to (a) 0.1 (the chain does not mix well) (b) 0.8 , (c) 4 (only few accepted moves).

In general, the samples $\{\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \dots\}$ are used to estimate properties of p like its mean and variance. Of particular interest is the mode structure of p as it is an important component in making inferences about a statistical model under study. This brief introduction to McMC suffices to motivate the McMC sonification model. Ongoing work in the collaboration with M. Hansen aims at extending McMC sonification to more complex McMC schemes.

8.3.2 Model Description

The McMC Sonification Model is defined to operate on arbitrary differentiable density functions p as they occur for instance in Bayesian models for the posterior distribution of model parameters. However, using non-parametric density estimation [Sil86], the model can also be applied to the analysis of high-dimensional datasets (see also eq. 8.14 on page 89).

Setup

McMC sonification, like the PTS Model, is a spatial model. The setup is completely determined by the density $p(\mathbf{x})$. p is unchanged during the sonification, and the only dynamical element is the state of the Markov chain given by its vector $\mathbf{x}_n$ at step n. For probing the density p, a test particle is introduced at position $\mathbf{v}(0)$.

Dynamics

The McMC simulation is sonified by computing a deterministic process on each accepted McMC step i. This is done by "injecting" a particle at $\mathbf{v}(0) = \mathbf{x}_i$ whose motion is determined by local features of p so that the sound is determined by information about p.

The particle dynamics is similar to the dynamics used in Particle Trajectory Sonification. Here, however, the potential $V(\mathbf{x})$ is replaced by the density $p(\mathbf{x})$ and in addition the dynamics is changed so that particles are accelerated towards the modes (local maxima) of p. In summary, particles follow a path in the domain of p according to the dynamics

$$m\ddot{\mathbf{v}}(t) = \nabla_{\mathbf{x}} p(\mathbf{v}(t)) - \gamma\dot{\mathbf{v}}(t) \ , \tag{8.26}$$

where m is the particle mass and γ a resistance constant. We recover the setup in eq. (8.12) by taking p to be a kernel density estimate based on $\mathbf{X}$ using a spherical Gaussian kernel with bandwidth σ.

The McMC simulation provides samples in $\mathbb{R}^d$ which are used as starting points for the deterministic process which explores the local environment and represents mode properties by sound. The following choices have been considered for the deterministic step, say i:

Method 1: Set one particle to $\mathbf{v}(0) = \mathbf{x}_i$ with $\dot{\mathbf{v}}(0) = 0$ and choose γ so small that $\mathbf{v}(t)$ performs many oscillations until it converges to the mode. Then use $s_i[t] = \|\dot{\mathbf{v}}(t)\|$ as a sound signal that is contributed for McMC step i.

Method 2: Introduce one particle as in Method 1, but now choose γ so that the mode is reached with no or only very few oscillations. Use the sequence $\{p(\mathbf{v}(0)), p(\mathbf{v}(1)), \dots\}$ to parameterize complex auditory grains (see Section 6.2.5). For instance by determining the frequency evolution of the auditory grain by the obtained density sequence.

Method 3: Run a mode search algorithm starting at $\mathbf{v}_i$ in order to find the nearest local maximum and then use parameterized auditory grains to present information about the mode and the mode search process. A mapping for this will be designed in the next section.

The acoustic output for Method 1 is identical to that of the PTS Model described in Section 8.2 and shares the same advantages and disadvantages. The main advantage is its conceptual simplicity, the major disadvantage is the computational complexity which makes Method 1 unsuitable for real-time monitoring. Because of this the other methods were considered as alternatives to provide similar information by sound, but requiring less effort.

Methods 2 and 3 are a computationally simpler link between the McMC process and the sound. However, these approaches require additional parameters to be set. Method 2 reduces the computation effort by shortening the number of steps until the particle reaches the mode. This scheme requires less than 10% of the time used by Method 1 for each McMC step, because the high "friction" quickly slows the particles to a stop. As each particle sweeps out its trajectory, the shape of the neighborhood close to the nearest local maximum can be perceived as a pitch variation pattern. Modes can be identified by the characteristic pitch which corresponds to the value of p. Method 3 reduces the required computation time further by applying mode search algorithms and therefore is the most efficient sonification.

Initial State and Excitation

The McMC simulation is started at any random position $\mathbf{x}_0$ for which $p(\mathbf{x}_0) > 0$ is given. Excitation in this model means simply triggering the start of the simulation. An interactive operation (using the GUI) is possible if a high-dimensional dataset is explored. Then the starting point can be selected by using the mouse pointer in a scatter plot of the dataset.

Model-Sound Linking

The sonification is computed by superimposing sound contributions of each accepted McMC step. The sound event for a McMC step depends on the method selected. For Method 1 and 2, the linking to sound has already been introduced above for one deterministic step. The sonification is composed then from

$$s[n] = \sum_{i=0} s_i[n - \lceil T \nu_{SR} i \rceil] \tag{8.27}$$

where ν_{SR} is the sampling rate and T denotes the time between the onsets of successive McMC steps and $s_i[k] = 0 \, \forall \, k < 0$. For Method 3, a model-sound linking is discussed in Section 8.3.3.

Listener

The listener is not localized with respect to the spatial configuration of the model. The sonification is rendered as a single sound source and aspects like sound spatialization, propagation or attenuation need not to be addressed.

8.3.3 Sonification Design

This section is concerned with the design of the McMC sonification applying Method 3. The information available at each step i of the McMC process can be divided into three groups: (i) local data, which is valid for only a single step, (ii) global data about the McMC process which is updated on each step and (iii) mode-specific data which is updated and stored in a mode database for each McMC step. The mode database provides a summary of current knowledge about the mode structure of p. Table 8.4 summarizes the available data in this three categories. Some McMC steps are illustrated in Figure 8.11.

Local Data	
$\mathbf{x}_i$	coordinates at step i
$p(\mathbf{x}_i)$	density at step i
$\nabla p(\mathbf{x}_i)$	gradient at McMC step i
d_i	distance to last McMC step
r_i	acceptance ratio
$\mathbf{m}_i$	coordinates of nearest mode found by mode search
m_i	index of the nearest mode in mode database
$p(\mathbf{m}_i)$	mode density at nearest mode m_i
d_i^m	distance between McMC step and nearest mode
A_i	boolean acceptance flag
Global Data	
c_i^a, c_i^r	counter for accepted and rejected McMC steps
N_m	number of modes in mode database
p_{max}	$= \max\limits_j p(\mathbf{m}_j)$
Mode Database — for all modes $j \leq N_m$	
$p(\mathbf{m}_j)$	mode center $\mathbf{m}_j$ and density at mode
c^j_{ma}, c^j_{mr}	counter for accepted/rejected McMC steps
δ_j	average distance of all attracted steps from mode center
$\mathbf{v}^j_m$	mean of all steps whose det. step converged to m_j
$\mathbf{S}_j$	sample covariance matrix of all attracted steps

Table 8.4: Available variables that provide information on the McMC process.

The sonification uses three auditory streams which can be switched on or off independently. The first stream contains auditory grains using Granular Synthesis[4] to present the Markov chain random walk through data space. The second stream provides information about rejected propositions of the Markov process and informs the listener further about the distance of the McMC step from the mode center. The third stream contains Auditory Information Buckets introduced below, which summarize the information collected in the mode database.

Auditory Information Buckets (AIB)

Auditory Information Buckets (AIB) are a technique for controlling the level of detail and the rate of auditory information events. We can imagine an AIB to be a container where we can put information into. A counter for the number of items and the data values themselves are stored. In addition a threshold (resp. a flushing condition) is defined that limits the AIB size. If the AIB counter exceeds the threshold, a flushing of the AIB is triggered, resulting in the synthesis of a sonification for an AIB sound event which may summarize the content in an auditory event. Thus the rate and complexity of bucket sonifications can be intuitively controlled by adjusting the threshold. The higher the threshold, the more complex the auditory event may be, but accordingly a flushing event will occur less frequently. AIBs thus provide a kind of auditory zoom, allowing to inspect data at a user controlled resolution.

For McMC sonification, AIBs are used to summarize the characteristics of the modes. Each

[4] see Section 6.2.5

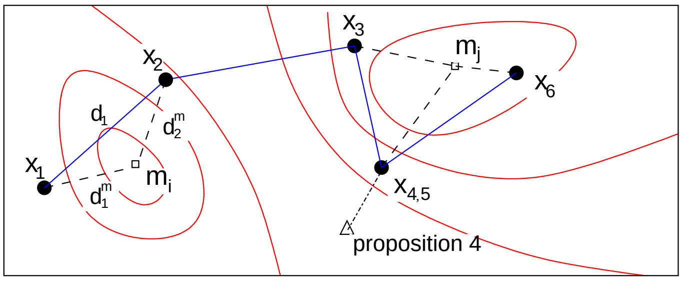


Figure 8.11: Illustration of 6 McMC steps in a bimodal distribution. The deterministic step converges to the modes as indicated by the dashed line. At step 4, the proposition is rejected, so that the chain remains at $\mathbf{x}_5 = \mathbf{x}_4$.

McMC step whose deterministic step converges to a specific mode contributes to the mode AIB which collects the McMC position $\mathbf{x}_i$. On a flushing of the AIB for mode j, the local sample covariance matrix $\mathbf{S}_j$ of the collected sample is computed and a sound is rendered that uses the eigenvalues of $\mathbf{S}_j$ as described below. The AIB sounds will be designed so that it is possible to draw conclusions on the shape of the attraction basin of a mode from the sound.

Nonlinear Pitch Mapping

An important property of auditory grains is their frequency. In the following design, the pitch of auditory grains correlates to the density value of the nearest mode. However, when this is done, two modes with very similar densities cannot be distinguished acoustically. Assume, that M modes $j = 1, \ldots, M$ have been discovered whose mode densities are $\{\pi_1, \ldots, \pi_M\}$. The problem is solved by a nonlinear mapping function $z = g(\pi)$ which maps density values π to pitch values z in such a way that a higher portion of the available pitch range $[z_{min}, z_{max}]$ is allocated in density regions that contain many values.

A resolution requirement is given by the density in π. Thus a nonlinear pitch mapping is constructed by

$$g(\pi) = \text{map}(c_\tau(\pi), [0, 1], [z_{min}, z_{max}]) \tag{8.28}$$

where

$$c_\tau(\pi) = \frac{1}{M} \sum_{j=1}^{M} \Phi\left(\frac{(\pi - \pi_j)^2}{\tau^2}\right) , \tag{8.29}$$

is the cumulative distribution function (CDF) derived from a kernel density estimate of the mode distribution using a Gaussian kernel with bandwidth τ. In this expression, $\Phi(\cdot)$ is the standard normal CDF. A good choice is $\tau^2 = 0.2\hat{\mathbf{C}}(\{\pi_1, \ldots, \pi_M\})$. Figure 8.12 shows $g(\pi)$ for a sample of 10 given density values. As a special case for $\tau = 0$, the empirical (step function) CDF of the sample $\{\pi_1, \ldots, \pi_M\}$ is used. As g is only evaluated at π_j, it suffices to maintain an ordered list of modes and mapping the index in the list to the pitch. Figure 8.12, (b) shows the corresponding pitch values for this approach. Nonlinear pitch mapping amplifies pitch differences while maintaining the order.

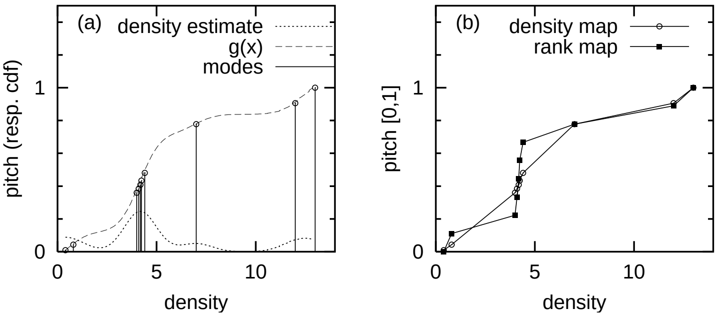


Figure 8.12: Nonlinear pitch mapping for a given sample of 10 modes. (a) shows the kernel density estimate, the CDF, the modes and their assigned pitch $g(\pi)$. Obviously, the nonlinear mapping increases the resolution at densities around the value 4. (b) shows the pitch assignment for $\tau = 0$, connected by straight lines.

McMC Process Monitoring Stream

The basic element of the McMC sonification is the process monitoring stream which provides transient information about the running McMC process. For every accepted McMC step, one auditory grain is added to this stream. As the McMC process is a serial evolution in time, the step index i has the natural meaning of a time index and "process time" is computed by $t(i) = T \cdot i$, where T is a scaling factor specified by the user. Depending on the analysis task, different scales are useful for inspection of the McMC process. With $T \approx 0.1$ s/step, individual steps can be resolved, whereas $T \approx 0.002$ s/step gives an overall impression about the modes of p and their relation to each other. The alignment of auditory grains with equally spaced points in time leads either to the perception of a monotonous rhythmical pattern or pitch at frequency $1/T$. This unwanted side-effect can be avoided by adding a random time jitter of $T/4$ to the onset.

The auditory grain signals are synthesized by multiplying periodic waveforms with an amplitude envelope with smooth attack and decay. Synthesis parameters are: fundamental frequency, duration, amplitude and timbre. Acoustic properties of the sounds in the PTS Model inspired the choice of the mapping from variables to sound attributes, so that the sound may be understood as the result of an imagined process.

Duration: In the PTS Model the duration depends on the density difference $p(\mathbf{m}_i) - p(\mathbf{x}_i)$. For the auditory grains this is realized for instance by the mapping

$$\text{duration} = \text{map}(p(\mathbf{m}_i) - p(\mathbf{x}_i), [0, p_{max}], [0.2T, 3T]) \ . \tag{8.30}$$

Fundamental Frequency: In the PTS Model, the frequency converges to a value that is characteristic for a mode. This is maintained by mapping mode density to pitch. However, with this mapping it is difficult to distinguish modes of similar density. Therefore nonlinear pitch mapping is applied and the mapping is

$$\log_2(\text{frequency}) = \text{map}(c_\tau(p(\mathbf{m}_i)), [0,1], [8,10]) \tag{8.31}$$

with $c(\cdot)$ given in eq. (8.29). Since $c(\cdot)$ is a monotonous function, density relations map to pitch relations.

Amplitude: In the PTS Model sound level is related to density difference $p(\mathbf{m}_i) - p(\mathbf{x}_i)$. Here, amplitude is used in a different but also intuitive way: the amplitude is used to communicate the “interestingness” of the mode by using loud grains for modes which are rarely visited. This is achieved by mapping $S = c_i^{m_i}/i$ to the amplitude by

$$10 \log_{10}(\text{amplitude}) = \text{const} + \text{map}(S^{-1}, [0, N_m], [0, 30]) \tag{8.32}$$

New modes are thus introduced with very loud grains and the amplitude is getting softer while the McMC sampler moves around in the attraction basin of a mode. Missing level variations indicate coarsely if the McMC simulation has converged to equilibrium. Figure 8.13 illustrates this stabilization process. The sonification is found in sound example MCMC-Ex-1 (Track 12).

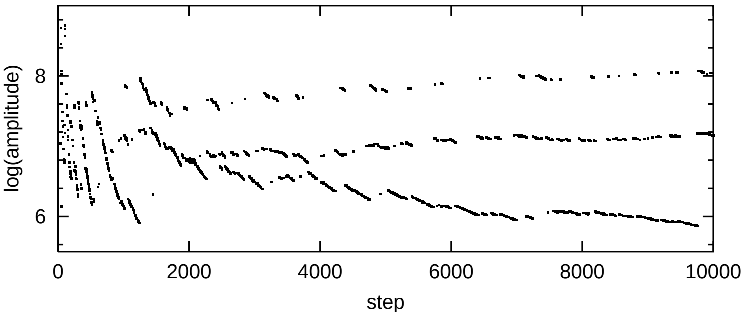


Figure 8.13: log(amplitudes) of the auditory grains in an McMC process monitoring stream for a mixture of the Gaussians with different a priori probabilities. Volume variations decrease with progressing convergence.

McMC Details Stream

The McMC details stream provides more details on the McMC process. In this stream decaying noise pulses of 2 ms duration are added for rejected propositions. From this, the listener obtains a coarse impression about the efficiency of the McMC process. The distance of the McMC steps to the mode is $d_i^m = \|\mathbf{x}_i - \mathbf{m}_i\|$. This stream includes auditory grains to inform the listener if the McMC simulation moves close to the center or in the tails of a mode by modifying the timbre of the grains in the McMC process monitoring stream. This is achieved by adding grains whose fundamental frequency is 2 times higher than the grain frequency in stream 1 and whose amplitude is mapped from the d_i^m by

$$10 \log_{10}(\text{amplitude}) = \text{map}(d_i^j/\delta_j, [1, 2], [0, 30]) \ , \tag{8.33}$$

assuming that the deterministic step converged to mode with index j. Therefore McMC steps in the tails of a mode sound brighter.

AIB Stream

The AIB stream gives an acoustic summary of the history of the McMC simulation with respect to the modes. Each McMC step contributes to its respective AIB as pointed out above. If the AIB counter exceeds a threshold T_b, an AIB event is generated. T_b is set to a constant value, so that the average number of AIB events per minute is small enough to avoid overlaps. In the current design AIB events have a duration of less than 0.5 s and are introduced with a pitched tone that allows associating the mode to the AIB. The mode marker duration is determined by

$$\text{duration} = \text{map}(\delta_j, \left[0, \sqrt{\text{Tr}(\mathbf{S}_j)/d}\right], [0.3, 1]) \,. \tag{8.34}$$

so that modes with wide tails are introduced with long tones. After 100 ms, an uprising chain of percussive tones is started. These represent the ordered eigenvalues $\lambda_k, \;\; 1 \leq k \leq d$ of $\mathbf{S}_j$ which is the sample covariance matrix of all McMC observations attracted to mode j. The number of tones n played within this arpeggio is determined by

$$n = |\{l : \sum_{k=1}^{l} \lambda_k < 0.9\,\text{Tr}(\mathbf{S}_j)/d\}| \,. \tag{8.35}$$

In that way, the number of tones gives some information about the shape of the mode. The tones within the arpeggio are located on a 50 ms time grid and the pitch of the kth tone is given by

$$\log_2(\text{frequency}) \approx \text{pitch} = \text{map}(k/d, [0, 1], [12, 13]) \,. \tag{8.36}$$

8.3.4 Examples

This section provides sound examples for synthetic and real-world datasets. To begin with, a sonification example using Method 1 is given. All further sound examples are generated with Method 3. To facilitate the understanding of the sonifications, the sound examples are split into different pieces so that it is easy to follow the different auditory streams.

McMC Sonification using Trajectory Audification

Sound example MCMC-Ex-2 (Track 13) is a sonification that uses audification of the squared particle velocity, described above as Method 1. For the example, a mixture of 3 Gaussians with different variances and mixing proportions was taken for p. Every 10th McMC step is audified, so that successive sounds are almost uncorrelated. Figure 8.14 shows a spectrogram of the sound signal. Particle audifications are perceived as percussive sounds. Due to the non-harmonic mode shape, the sound is spectrally complex in its attack phase. The decay is spectrally simpler because the mode can be approximated by a quadratic form close to the mode center leading to harmonic motion.

McMC Sonification for Cluster-Analysis

This section presents sound examples for McMC sonification using Method 3. A dataset $\mathbf{X}$ with samples drawn from a mixture of 3 normal distributions in $\mathbb{R}^6$ with different covariance and a priori probabilities is used. The target density p is computed by kernel density estimation with a spherical Gaussian kernel with bandwidth $\sigma = \sqrt{0.2\text{Tr}(\hat{\mathbf{C}}(\mathbf{X}))/d}$. The jumping distribution is $\mathcal{N}\{0, \sigma^2 \boldsymbol{I}_6\}$ with $\sigma^2 = 0.8\text{Tr}(\hat{\mathbf{C}}(\mathbf{X}))/d$. The three auditory streams are presented in separate sound files and it will be discussed what can be perceived in the different streams.

For the first sound example 100 McMC steps are played per second ($T = 0.01$ s/step). Figure 8.15 shows the sound signal for the three auditory streams.

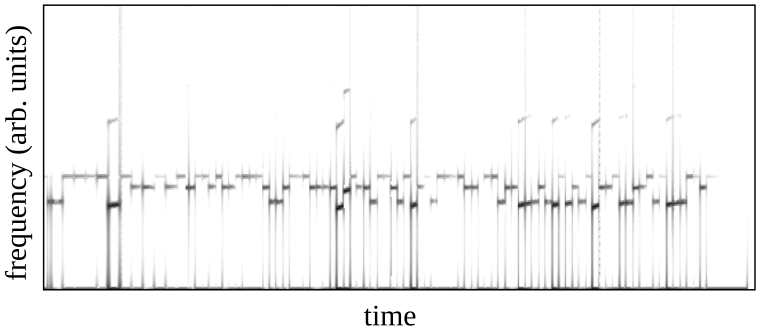


Figure 8.14: Spectrogram of a McMC sonification for 100 McMC steps in a density p given by a mixture of 3 Gaussians with different variances. The three different pitches correspond to the three modes as described in Section 8.2.

- The McMC process stream can be heard in sound example MCMC-Ex-3.1 (Track 14). Three pitches can be discerned. The pitches correspond to to the three modes. Amplitude changes can be perceived while the McMC chain stays close to a mode. In the beginning, some high pitched percussive sounds can be heard: this are marker sounds played on each detection of a new mode of p.

- The McMC details stream (sound example MCMC-Ex-3.2 , Track 14) consists of a noisy continuous sound pattern. Each noise burst indicates a rejected proposition and obviously the rejection rate is rather independent of the McMC step. This is caused by the large jumping variance. By reducing σ^2 of J, the rate of the noisy ticks gets smaller. In this stream, you can also hear pitched auditory grains similar to those in the first stream. The louder they are, the bigger the distance of an McMC step to its mode is. In combination with stream 1, this causes an increased brightness of the auditory grains in stream 1.

- The AIB stream (sound example MCMC-Ex-3.3 Track 14) contains the AIB sound events. Again, 3 different pitches can be resolved. The middle one is the most frequent, indicating that this mode has the highest probability mass (mixing proportion). From the arpeggio one can conclude that this mode also exhibits variability along more than 3 directions.

- A mix of all auditory streams is available in sound example MCMC-Ex-3.4 (Track 14).

The next example is a sonification of the same density, now using $T = 0.002$s, so that 6000 McMC steps are presented in 12 s. Sound examples are given as above for the different streams (Track 15) (MCMC-Ex-4.1 (stream 1) , MCMC-Ex-4.2 (stream 2) , MCMC-Ex-4.3 (stream 3)) and for a mix of all streams (MCMC-Ex-4.4) Transitions between the clusters with medium pitch and high pitch are very frequent in comparison to transitions between the mode of low pitch and high pitch. Indeed, the corresponding cluster centers have a smaller distance. As a last example a McMC sonification for a different density with 6 modes is given. One McMC step lasts 8 ms. All 6 modes are detected by the simulation. Sound Examples: (Track 16) MCMC-Ex-5.1 (stream 1) , MCMC-Ex-5.2 (stream 2) , MCMC-Ex-5.3 (stream 3) , MCMC-Ex-5.4 (mix).

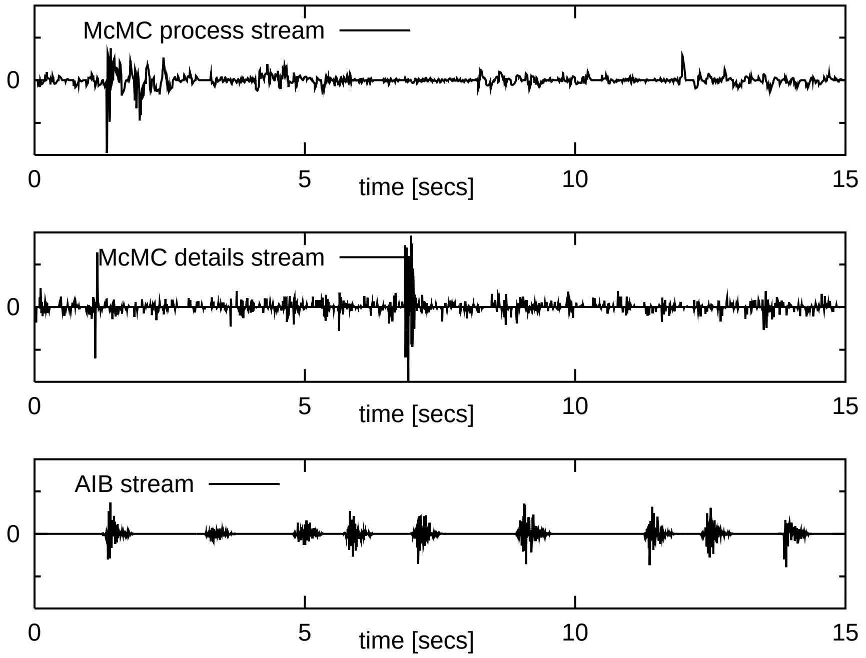


Figure 8.15: The signal plot of sound example MCMC-Ex-3 presents a rough outline of the content of the different auditory streams and may help new listeners to distinguish parts in the mixed sound track.

McMC Sonification for the Iris Dataset

This example presents McMC sonifications for the Iris dataset (see p. 10). After removing the class label, the data points group in two separated clusters, one for class 1 and one cluster containing the data points of the other two classes. The McMC sonification allows to perceive this bimodal structure.

The density is again computed by kernel density estimation with a spherical Gaussian kernel with bandwidth σ. The following set of examples demonstrates the influence of σ on the sound: (all examples in Track 17)

- For large values for σ^2, p is unimodal, as can be heard in the sound examples MCMC-Ex-6.1 and MCMC-Ex-6.2 .
- Reducing σ^2, p at first becomes bimodal. This can be clearly heard in the sound examples MCMC-Ex-6.3 , MCMC-Ex-6.4 , MCMC-Ex-6.5 , MCMC-Ex-6.6 . Obviously, the bimodal structure is quite stable under variation of σ^2.
- By further reducing σ^2, more and more local maxima are created and detected by the McMC. This is audible in the sound examples MCMC-Ex-6.7 and MCMC-Ex-6.8 .

8.3.5 Conclusion

McMC sonification is a new tool for monitoring McMC simulations and to explore the structure of a given high-dimensional density function. The sonification summarizes information about the modes, their numbers and their density in an auditory display. The sonification follows a model-based approach and is formulated in a way that makes it possible to apply it to arbitrary densities. Given that the parameter mapping settings are not changed any further, McMC sonification allows the listener to familiarize with the sound of the model and to improve his performance in relating sound to features of p.

8.4 Principal Curve Sonification

Principal Curve Sonification (PCS) is an auditory display for high-dimensional datasets developed in collaboration with P. Meinicke [HMR00]. Analogous to binocular vision where two views of a scene from different points in space are used to build a 3-dimensional impression, auditory perception profits from listening to a soundscape at different points in time. Human listeners for instance move their head frequently to increase spatial resolution in detecting sound sources. Subtle changes of the soundscape while a listener moves around also improves orientation. Blind people use such auditory clues for instance in order to detect if they are approaching a wall. Such listening skills are addressed in PCS.

Auditory display of high-dimensional datasets have to address the question of how to use time in sonification, as it is inconvenient to play all the data simultaneously. Sound may be regarded as a one-dimensional medium where time defines the axis. The data may be regarded as points in a high-dimensional space. The mediator between space and time is motion. Moving a virtual listener (or microphone) in data space provides an intuitive serialization of the data by playing them in relation to the position of a virtual listener. In our physical world, objects move on continuous paths. Therefore, it is plausible to move the virtual listener (or a microphone) on a curve in data space. This basically is the model of *curve sonification*. As a particularly well suited choice for a trajectory in data space the *principal curve* (see Section 2.3.5) of the data may be used.

In the implementation of PCS at hand, a scatter plot of the dataset is displayed to the user. The curve is visualized in the plot as well as a line. As a first implementation of PCS, the virtual listener moves along the curve at constant velocity and an auditory marker is played for each data point when its projection position on the curve is passed. A more elaborate approach is proposed which also integrates auditory clues that are observed while moving in the real-world as e.g. the Doppler effect and level increase and decay on approaching or departing from a sound source.

Principal Curve Sonification (PCS) can be used for various purposes: firstly it provides information about the clustering of high-dimensional data and the relation of the clusters to each other. Another application is to monitor features or variables for systematic variations along the curve. PCS may be used to detect outliers or to compare the distribution of two given datasets. Appropriate specializations are presented in the following.

Finally, PCS is also simply a means for monitoring the progress of Principal Curve learning which helps to value the quality of the curve during the adaptation. Especially in a case where no criterion exists to determine what length or complexity is suitable, it is up to the user to select a curve that does not “overfit” the data. Here, PCS may be a valuable exploratory tool to assist finding a decision.

8.4.1 Model Description

The PCS model is defined for datasets $\mathbf{X} \in M(N \times d, \mathbb{R})$ with row vectors $(\mathbf{x}_i^{\mathrm{T}}, \mathbf{y}_i^{\mathrm{T}})$, $i = 1, \dots, N$ containing the records. The dataset is split into an input part $\mathbf{x}_i \in \mathbb{R}^{d_{in}}$ containing independent columns and an output part $\mathbf{y}_i \in \mathbb{R}^{d_{out}}$ for dependent components. The distinction is made since only the input part is used to determine the spatial layout of the model.

Setup

PCS is a spatial model using a d_{in}-dimensional Euclidean model space. Each data point $\mathbf{x}_i$ is represented by an object in model space at coordinates $\mathbf{x}_i$. The acoustic properties remain undefined in the basic model setup. One may for instance have in mind that each object may produce a continuous sound whose properties depend on the output part of the dataset. Particularly when Parameter Mapping Sonification is applied to determine the object sound, this allows an efficient sound synthesis.

The second element of the setup is a continuous curve in model space. Description and representation of curves in high-dimensional spaces have already been discussed in Section 2.3.5. As a special case of the general definition for curves in $\mathbb{R}^d$ here polygonal lines, represented by an ordered set of vertex vectors $\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_M\}$ are used.

The length of such a curve $\mathbf{f}$ is computed by

$$l(\mathbf{f}) = \sum_{i=1}^{M-1} \|\mathbf{v}_{i+1} - \mathbf{v}_i\| \,. \tag{8.37}$$

and a unit-speed parameterization $\mathbf{f}(\lambda)$ of the curve is used [HS89]. Practically, for a given index λ, the point on the curve is found by interpolating between the points $\mathbf{v}_a$ and $\mathbf{v}_{a+1}$ for which the corresponding arc length is just smaller (resp. larger) than λ. Thus $\mathbf{f}(\lambda)$ can be computed for all values $0 < \lambda < l(\mathbf{f})$.

For any point $\mathbf{x}$, a projection index is now defined by

$$\lambda_f(\mathbf{x}) = \max\{\lambda : \|\mathbf{x} - \mathbf{f}(\lambda)\| = \min_s \|\mathbf{x} - \mathbf{f}(s)\|\} \tag{8.38}$$

which is the largest one of all indices whose curve point is closest to $\mathbf{x}$.

Before PCS can be used, a curve must be computed. An algorithm for learning principal curves was presented in Section 2.3.5.

The last component of the model is a virtual listener, resp. microphone which moves along the curve. For its characterization, at least the coordinates are needed. As the microphone moves only on the curve, it suffices to specify the projection index λ_m. A more realistic model would also require defining the orientation (e.g. of the head of the virtual listener in model space). Alternative approaches for this will be addressed later.

The model aims at defining a spatial and temporal relation between the data-driven objects and a virtual listener on the curve. So far, no dynamical elements have been introduced that cause sound generation. At this point, different specializations are possible to connect the model to sound generation:

- Dynamical Model Elements: multi-dimensional oscillatory systems are attached for each data point as defined in the Data Sonogram model.
- Parameter Mapping: "excitation" of object i triggers the generation of a sound that is determined by the output vector $\mathbf{y}_i$ by using Parameter Mapping.
- Stationary Sound Patterns: an acoustic pattern is defined for each object which may include rhythmical and harmonic elements to represent a data point.

To demonstrate feasibility, the second approach is adopted. Dynamical Model Elements were not used as they are computationally very demanding, and stationary sound patterns have not been addressed so far.

Dynamics

The only dynamical element in this model is the listener. In the current implementation, the listener moves along the curve with a constant velocity, starting at $\mathbf{f}(0)$ and ending at $\mathbf{f}(l(\mathbf{f}))$.

Initial State

The listener is positioned on $\mathbf{f}(0)$, the starting point of the curve. In the Stationary Sound Patterns approach, the object sounds are started at $t = 0$.

Excitation and Interaction Types

The user can only change the perceived soundscape by moving on the curve. This can be realized by giving the user control over velocity and direction. The currently implemented user interface, however, only allows the user to specify the duration T for moving once along the PC with constant velocity. This was chosen because real-time control was not feasible at the time the model was set up. The sonification is computed offline and played back without the possibility of having any interaction afterwards. In the Parameter Mapping approach, the objects are excited at that point in time when the listener passes their projection index, which triggers the generation of the corresponding object sound.

Model-Sound Linking

The sound is determined by a sound generation algorithm that must be designed and mapped either from the data available or observables of interest. As the object sound sources are precisely localized in model space, however, the mechanism of sound propagation in model space to the virtual listener has to be addressed.

Listener

The object sound amplitude is attenuated during propagation to the virtual listener by $1/(r + \epsilon)^q$ where $r = \|\mathbf{x}_i - \mathbf{f}(\lambda_m)\|$. q has to be specified by the user. ϵ prevents the amplitude from diverging for small distances r. Representing the virtual listener as a single point only allows to compute a monaural sonification. In everyday listening we use binaural clues to localize sound sources, and we perceive pitch changes on relative motion to a sound source, explained by the Doppler effect [BS74]. Both effects may be included in PCS. The Doppler effect is quite easily transferred to high-dimensional settings. Integration of spatial clues, however, requires that the virtual listener is assigned a head orientation and a head size in model space. There is no intuitive way to do this, particularly as the dimensionality of model space differs in general from 3. Two alternatives for listener orientation are:

- Accompanying coordinate system: the virtual listener is looking along the tangent of the curve. In smooth curves the acceleration vector $\ddot{\mathbf{f}}(\lambda)$ is orthogonal to the tangent if the listener moves at constant velocity. Both vectors determine a 2d plane. The head of the virtual listener is oriented along the tangent so that the ears are in the plane. The objects are projected on the plane to determine their relative position (azimuth angle and distance). Experiments with this approach have shown that spatialization is confusing since the hearing plane changes its orientation frequently. Such spatialization here did not prove helpful for understanding the spatial structure of the data.

- Fixed coordinate system: The virtual listener is always oriented along the y-axis of the shown 2d scatter plot so that the ears are in the projection subspace of the graphical display. The objects orientation relative to the virtual listener is taken from their relative position in the plot. Although this spatialization method is much easier and it seems to be more unnatural to move around while keeping the orientation unchanged, the auditory information is better suited to interprete the sound with respect to the data.

Additional Auditory Elements

So far, the sonification consists of a single auditory stream that contains the object sounds mixed together. PCS is now extended with two additional auditory streams: (i) a stream that summarizes local distance and local density, and (ii) a stream that gives information about the curve itself like its curvature. Both streams are synthesized by using time-variant oscillators, whose frequency and amplitude are continuously controlled by the observables. Stream (i) applies a sinusoidal waveform and maps the average distance of objects to the virtual listener to sound frequency. A local density estimate at the position of the virtual listener determines the amplitude. The mapping causes that the sound automatically diminishes in regions of the curve where no data is available to compute an accurate average distance.

Stream (ii) consists of a rhythmic noise pattern. A local estimate of the PC curvature is mapped to the rate of the pattern. For smooth curves, curvature is determined as $\ddot{\mathbf{f}}(\lambda)$. For the adopted representation of curves by polygonal lines, the curvature may be computed by using the angle α between three points on the curve

$$\cos(\alpha(\lambda)) = \frac{(\mathbf{f}(\lambda - \Delta\lambda) - \mathbf{f}(\lambda)) \cdot (\mathbf{f}(\lambda + \Delta\lambda) - \mathbf{f}(\lambda))}{\|\mathbf{f}(\lambda - \Delta\lambda) - \mathbf{f}(\lambda)\| \|\mathbf{f}(\lambda + \Delta\lambda) - \mathbf{f}(\lambda)\|} \; . \tag{8.39}$$

With $\Delta\lambda \approx 2l(\mathbf{f})/M$, an appropriately smooth curvature observable is obtained. For the computation of curvatures close to the endings of the curve, the curve is linearly extrapolated beyond the first or last line segment.

Data-Model Assignment

The available information can be organized into three categories: complex observables, attributes of the data point and properties of the PC. Table 8.5 summarizes the used variables and their acoustic representation. The selection of auditory elements and their mapping from features is a subjective choice. The benefit of the sonification model however is that this selection has to be made only once and then PCS can be used without the need for any further parameter tuning for a large class of datasets. As a consequence the user of PCS has the chance to familiarize with the sound and to improve his sound interpretation-skills. The choice of synthesis technique, sound parameters and mapping depends on the approach and for each of them one possibility will be given.

Dynamical Model Elements: a point mass m moves in a d_{out} dimensional potential given by $V(\xi) = \sum_i^{d_{out}} k_i \xi_i^2$. The dynamics is identical to the dynamics used in the data sonogram model.

	Feature	Sonification
Observables:	Local probability density	Time-variant sine oscillator amplitude
	Local intrinsic dimensionality	-
	Local averaged distance to PC	Time-variant sine oscillator pitch
Data Points:	Relative orientation to the listener	Spatialization of tick sounds
	Distance from PC	Volume of tick sounds
	Data Features (e.g. class label)	Frequency of tick sounds
PC properties:	Velocity on PC	Wavetable sound volume
	listener orientation	Intensity panning
	Curvature of PC	Frequency of noise pattern

Table 8.5: Information given in PCS in the three auditory streams.

The solution is a superposition of d_{out} modes of vibration with frequencies $f_j = \sqrt{k_j/m}/2\pi$. The stiffness constants are mapped by the features y_{ij} by

$$k_j = \text{map}(y_{ij}, [\min_i y_{ij}, \max_i y_{ij}], [\hat{k}_j, \hat{k}_j + \Delta k]) \ . \tag{8.40}$$

Parameter Mapping: A function $s(t, \boldsymbol{\theta})$ is used for sound generation. The parameter vector $\boldsymbol{\theta} \in \mathbb{R}^q$ describes the acoustic attributes and is mapped from $\mathbf{y}_i$ by a mapping function $\mathbf{g} : \mathbb{R}^{d_{out}} \to \mathbb{R}^q$. To give an example, let $s(t, \boldsymbol{\theta}) = \theta_0 t + \theta_1 \sin(\theta_2 t) \exp(-\theta_3 t)$. The sound for an object i is then given by $s(t, \mathbf{g}(\mathbf{y}_i))$.

Stationary Sound Patterns: The sound of each object i is a continuous sound pattern, also called auditory texture, which is determined from the $\mathbf{y}_i$. To give an example, let

$$s(t, \boldsymbol{\theta}) = \sum_{j=1}^{d_{out}} \theta_j \sin(2\pi\nu_0 jt) \sin(2\pi\nu_1 jt) \tag{8.41}$$

so that ν_0 is in the audible frequency range and ν_1 about 1 Hz. θ_j are determined from y_{ij} by a suitable mapping. In contrast to the Parameter Mapping approach above, these sounds are played during the whole sonification time. The perceived level however, depends on the distance of an object to the moving virtual listener.

8.4.2 Implementation

Computation of PC and the PCS is done separately. PCS is computed offline by computing a sound vector $s[n]$ representing the sonification for a single path from beginning to end of the curve. Computation takes a number of seconds and after it is finished the sound is played to the listener. The benefit of computing the PCS offline is that there are no problems with real-time sound synthesis and that the range of observable values, like average local probability density is known and can be used to determine the mapping to sound parameters automatically.

8.4.3 Examples

This section presents some sound examples for synthetic and real-world datasets in order to demonstrate usage and typical sounds of PCS.

In the first example, a noisy spiral dataset in $\mathbb{R}^3$ is used. Figure 8.16,(a) shows a scatter plot of the dataset. The principal curve approximates the data quite well. The PCS for this dataset is given in sound example PCS-Ex-1.1 (Track 18). A spectrogram of the sound signal is shown in

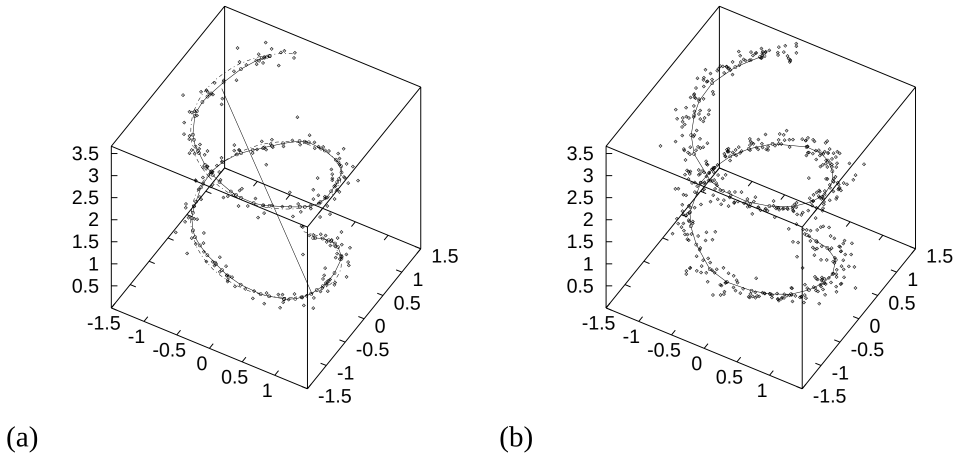


Figure 8.16: Principal curve (solid line) for the noisy spiral. The generator curve is shown as a dotted line. (b) noisy spiral with variance modulation. The modulation is easily overlooked in the visual display.

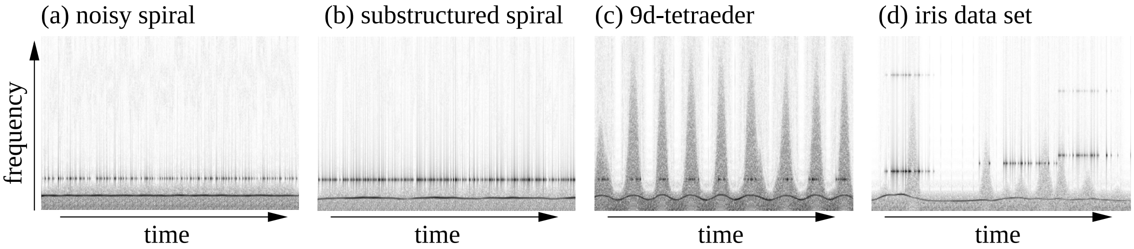


Figure 8.17: Spectrogram of the PCS examples (See text for further comments).

Figure 8.17. The sound indicates that the local density (pitch of the time-variant oscillator), the distribution of data along the PC (heard as tick rate) and the curvature (brightness of the noise) are almost constant. This indicates a uniform distribution on a 1d curved manifold. The sound indicates that the data do not cluster into different groups.

The next sound example uses a modified noisy spiral in which the noise variance is modulated periodically along the spiral. As this can hardly be seen in the scatter plot (see Figure 8.16, (b)), it is an example of a pattern that is likely to remain unnoticed in a visual display. In contrast, it is easily detected in the sonification. In sound example PCS-Ex-1.2 (Track 18) the variations of density are clearly perceived as pitch and level variations of the time-variant oscillator.

The next two examples demonstrate the application of PCS for cluster analysis. The first dataset is a mixture of 10 clusters in $\mathbb{R}^9$ whose mass centers are the 10 vertices of a 9-dimensional hyper-tetrahedron (see page 10). The PCS (sound example PCS-Ex-2 , Track 19, spectrogram in Figure 8.17,(b)) allows to count the number of clusters and to compare the cluster size, e.g. by attending the tick density. Furthermore, the relative distances between clusters can be perceived as the rhythmical structure of the sound. Obviously, the high symmetry of the dataset is maintained in the sound: as all clusters have a constant pairwise distance the time between the ticks for two clusters is also always constant. A 2d scatter plot of the dataset (Figure 8.18) is unable to show this symmetry.

The last example is a PCS of the Iris dataset [Fis99]. The PC is computed from the 4-

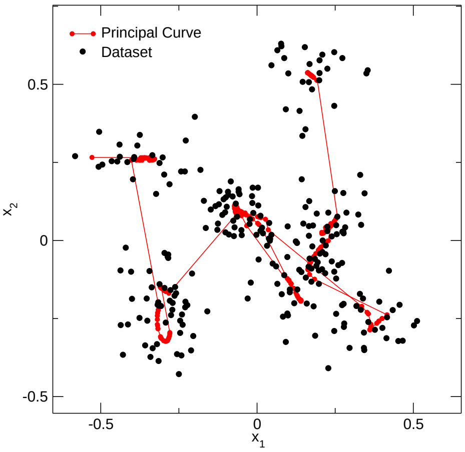


Figure 8.18: Scatter plot of a dataset with clusters aligned to the vertices of a 9d hypertetrahedron. The scatter plot fails to resolve the 10 clusters. The PC passes the 10 clusters one by one.

dimensional feature vectors, the class attribute is excluded. Instead, it controls the pitch of the object sounds. The sound example is PCS-Ex-3 (Track 20) with spectrogram in Figure 8.17,(d). It can be heard that the Principal Curve passes through the three groups of Iris plants one by one. Particularly, the gap between the first and the second group becomes audible and indicates the separability of the classes.

Psychophysical Validation

To check the hypothesis that PCS facilitates the detection of such kind of structures in data as the variance modulation in the modified noisy spiral example, a psychophysical experiment was carried out with 15 human subjects. The task consisted in detecting the number of noise variance modulations in noisy spiral datasets as shown in Figure 8.16. Controlled variables were the presentation type (visual/auditive/both) and the intensity of the modulation. The performance on 90 datasets was evaluated for each subject. The relative error was measured. That is the number of wrong answers under each condition, divided by the total number of wrong answers of a participant. Figure 8.19 shows the results. The relative error is significantly reduced on addition of the auditory display. Processing time was measured for each example. The timer was started at the time the plot was displayed (resp. the sonification was started) and stopped when the user gave the answer by selecting one of 5 buttons. Possible answers were: no, 3, 4, 5 or 6 variance oscillations along the spiral. As subjects needed a different length of time for the experiment, the relative processing time was used for further analysis. This is the processing time per example divided by the time for all answers. The average relative processing time decreased by 31% on addition of PCS.

8.4.4 Conclusion

PCS is a new technique to render auditory representations for high-dimensional datasets. The motivation for the model was to find a “natural meaning” for the time axis while presenting data by sound. This was achieved by developing the model of curve sonification which connects time with a corresponding position of a virtual listener on a trajectory in data space. The Principal

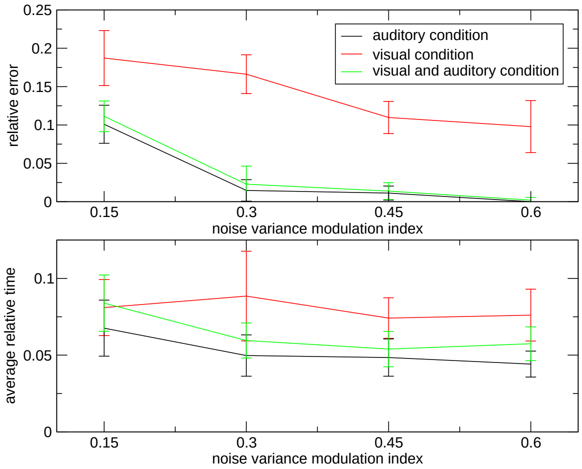


Figure 8.19: Results of the psychophysical experiment to compare user performance on detecting variance modulations in the noisy spiral dataset. The relative error decreased significantly ($\alpha = 1\%$) on addition of PCS. The error bars show the standard deviation. The second plot shows the average relative time needed to give an answer. Obviously, subjects performed faster with the auditory display.

Curve is an intuitive and sensible choice for this trajectory since like time it is a one-dimensional manifold. In contrast to other linear curves like the first principal component axis the PC is more flexible and thus able to approximate the data better.

The concept of curve sonification, however, is not restricted to principal curves. For instance the connected chain of neurons in a 1d SOM can be used as well, allowing to carry this concept over to the auditory presentation of neural networks and their adaptation.

A psychophysical experiment was carried out, which showed that PCS does provide useful information about the dataset. The current implementation of PCS can be improved in two regards: firstly, the sound is currently not connected to the visualization. Even a simple graphical marker that moves along the visualized curve during playback of the sonification would enhance the comprehension of the sound to the data greatly; and secondly the missing interactivity: if the user could move interactively on the curve, this would enable him to to inspect parts of the dataset at the resolution and level of detail he prefers. These extensions will be subject of following work.

8.5 Data Crystallization Sonification

The Data Crystallization Sonification (DCS) model is a technique to browse high-dimensional data for exploring local and global features of the data, particularly concerning the intrinsic data dimensionality (see Section 2.1).

Regarding the sound of musical instruments, we find that such sounds can often be divided into two parts: a transient phase and a stable state that corresponds to quasistationary oscillations. For the identification of instrument classes, the transient phase is particularly important. The dynamical evolution of timbre during the transient phase contains specific information about the instrument.

The aim of this model is to define a dynamical complex sound pattern so that the global data distribution determines the "instrument class" whereas local properties determine the transient phase of the sound. For this purpose a spatial model is used, in which each data point is represented by a point in model space. A crystallization process is started at a “condensation nucleus” selected by the user in a scatter plot of the given dataset, which includes neighboring points successively into the growing “crystal solid”. With each point inclusion, the acoustic properties of the crystal solid changes. The dynamics is defined so that the sound corresponds to the intrinsic data dimensionality. The sonification ends when all data points are included into the solid. As this process resembles crystallization starting at the condensation nucleus, this metaphorical name was chosen for the model.

Modal synthesis is applied to generate the sound. The modes are taken to be multiples of the fundamental frequency (harmonics). The energy within the modes is determined by the eigenvalue spectrum of the crystal dataset covariance matrix. The DCS Model provides information about the intrinsic dimensionality of the data. Furthermore, although this is a merely side effect, the clustering of data can be perceived.

8.5.1 Model Description

Setup

The Data Crystallization Sonification (DCS) Model is based on a spatial model. The model space is an $\mathbb{R}^d$. Each data point is represented by a point (or atom) located at $\mathbf{x}_i, \;\; i = 1, \ldots , N$.

The data points remain fixed during the sonification and thus constitute no degrees of freedom. The model of a crystal is only used as a metaphor to simplify the interpretation of the sonifications.

Dynamics

Starting at a vector $\mathbf{x}_c$ (condensation nucleus) which is specified by the user, a crystal growth begins. This means that data points are successively included into the set of points which belong to the crystal. There are various possibilities for data point inclusion. One method is to order all points according to their distance to $\mathbf{x}_c$ and include them in this order. Alternatively, one could include at each step the point with the smallest distance to the crystal. Generally, the different strategies for agglomerative clustering [JD88] can be applied here.

To begin with, the following simple inclusion rule is used: inclusion of a point with coordinate vector $\mathbf{x}$ is done at sonification time

$$t = \text{map}(\|\mathbf{x} - \mathbf{x}_c\|, [0, \max_i \|\mathbf{x}_i - \mathbf{x}_c\|], [0, T]) \tag{8.42}$$

where T is the total duration of the crystallization process.

Initial State and Excitation

Let $I_c(t)$ denote the subset of data points in the crystal at time t. The initial conditions are given by the empty set $I_c(0) = \{\}$. As no crystal exists at the beginning, the initial sound is silence. Excitation in this model means to initiate a crystallization process, which is done by selecting a condensation nucleus.

Excitation and Interaction Types

The user interacts with the sonification system by using the mouse pointer. Clicking the mouse into a 2d scatter plot of the dataset triggers the sonification. For the condensation nucleus vector $\mathbf{x}_c$ the coordinates of the nearest neighbor to the mouse coordinates in the scatter plot are taken. Alternatively, the mouse pointer coordinates can be used to determine the position within the visualized subspace and column means are used for the other vector components.

Model-Sound Linking

The sonification is computed from the temporal evolution of the crystal. As no explicit dynamics is defined, the sound pressure is not bound directly to any dynamic element. Details about the connection from crystal properties to sound properties will be described below.

Listener

The listener is not localized in regard to the model space and thus a monaural sonification is rendered. A second (stereo) audio channel thus remains free for other purposes as for instance to compare two sonifications which started from different points or to compare the sonification of two datasets.

Sound Synthesis

Modal Synthesis is applied for sound computation. This means that the sonification is computed by using additive synthesis

$$s(t) = \sum_{i=1}^{d} a_i(t) \sin(\omega_i(t)t + \phi_i) \tag{8.43}$$

where the mode frequencies ω_i and the amplitudes of these oscillations a_i are time-variant functions.

For the implementation, the time series $a_i(t)$ and $\omega_i(t)$ are discretized using control points on a 5 ms grid. The synthesis is done by superimposing the modes and storing the result in an output vector $s(t)$. $a_i(t)$ and $\omega_i(t)$ are interpolated between control points. The basic building block of the sound thus is a time-variant sine oscillator with amplitude and frequency control. All initial phase values ϕ_i are set to 0. They are not relevant for the auditory perception since all modes are in the range of audible frequencies and multiples of the fundamental frequency, so that rhythmical elements like beats can not occur.

Data-Model Assignment

The model determines the inclusion order of data points into a growing crystal. This does not suffice to determine the sonification. It must additionally be defined how the crystal's acoustic properties depend on the elements included.

The aim of the model is to encode information about the crystallization process into the temporal and spectral evolution of sound. In order to do so, a closer look at the temporal and timbral evolution of physical sounds will now be taken. Many musical instruments produce almost periodic oscillations of air pressure. They can thus be characterized by a Fourier series. The modes are harmonic oscillations whose frequencies $f_n = nf_1$ are multiples of a fundamental frequency f_1.

A simple description of timbre can be given by the vector of partial tone amplitudes $(a_1, a_2, ...)$, also called timbre vector (see Section 6.2.1). Figure 8.20 shows a timbre vector for a violin and a metal flute.

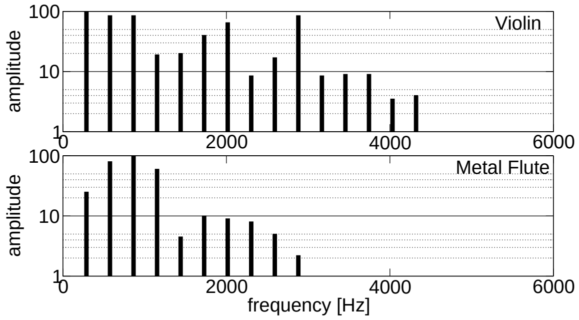


Figure 8.20: Timbre of two musical instruments, characterized by their timbre vectors. The plot shows the amplitudes of the partial tones, which are also called harmonics. Data from [BS74].

However, synthesized periodic signals that use the constant timbre vectors sound different from the real instrument because the temporal evolution of timbre is missing which is also an essential part for the characterization of musical instruments. Especially the attack phase (usually the first 50-80 ms) is important for humans to classify and distinguish musical instruments. Another important aspect for the identification of an instrument is the temporal evolution of sound level called amplitude envelope. In physical systems, timbre and amplitude are often connected with each other. To give an example: with increasing amplitude of a mass attached to a real spring, the oscillation deviates from harmonic motion due to nonlinearities of the force laws and this results in the contribution of higher harmonics in the spectrum. So amplitude and timbre vector are connected. Figure 8.21 shows the transient phenomenon for two instruments. In many instruments, the most important variations of timbre occur in the transient phenomenon.

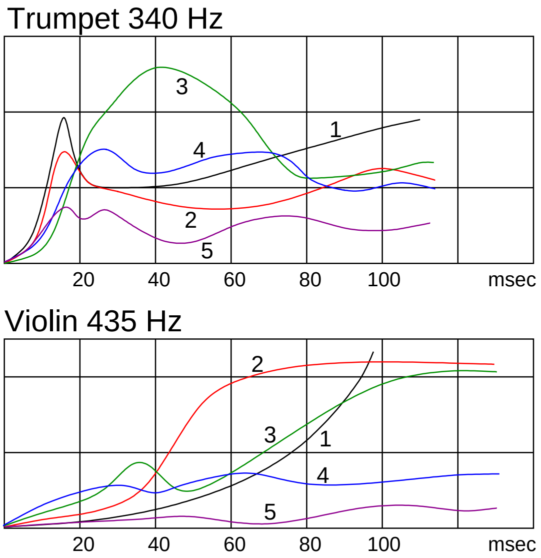


Figure 8.21: Transient Phenomenon for tones from two musical instruments, given by the temporal evolution of the partial tones, here shown for the first 5 partial tones. Data from [BS74].

Whereas instrument sounds are characterized by their timbre vector, in the DCS model the crystal $I_c(t)$ at time t can be characterized by its dataset covariance matrix

$$\mathbf{S}(t) = \frac{1}{|I_c(t)|} \sum_{\mathbf{x} \in I_c(t)} (\mathbf{x} - \bar{\mathbf{x}}(t))^{\mathrm{T}} (\mathbf{x} - \bar{\mathbf{x}}(t)), \tag{8.44}$$

which describes the variance structure of the crystal subset. The crystal mean is here given by

$$\bar{\mathbf{x}}(t) = \frac{1}{|I_c(t)|} \sum_{\mathbf{x} \in I_c(t)} \mathbf{x} \ . \tag{8.45}$$

The idea of the crystallization model is to perceive the temporal evolution of this matrix during the crystallization process. The covariance matrix is known to include information about the intrinsic dimensionality of the dataset. The eigenvalues $\lambda_i, \ (i = 1, \dots, d)$ of $\mathbf{S}(t)$ are the variances along the principal axes of the crystal dataset, as shown in Section 2.3.1.

Using the ordered eigenvalues as strength of the harmonics provides the assignment from data to acoustic properties. For each time t the eigenvalues $\lambda_1(t) \geq \lambda_2(t) \geq \dots \geq \lambda_d(t)$ of $\mathbf{S}(t)$ are computed. The partial tone amplitudes $a_i(t)$ are then determined by $\lambda_i(t)$ using

$$a_i(t) = \left(\frac{\lambda_i(t)}{\sum_i \lambda_i(t)} \right)^p \ . \tag{8.46}$$

To smooth the timbre evolution, values between successive control points are again interpolated. Using $p = 0.5$, the square root in eq. (8.46) forces the acoustic energy $E(t) = \sum_i a_i(t)^2$ to be a constant value. Then the timbre vector moves on the surface of a d-dimensional sphere. The perceptual effect of timbre change can be amplified by using $p > 1$ as will be demonstrated in Section 8.5.3.

Unfortunately, due to the ordering of the eigenvalues, the energy in the fundamental frequency always dominates the timbre vector. The main effect of this mapping is that data distributions with a high intrinsic dimensionality have more energy in higher harmonics of the sound and thus have a brighter timbre whereas data that are distributed along a single dimension sound like pure sine tones.

The mode frequencies $\omega_i(t)$ are integer multiples of the fundamental frequencies ω_0. As a generalization, within this model

$$\omega_i(t) = (1 + (i-1)h_\omega)\omega_1(t) \tag{8.47}$$

is used, where h_ω is the harmonics factor. Setting $h_\omega = 1$ yields the series of harmonics. To enlarge the spectral range, values for $h_\omega > 1$ can be assigned. The acoustic effect of h_ω is to stretch the spectrum and intensify the perception of timbre changes as demonstrated below.

The metaphor of crystallization is furthermore used to derive the amplitude envelope of the sound: assume that the inclusion of each data point into the crystal dataset releases a constant amount of “crystallization energy”. This energy dissipates in acoustic radiation. Thus, the energy stored in the crystal decays exponentially and increases at each point inclusion. The temporal evolution of the energy $E(t)$ is described by

$$\frac{dE}{dt} = -\gamma E + \sum_i g_i \delta(t - t_i) \ , \tag{8.48}$$

where t_i is the inclusion time for data point $\mathbf{x}_i$ and g_i is the amount of energy used for sound from inclusion of data point $\mathbf{x}_i$. The solution is an exponential energy decay with discontinuous steps of size g_i at t_i. One simple possibility is to set $g_i = \text{const}\ \forall i$. As an alternative, one might regard those inclusions as more interesting which change the crystal's variance ellipsoid and use this feature to assign g_i. This can, for instance be done by setting g_i to the change of the first eigenvector of $\mathbf{S}(t)$ on inclusion of $\mathbf{x}_i$ into the crystal. Thus the more a data point inclusion changes the variance ellipsoid the louder the inclusion becomes audible. As a side effect, the stabilization of the main variance axis with growing crystal size causes a smooth decay of sound level.

So far, the fundamental frequency remains constant during the sonification. As a modification of the basic model, the perceptually salient attribute pitch is determined by the total crystal variance. In musical instruments, the fundamental frequency often scales inverse with the length or volume of a string or acoustic tube. Similarly, in this model, higher pitches indicate a smaller crystal (variance ellipsoid). With increasing crystal size, the variance $V_t = \text{Tr}(\mathbf{S}(t))$ stabilizes to $V = \text{Tr}(\mathbf{S}(\mathbf{X}))$. Using for instance

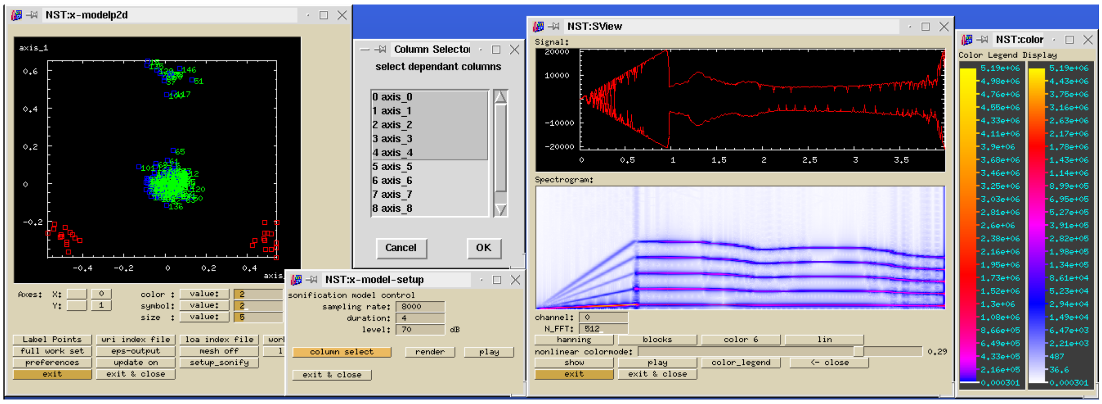
$$\omega_1(t) = \hat{\omega}/\text{map}(V_t, [0, V], [1, q]), \tag{8.49}$$

with $q = 2$, the pitch decreases by an octave during the crystal growth.

A compromise must be found since for very small pitch variations $q \approx 1$, the variance growth cannot be resolved and for very large pitch variations ($q \gg 1$), the timbre variations are perceptually masked. According to the author's experiences, at a pitch range of about 3 halftones a suitable balance is reached.

8.5.2 Implementation

Figure 8.22 shows the GUI and dataset visualization used for the DCS Model. The scatter plot



NST:x-modelp2d

Axes: X: 0

Y: 1

color : value: 2

symbol: value: 2

size : value: 5

Column Selector

select dependant columns

0 axis_0
1 axis_1
2 axis_2
3 axis_3
4 axis_4
5 axis_5
6 axis_6
7 axis_7
8 axis_8

NST:x-model-setup

sonification model control

sampling rate: 8000

duration: 4

level: 70 dB

NST:SView

channel: 0

N_FFT: 512

nonlinear colormode: 0.29

NST:color

Color Legend Display

Figure 8.22: User Interface for controlling the Data Crystallization Sonification Model.

on the left side shows a projection of the dataset onto two axes. The selected points (green) are used for crystallization which starts at the mouse coordinates. The right window visualizes the resulting sound signal and allows to repeat playback of the rendered sound. The "column select" window allows a specification of the axes used for the setup of the model space.

8.5.3 Examples

To demonstrate the sound space of the DCS Model and to introduce the dependencies between the sound and structures in the data, a series of synthetic datasets and a real-world dataset are used.

Gaussian Distributed Data

The first sound examples (in Table 8.6) are crystallization sonifications for the dataset that is sampled from $\mathcal{N}\{0, \boldsymbol{I}_5\}$. Crystallization is started (a) at the center, (b) from the tail and (c) far outside. Figure 8.23 shows the sound signal and its spectrogram for (a).

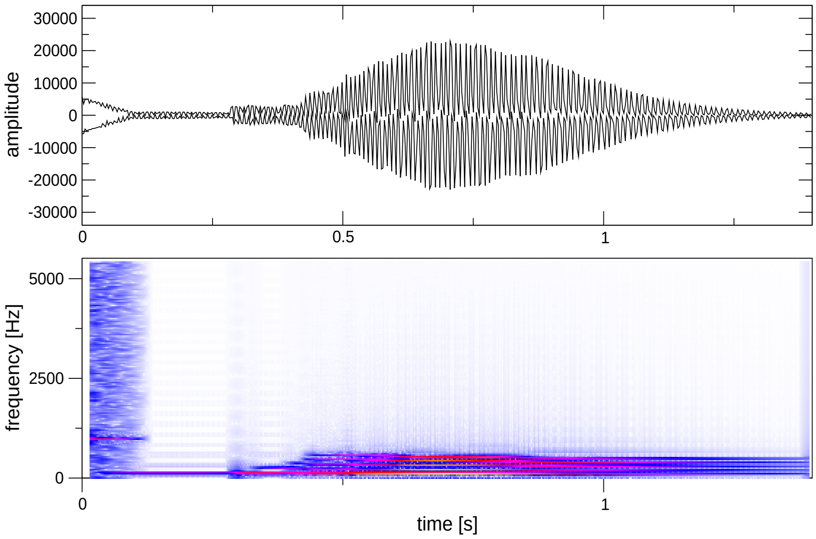


Figure 8.23: Signal Plot and Spectrogram for a DCS for a dataset sampled from $\mathcal{N}\{0, \boldsymbol{I}_5\}$. The condensation nucleus was the mean of the dataset.

File/Track 21:	DCS started at center , in tail , from far outside
Description:	DCS for dataset sampled from $\mathcal{N}\{0, \boldsymbol{I}_5\}$ excited at different locations
Duration:	1.4 s

Table 8.6: Sound examples for Crystallization Sonifications for a dataset sampled from $\mathcal{N}\{0, \boldsymbol{I}_5\}$.

The sound examples illustrate basic aspects of the sonification. At the beginning, a high pitched tick sound is audible which marks the beginning of the sonification. During the sonifications, the progress in crystal growth can be heard from the pitch decay. Obviously, the growth is slower with excitation from the center of the cluster. During the sonification the brightness is constant. This implies that the crystal covariance ellipsoid does not change significantly in shape. The initial sound distinguishes however due to the different location of the condensation nucleus. As expected, in the end all three sonifications sound the same.

Mixture of two Gaussians

This example uses a dataset with 300 records drawn from a mixture of two normal distributions with density $0.3g(\mathbf{x}; -0.75\hat{\mathbf{x}}_0, 0.1\boldsymbol{I}_6) + 0.7g(\mathbf{x}; 0.75\hat{\mathbf{x}}_0, 0.25\boldsymbol{I}_6)$.

Figure 8.24 shows a scatter plot of the dataset and the condensation points used for the following examples. The sonifications depend on the excitation point: sound example DCS-Ex1A (Track 22) starts the crystallization at point A in Figure 8.24, sound example DCS-Ex1B (Track 22) at point B in the smaller cluster. It can be heard that the frequency decay, and thus the variance increase, is faster at the beginning of example A. In example B the pitch remains constant on a plateau which indicates the separation of the clusters. When inclusion of data points from the larger cluster starts, the variance again increases and the pitch decay accelerates. The temporal evolution of timbre vector components is illustrated in Figure 8.25.

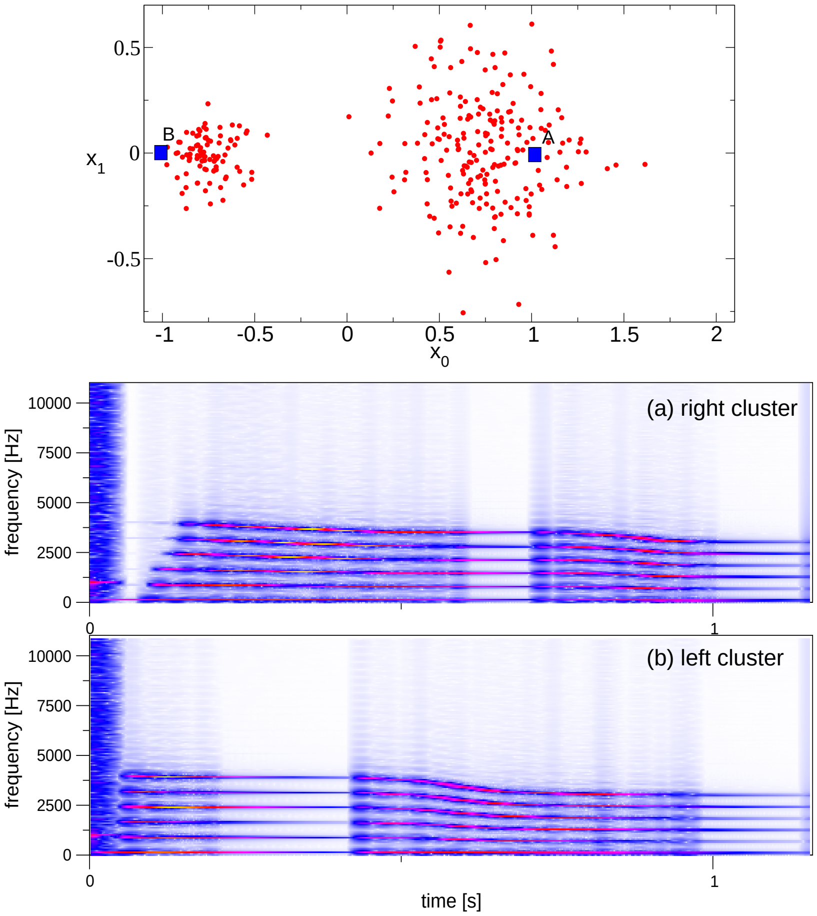


Figure 8.24: Scatter plot of the mixture of two Gaussians in $\mathbb{R}^6$. Below the spectrogram of two DCSs are shown. In plot (a) the DCS started in B, in (b) the DCS started in A

Variation of Harmonics Factor

This example demonstrates the influence of the harmonics factor h_ω on the sound. All sonifications are started from within cluster A in Figure 8.24. With an increasing of the harmonics factor the spectrum enlarges. Table 8.7 contains the sound examples. With increasing h_ω, the spectrum

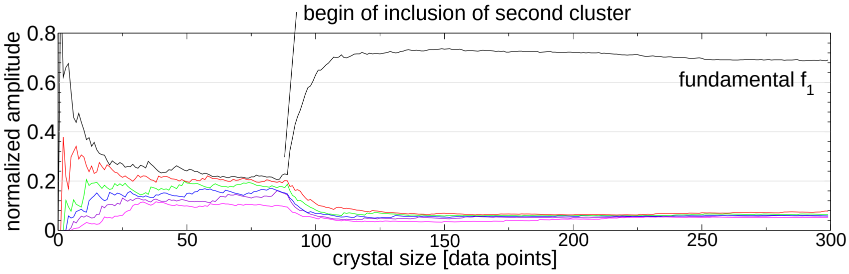


Figure 8.25: Temporal evolution of timbre vector components during crystal growth. The DCS was started at point B in Figure 8.24. Inclusion of points within cluster B leads to a stabilization of the timbre vector. On inclusion of points within cluster A, the timbre vector concentrates on the fundamental since the variance along the first principal axis then contributes most of the crystal variance.

File/Track 23:	h_ω = 1 , 2 , 3 , 4 , 5 , 6
Description:	DCS for a mixture of two Gaussians with varying harmonics factor
Duration:	1.4 s

Table 8.7: Sound examples for DCS on variation of the harmonics factor.

gets more and more stretched.

Variation of Energy Decay Time

The following examples (Table 8.8) demonstrate the effect the energy decay time constant has on the sound of DCS. As the amplitude of all modes decays exponentially with time and each point inclusion increases the crystal energy, a high decay time has the effect of smoothing the amplitude envelope, whereas small decay time constants cause the separation of the point inclusions.

File/Track 24:	$\tau_{1/2}$ = 0.001 , 0.005 , 0.01 , 0.05 , 0.1 , 0.2
Description:	DCS for mixture of two Gaussians varying the energy decay time $\tau_{1/2}$
Duration:	1.4 s

Table 8.8: Sound examples for DCS on variation of the energy decay time.

Variation of DCS Time

Depending on the interest on the data, it is necessary to listen to the sonifications on different time scales. Changing the total duration for the whole crystallization process allows to choose the temporal resolution. The sound examples in Table 8.9 differ in their total duration T. With increasing duration, the granularity of the crystal growth gets more and more audible. However, to facilitate the comparison of sonifications, and to accelerate browsing of datasets, durations less than 2 s proved useful, as the whole sound then fits well into short-term auditory memory.

File/Track 25:	T/s = 0.2 , 0.5 , 1 , 2 , 4 , 8
Description:	DCS for mixture of two Gaussians varying the duration T. ($\tau_{1/2} = 0.05$).

Table 8.9: Sound examples for DCS on variation of the sonification time.

Influence of Data Dimensionality

This sound examples (Table 8.10) demonstrate the influence of data dimensionality on the timbre. The sound examples were rendered by selecting a subset of the columns. With increasing model space dimension the timbre becomes more brilliant.

File/Track 26:	selected columns of the dataset: (x_0) , (x_0, x_1) , $(x_0, \ldots, x_2)$, $(x_0, \ldots, x_3)$ $(x_0, \ldots, x_4)$, $(x_0, \ldots, x_5)$
Description:	DCS for mixture of two Gaussian clusters varying the dimension
Duration:	1.4 s

Table 8.10: Sound examples for DCS on variation of model space dimension.

Intrinsic Dimension

For the following example, a dataset with three clusters (C_0, C_1, C_2) in $\mathbb{R}^{10}$ of intrinsic dimensions $d_{C_1} = 2, d_{C_2} = 4, d_{C_3} = 8$ is used. Figure 8.26 shows a scatter plot and indicates different positions that are used as condensation nucleus.

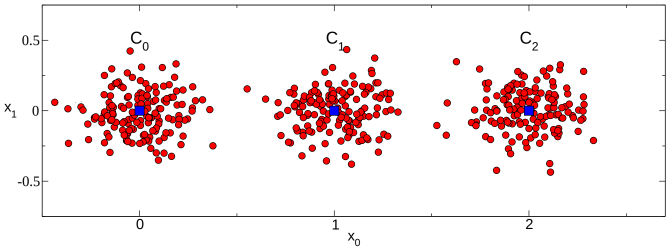


Figure 8.26: Scatter plot of the dataset $(d = 10, N = 450)$ drawn from a mixture of three Gaussians with covariance matrices of different rank.

The sonification examples (see Table 8.11) start the condensation at the points C_0, C_1, C_2 shown in the plot. It can be heard, that the initial brilliance depends on the starting position. However, with proceeding crystal growth, the sound becomes more and more mellow, since the global data distribution is dominated by the 1d-chain that the clusters form. For the sonifications, the parameters $T = 1.5$ s, $\tau_{1/2} = 0.05$ s, $h_\omega = 2$, $p = 2$ and a pitch range of 0.14 octaves are used.

Intersection Manifolds

This example presents DCS for a 5-dimensional dataset sampled from a mixture of a uniform distribution in the (x_0, x_1)-plane with $(-0.5 < x_0 < 0.5, -5 < x_1 < 5)$ and a $\mathcal{N}\{0, 0.04\boldsymbol{I}_5\}$ distribution shown in Figure 8.27.

File/Track 27:	starting point: $\underline{C_0}$, $\underline{C_1}$, $\underline{C_2}$
Description:	DCS for a mixture of three Gaussians in $\mathbb{R}^{10}$ with different rank$(\boldsymbol{S}) = \{2, 4, 8\}$.
Duration:	1.9 s

Table 8.11: Sound examples for DCS for different excitation location.

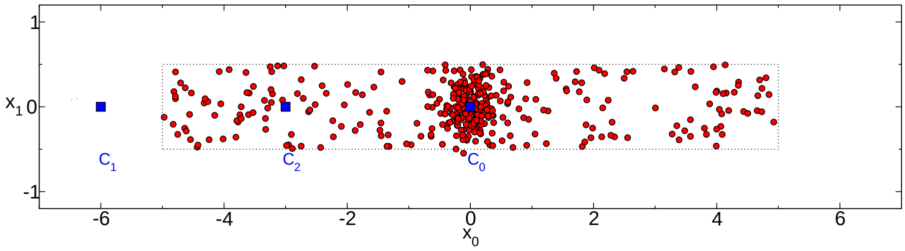


Figure 8.27: Mixture of a 2d uniform distribution and s 5d Gaussian.

DCSs were excited from different points in model space. Sound examples are compiled in Table 8.12. Starting from the edge (C_1, C_2) the initial timbre consists of two harmonics, as the distribution is two-dimensional. With an increasing number of points, the crystal becomes more and more one-dimensional until the data cloud at the origin is included. Inclusion can be perceived from an increase of harmonics and an increased volume. Inclusion of the rest of the data points then again decreases the sound brightness. This sound is very different from the sonification where the condensation nucleus is set to the origin: here the timbre begins complex, with five harmonics and timbre changes, developing to a rather pure sine sound at the end.

File/Track 28:	condensation nucleus in (x_0, x_1)-plane at <u>(-6,0)=C1</u> , <u>(-3,0)=C2</u> , <u>(0,0)=C0</u>
Description:	DCS for the mixture of a uniform 2d and a 5d Gaussian
Duration:	2.16 s

Table 8.12: Sound examples for DCS for a mixture of a 2d distribution and a 5d cluster.

DCS of the Cancer Dataset

The final example applies DCS to the cancer dataset[5]. A scatter plot of the data is shown in Figure 8.28. The sound examples are collected in Table 8.13.

8.5.4 Conclusion

Data Crystallization Sonification is a new technique for auditory browsing of high-dimensional datasets. The model of crystallization defines a growth process in model space which is connected to the dynamical evolution of timbre and other acoustic properties of the sonification and hereby it provides the key for the interpretation of the sound with respect to the underlying data. The sonification aims at using the dynamic evolution of timbre for encoding information about the data. This is very reasonable because humans are trained to interprete such dynamic sound variations as they occur frequently in many environmental sounds like the sounds of musical instruments.

[5] See page 10.

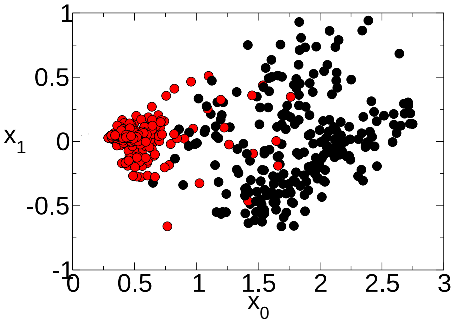


Figure 8.28: Scatter plot showing a projection on the first two principal axes of the cancer dataset. Red points represent benign tumors.

File/Track 29:	condensation nucleus in (x_0, x_1)-plane at benign 1 , benign 2 malignant 1 , malignant 2
Description:	DCS for the cancer dataset for different excitation points.
Duration:	0.8 s

Table 8.13: Sound examples for DCS for the cancer dataset.

As the sonification always starts with localized properties and ends with an auditory presentation of global features, the relative sound change can be used to relate sonifications from different points to each other. The implementation exemplified an indicator for intrinsic dimensionality which was used to determine the timbre vector. This allows to compare the local dimensionality with the global dimensionality of the dataset. However, other observables like the average distance between neighboring points, density estimates, etc. may be interesting alternatives and the presented model can be used as a template for designing such models.

As a side-effect DCS allows to perceive the clusters in the dataset as "waves of higher amplitude", although this was not a design objective. An interesting aspect for future work would be to enhance the set of possible interactions and techniques to query information from the crystal.

8.6 *Growing Neural Gas Sonification*

The Growing Neural Gas Sonification (GNGS) Model is based on the adaptive incremental network model of Growing Neural Gas (GNG) Networks [Fri95]. It is able to learn important topological relations in a given distribution. The aim of the sonification model is to use the GNG as a means to query the data in an intuitive way, enabling the listener to pick up information about the structuring and topological features of the data by listening. While investigating the model, another profitable application of sonification became obvious: sonification can be used to monitor the adaptation of the neural gas growth, which provides information about the degree of approximation, the intrinsic data dimensionality and the model complexity. So GNGS may be a helpful tool to assess complexity properties of the GNG like its underfitting or overfitting.

Physical objects in our environment that produce interaction sounds (e.g. a cup or a guitar body) can be modeled by a network of masses, springs and dashpots. The interaction of these elements leads to characteristic vibratory patterns of motion which are perceived as sound. It seems that human auditory perception is highly trained to distinguish sounds that are generated by such a kind of process. The idea of the GNGS model is to use the neural network model to define an acoustic object resp. “virtual instrument” which can be excited by the user in the same way as real objects, and that mimics some of their physical properties, especially the propagation of energy waves in the object.

This sonification model utilizes the neurons as information nodes in model space. Local probing of the GNG injects energy to a probed neuron. Energy propagates through the GNG network along the edges to topological neighbor neurons and is partly dissipated at the neurons as sound radiation. The acoustic properties of a neuron are determined by local observables like the number of its edges, its approximation error or its average distance from the other neurons. Unconnected subgraphs of the GNG can not exchange energy and will thus show an independent acoustic behavior.

For interaction with the model, a scatter plot of the dataset and the interconnected GNG graph are displayed to the user. Sound invocation is done by selecting a neuron in the visual display with the mouse pointer. The sonification is rendered offline and played to the user afterwards. Typical sonifications take a few seconds so that the interaction loop is not interrupted for too long.

The GNGS model can be used to explore and compare the intrinsic dimensionality of data. Additionally it can be used to investigate the clustering of data. Process monitoring of GNG network growth enables to judge the complexity of the GNG model in terms of overfitting or underfitting.

8.6.1 Introduction to Growing Neural Gas Networks

In the setting of unsupervised learning, only input variables are available but there is no information about the desired output. In these situations one possible objective for learning is to reduce the data dimensionality. A standard technique for this is to project the data on a few principal directions which span a linear subspace so that a good approximation is obtained as discussed in Section 2.3. In the case of the data being distributed on a low-dimensional curved manifold in data space, however, linear approximations would require a higher dimensionality than the data intrinsically has. In such cases, nonlinear projection methods allow reduced approximation errors with manifolds of lower dimensionality. Principal Curves and Principal Surfaces are nonlinear manifolds which are able to reduce the data dimensionality in this sense [LT94]. The Self-Organizing Map (SOM) approximates such nonlinear mapping by using reference points (neurons) whose topological neighborhood is defined by the SOM structure [CM94]. A SOM or Principal Surface can give only poor approximations if the intrinsic dimensionality of the data is higher than that of the neuron grid. This leads to another interesting objective for unsupervised learning: finding out what topological properties a data distribution has.

The Growing Neural Gas was introduced as a means of learning topologies by Fritzke [Fri95]. The model builds on the “competitive Hebbian learning” (CHL) [Mar98] and the neural gas algorithm. CHL aims at finding a subgraph of the Delaunay triangulation, limited to those areas of input space which are covered by the distribution. Such a graph is called the “induced Delaunay triangulation” and is shown to optimally preserve topology [Mar98]. It is constructed for a given set of reference vectors or centers by connecting the closest two centers to each input $\mathbf{x}$ by an edge. Therefore a prerequisite for CHL is a suitable set of reference points which can for instance be obtained by vector quantization. As a special case, the neural-gas (NG) [MS91] can be taken, where basically for each input the coordinates of the K nearest centers are adapted with decreasing K during learning. Both the adaptation range K and the adaptation strength underly a decay schedule, for which the number of adaptation steps has to be defined in advance. An alternative to the successive application of the NG and the CHL algorithms for topology learning was the Growing Neural Gas (GNG) proposed by Fritzke. The GNG is an incremental network model where the edges as well as the neurons are modified during learning.

The GNG algorithm

The implemented GNG algorithm is the same as presented in Fritzke [Fri95]. It is briefly summarized here to introduce the variables as they are used for the sonification.

The GNG is given by a set of neurons or reference vectors $\mathbf{w}_i$ which are regarded as positions in input space and a list of undirected edges e_j among neuron pairs. Each neuron additionally possesses storage for an error accumulator R_i, and each edge holds a counter for its age A_j.

The algorithm starts with two units $\mathbf{w}_1, \mathbf{w}_2$ at any random position in input space.

1. Draw a data point $\mathbf{x}$ from the underlying distribution.
2. Find the nearest and second nearest neurons i_1, i_2.
3. Increment the age of all edges emanating from neuron i_1.
4. Update $R_{i_1} \leftarrow R_{i_1} + \|\mathbf{w}_{i_1} - \mathbf{x}\|$
5. Update neuron positions for neuron i_1 and its topological neighbors n by

$$\begin{aligned} \Delta \mathbf{w}_{i_1} &= \epsilon_b(\mathbf{x} - \mathbf{w}_{i_1}) \\ \Delta \mathbf{w}_n &= \epsilon_n(\mathbf{x} - \mathbf{w}_n) \end{aligned}$$

6. Create an edge j between neuron i_1 and i_2, if it does not already exist. Set its age $A_j = 0$.
7. Remove those edges with an age larger than a_{max}. Remove those neurons without edges.
8. Every λ steps
   - Insert a new neuron q half-way between the neuron $q_1 = \arg\max_i R_i$ and its topological neighbor neuron q_2 with the largest error. The edge between q_1 and q_2 is removed and edges from both neurons to the inserted neurons are added.
   - Update $R_{q_1} \leftarrow \alpha R_{q_1}, \;\; R_{q_2} \leftarrow \alpha R_{q_2}$ and set R_q = R_{q_1}.
9. Multiply all errors R_j with a constant $\rho < 1$.
10. Proceed with step 1 until a stopping criterion is fulfilled.

The constants $\epsilon_b, \epsilon_n, a_{max}, \lambda, \alpha, \rho$ determine the operation of the GNG algorithm. Their role within the algorithm is intuitively clear, see [Fri95] for a detailed discussion of the roles of these parameters. For the datasets used in this work it turned out that fixed values $\epsilon_b = 0.2$, $\epsilon_n = 0.006$, $\alpha = 0.5, a_{max} = 50$ and $\rho = 0.995$ work fine. λ was set to a constant fraction of the dataset size. As a stopping criterion, a final net size was used as indicated below.

The GNG algorithm was modified in two respects for its usage in sonification:

- Input signals are obtained by drawing a random sample from the dataset and adding Gaussian noise $\eta \sim \mathcal{N}\{0, \sigma^2 \boldsymbol{I}_d\}$. The purpose of this noise term is “smoothing” the network by preventing the net from copying the data samples. During the adaptation run σ^2 can be controlled manually.

- Besides the “Growing Mode” described in the algorithm, an “Optimization Mode” is introduced, where the neuron insertion (see step 8 above) is skipped. The mode can be chosen interactively. In optimization mode, excessive neurons and edges are deleted until a stationary state is reached.

The noise variance σ^2 is a resolution parameter controlling the complexity of the net. At the end of this chapter, an annealing sonification will be proposed which makes particular use of the variation of σ^2.

8.6.2 Model Description

The GNGS model is defined for datasets given by a data matrix $\mathbf{X} \in M(N \times d, \mathbb{R})$ whose row vectors $(\mathbf{x}_i^{\mathrm{T}}, \mathbf{y}_i^{\mathrm{T}})$ are the given records with $i = 1, \dots, N$. Currently, only the input part $\mathbf{x}_i \in \mathbb{R}^{d_{in}}$ is used. In the context of unsupervised learning an output part is missing anyway. The basic GNGS model can easily by extended to include auditory information about the output part. However, such extensions are not going to be focused here. In any further discussion $d = d_{in}$ will be used.

Setup

The GNGS model is a spatial model in an $\mathbb{R}^d$ model space. The GNG graph is given by the neuron weight vectors $\mathbf{w}_i$, which are interpreted as coordinates of the neuron objects in model space, and a set of edges which connect pairs of neurons. These connections are represented as straight mesh lines in the plots. In this model the edges are used as “conductors” which transport energy between neurons. A typical GNG graph is illustrated in Figure 8.31. It shows the projection of the setup in model space onto two selected axes.

For the GNG probing interaction which will be discussed first, the GNG configuration remains unchanged during exploration. Instead, each neuron i is given an energy variable E_i which is initially set to 0.

Dynamics

The dynamics describes the energy exchange between neurons which are connected by an edge. The energy vector $\mathbf{E}$ is updated by numerical integration of the energy flow equation

$$\frac{dE_i}{dt} = -gE(t) - \sum_{j \in I_N(i)} q(E_i(t) - E_j(t)) \tag{8.50}$$

where $I_N(i)$ is the set of indices of all neurons that are connected with neuron i. The first term of eq. (8.50) causes an exponential energy decay. This energy can be imagined to be radiated as sound. The second term causes energy diffusion through the network graph. With $g = 0$, the equilibrium state would be a uniform distribution of the total energy $E_{tot} = \sum_i E_i$ on all connected neurons. The speed of energy decay and energy transport can be controlled independently by g and q. Technically, the dynamics is implemented as a series of update steps iterating through all edges, where a fraction q_{step} of the energy difference between connected neurons is transported to the neuron with smaller energy.

Initial State and Excitation

The energy vector $\mathbf{E}$ is initialized to 0. The system thus is in a state of equilibrium and silent. Excitation is done by probing the GNG: selecting a neuron in the GNG display with the computer mouse increases its energy by a value 1. This interaction is analogue to striking (e.g. on a metal bar), where the kinetic energy of a system is punctually changed.

Model-Sound Linking

The sonification is the superposition of all neuron sounds. The neuron sound level resp. the sound amplitude is determined by its energy E_i. Currently, a linear sine oscillator is used for the neuron sound which has besides level frequency as the only parameter. In general, timbre and temporal structuring of the sound can be used to relate local attributes to the sound. A simple Data-Model Assignment will be presented below.

Listener

The listener is not located anywhere in relation to the model space. The sound contributions of the neurons are simply mixed to a mono sound vector.

Sound Synthesis

Additive Synthesis (see S. 6.2.1) is used for sound computation. For each neuron, a time-variant oscillator with frequency and amplitude control is used. Frequency and amplitude values are specified on equally spaced points in time and interpolated between these control points.

Data-Model Assignment

The GNG is characterized by the neuron positions $\mathbf{w}_i$, the list of edges, the edge age vector and the error variables R_i. These variables are possible candidates to determine acoustic properties. However, as the aim was to learn something about the dimensionality or topology of the network, a derived quantity is used: the number of edges $n_e(i)$ emanating from neuron i. This variable is used since it correlates with the intrinsic dimensionality of the data. Figure 8.29 shows the average number of edges per neuron for GNG networks with 100 neurons adapted to a uniform q-dimensional distribution. The average number of edges grows monotonously with subspace dimension.

$n_e(i)$ is used to determine the frequency of the time-variant oscillator that is used to generate the neuron sound, whereas the amplitude is determined by the energy as pointed out above. In the GNG probing, $n_e(i)$ remains constant and thus each neuron contributes a tone of fixed frequency. The frequency is mapped from n_e by

$$f(n_e) = f_0 2^{5n_e/12} \tag{8.51}$$

which causes a shift of 5 musical half-tones for each neighbor. The interval can be adjusted by the user so that all frequencies are within the audible range. A suitable value for f_0 is 150 Hz.

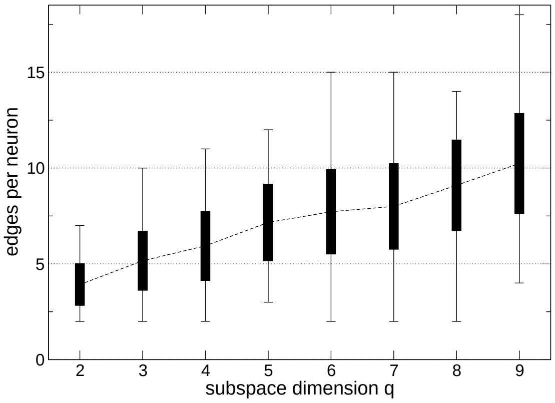
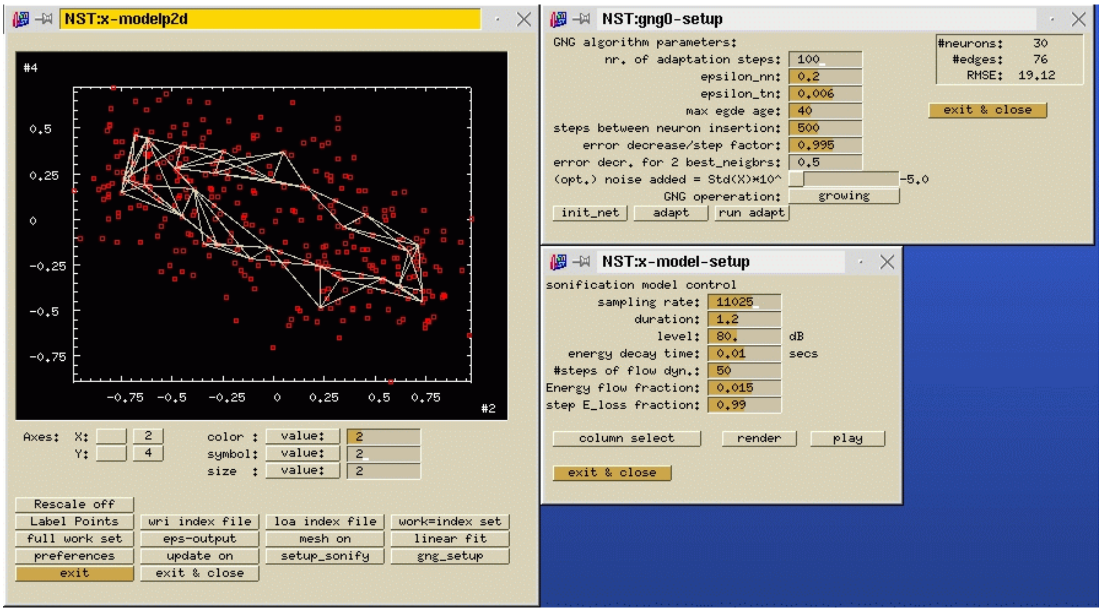


Figure 8.29: Number of edges per neuron for q-dimensional uniform distributed data (1000 records) in $[0, 1]^q$ using 100 neurons. The box limits indicate the standard deviation.

8.6.3 Implementation

Figure 8.30 shows the visual display for interacting with the sonification model. A number of parameters can be changed here like the sampling rate, the total duration, level, number of dynamic steps, the energy gain g and the flow speed q_{step}. Computation of the sonification is triggered by selecting a neuron in the scatter plot.



Axes: X: 2
Y: 4

color : value: 2
symbol: value: 2
size : value: 2

GNG algorithm parameters:
nr. of adaptation steps: 100
epsilon_nn: 0.2
epsilon_tn: 0.006
max egde age: 40
steps between neuron insertion: 500
error decrease/step factor: 0.995
error decr. for 2 best_neigbrs: 0.5
(opt.) noise added = Std(X)*10^ -5.0
GNG opereration: growing

#neurons: 30
#edges: 76
RMSE: 19.12

sonification model control
sampling rate: 11025
duration: 1.2
level: 80. dB
energy decay time: 0.01 secs
#steps of flow dyn.: 50
Energy flow fraction: 0.015
step E_loss fraction: 0.99

Figure 8.30: User Interface for controlling the GNGS model. The plot shows a 10d noisy circle dataset. The projection of the data obscures the ring topology which is obviously found by the GNG graph.

8.6.4 Examples

To introduce the GNG sonifications, synthetic datasets are used in order to have precise control of the intrinsic data dimensionality.

Mixture of Gaussians

For the first example, data were sampled from a mixture of 3 Gaussians with density

$$p(\mathbf{x}) = \frac{1}{3}\left(g_{0-1}(\mathbf{x}; 0, \sigma^2 \boldsymbol{I}_2) + g_{0-3}(\mathbf{x}; \hat{\mathbf{x}}_0, \sigma^2 \boldsymbol{I}_4) + g_{0-7}(\mathbf{x}; 2\hat{\mathbf{x}}_0, \sigma^2 \boldsymbol{I}_8)\right) \tag{8.52}$$

and $\sigma^2 = 0.2$. g_{a-b} denotes here the density of a normal distribution in the variables $\{x_a, x_{a+1}, \ldots, x_b\}$ which is 0 if any of the other variables is unequal to zero. The intrinsic dimension of the clusters is 2, resp. 4 or 8. Figure 8.31 shows a projection of the dataset on the (x_0, x_1)-axes and a GNG network with 50 neurons. Three sound examples (see Table 8.14) are rendered by clicking on different neurons close to the 3 cluster centers. The brilliance of the sound increases with intrinsic data dimensionality. The attack phase is determined by the selected neuron. At the end of each sonification the energy is distributed to all neurons that are in the same subgraph as the initial neuron. Energy propagation to topological neighbors causes the spectrum to broaden if these have a different value of n_e.

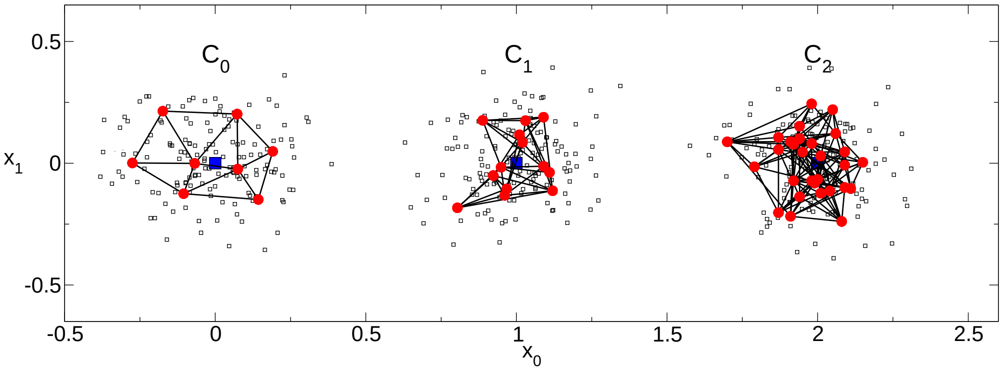


Figure 8.31: GNG for a mixture of 3 Gaussians with intrinsic dimensionality 2, 4, 8 from left to right. The dataset and the GNG graph are shown. The GNG was adapted in 50000 steps with parameters $\epsilon_b = 0.2$, $\epsilon_n = 0.006$, $a_{max} = 40$, $\rho = 0.995$, $\alpha = 0.5$, 1000 steps per neuron insertion. The shown GNG has 50 neurons and 170 edges.

File/Track 30:	Cluster C_0 (2d): a , b , c
	Cluster C_1 (4d): a , b , c
	Cluster C_2 (8d): a , b , c
Description:	GNGS for a mixture of 3 Gaussians in $\mathbb{R}^{10}$, plot shown in Figure 8.31
Duration:	1 s

Table 8.14: Sound examples for GNGS Probing.

Noisy Spiral Dataset

The following example demonstrates energy propagation for a curved one-dimensional data distribution. A 2d noisy spiral dataset is used, since it is suitable to demonstrate some dependencies of GNGS on network size while at the same time maintaining the ease of a simple visualization. Figure 8.32 shows scatter plots of the dataset and typical GNG graphs for different network sizes. In the sound examples (see Table 8.15) for the GNG network with 3 neurons (plot (a)), all neurons contribute the same pitch because all neurons have two neighbors. Such a small network is incapable of properly reflecting data dimensionality in its graph. The situation changes with increasing network size. In (b), the GNGS changes to mixtures of 2 or 3 tones of higher pitch. The graph indicates that most neurons have about 4 topological neighbors. The initial pitch depends on the number of edges at the selected neuron. Since the energy is distributed equally among all

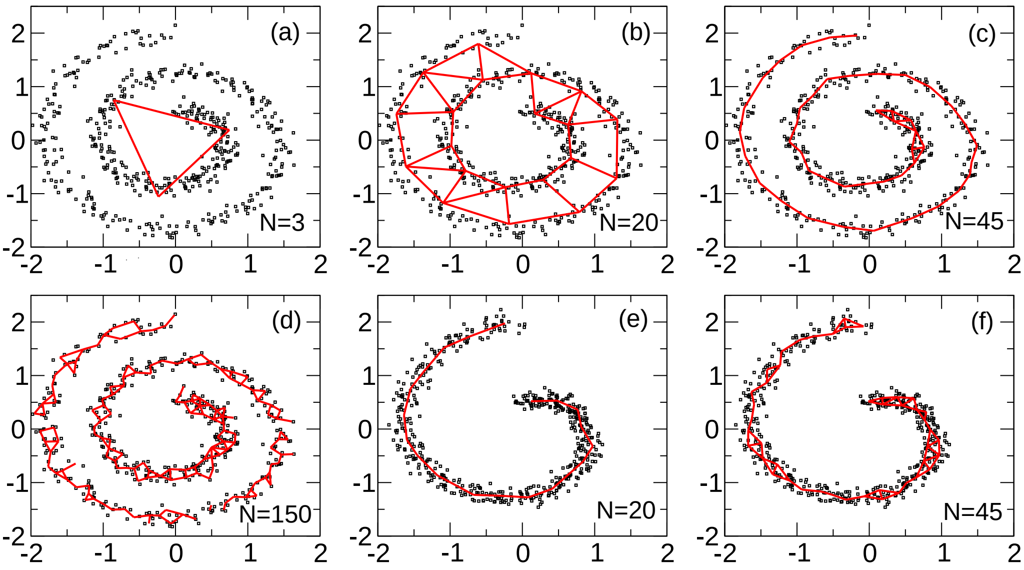


Figure 8.32: Noisy Spiral dataset and GNG graphs: (a) - (d) show the noisy spiral with 2 rotations. Obviously, with increasing GNG size, the GNG adapts to a suitable fit until the GNG is overfitting the data. (e) and (f) show a noisy spiral with only 1 rotation. Here the GNG begins to overfit with 20 neurons.

File/Track 31:	examples for Figure 8.32
	(a) GNG with 3 neurons 1 , 2
	(b) GNG with 20 neurons end , middle , inner end
	(c) GNG with 45 neurons outer end , middle , close to inner end , at inner end
	(d) GNG with 150 neurons outer end , in the middle , inner end
	(e) GNG with 20 neurons outer end , in the middle , inner end
	(f) GNG with 45 neurons outer end , in the middle , inner end
Description:	GNG probing sonification for 2d noisy spiral dataset
Duration:	1 s

Table 8.15: Sound examples for GNGS for the noisy spiral dataset shown in Figure 8.32.

neurons at the end of the sonification, the relative level at each pitch indicates how many neurons have the associate number of edges. Obviously, the GNG graph is a two-dimensional ring. The limited network size prevents the GNG from discovering the actual spiral structure. This is achieved at a network size of about 45 neurons, shown in (c). The GNGS sound examples now again show a simpler sound that consisting mainly of one pitch, since the number of 2-edged neurons dominates. However, at the inner end of the spiral, the GNG starts to adapt to the noise which becomes audible by more complex sound when the GNG is probed there. If the adaptation is continued, the neurons converge more and more to the samples. The GNG graph can collapse into subgraphs. Some GNGS examples are given for the GNG graph shown in plot (d).

Depending on the data at hand, a different GNG network size is appropriate to describe the structure. Plot (e) and (f) in Figure 8.32 show a noisy spiral with one rotation. Here, 20 neurons suffice to get a suitable graph and overfitting begins with 45 neurons. It is up to the researcher to evaluate the quality of a fit so that exploratory tools are here required.

8.6.5 Monitoring of GNG Growth

In this Section, the GNGS model is extended to monitor the growth process of the GNG network. This has been inspired by the experience that the network graph depends strongly on the network size, and the fact that auditory perception is much better in *comparing* GNG probing sonifications than in drawing direct conclusions from the sound to data dimensionality.

With this model extension, the aim is to summarize the growth process by using sound. It is of particular interest if changes in the monitoring sonification correlate with the model beginning to overfit the data. This is regarded as important as there is no canonical criterion when to stop the GNG growth. In the toy examples of the last section, overfitting was easily detected from the graphs. In high-dimensional settings this can be more difficult. In such situations, GNG process sonifications may provide valuable assistance. In addition there are many events like neuron insertion, neuron deletion, edge creations that are usually not monitored while using GNG models. These may by integrated into sonification by creating further auditory streams to keep the user in touch with the progress of the adaptation.

In contrast to the GNGS model presented in the last section, here acoustic energy is always distributed uniformly among all neurons and remains constant over time. The GNG process sonification is rendered by performing a sequence of adaptation/sonification cycles. N_a adaptation steps are performed in a single adaptation cycle. In the following examples, $N_a = 500$ is used and neurons are inserted every 300 adaptation steps. In a sonification cycle, the sound is computed by superimposing an auditory grain for each neuron. The auditory grain frequency is associated to the number of edges per neuron n_e exactly as in the previous approach. The auditory grains are further characterized by their amplitude envelope which is here chosen as a triangular window function. The density of the auditory grains in time is determined by the number of cycles and the total duration of the sonification.

Sound Examples

Sonifications are presented for the four datasets shown in Figure 8.33. The sonifications are rendered with 80 cycles in 5 s sonification time. A sampling rate of 44100 Hz is used and the auditory grains last 200 ms. Parameters of the GNG model are kept as described in the caption of Figure 8.31.

The first example is a process sonification for a GNG growth in a dataset given by the noisy

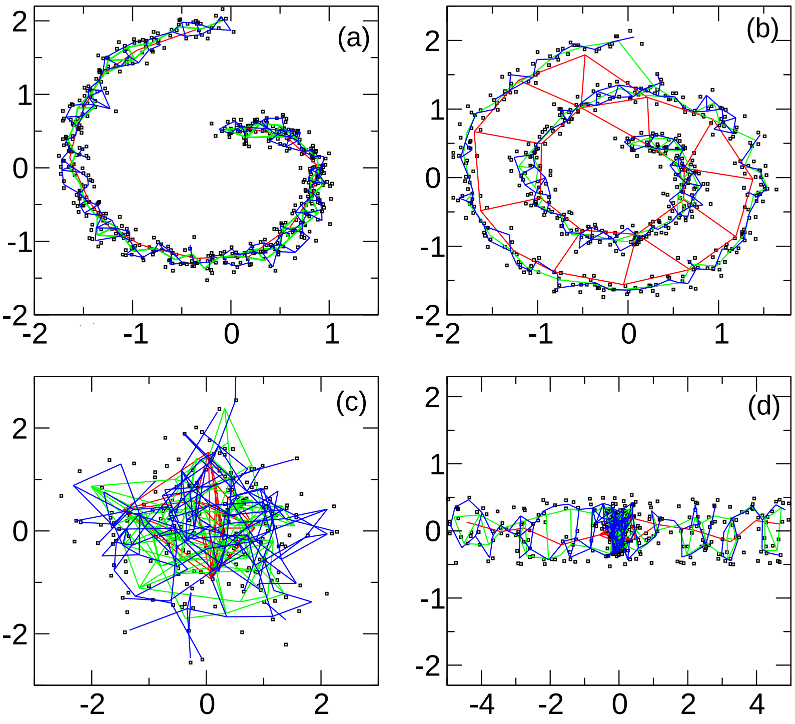


Figure 8.33: Scatter plots of example datasets and GNG networks at 3 stages of the growth process. The GNG graphs correspond to network size 20 (red), 50 (green) and 135 neurons(blue). (a) noisy spiral with one rotation, (b) noisy spiral with two rotations, (c) 5d Gaussian (d) mixture of a 2d rectangular uniform distribution and a 5d Gaussian.

spiral shown in Figure 8.33, (a). The sonification (see Table 8.16) begins with low pitched spectrum and thus few edges per neuron. With increasing network size a stable pitch structure is audible for some time, consisting of a low pitched tone. During this time the GNG structure is a one-dimensional chain with two edges per neuron. With increasing network size the GNG adapts to the noise. At this point the number of edges increases and as a result the sound brilliance rises. In the end the level in the lowest tone increases again. This is caused by deletion of edges due to overfitting. Some neurons are then only activated by single data points. Such an increase of loudness at the lowest tone generally indicates overfitting and may be used as an auditory feature to value model complexity.

File/Track 32:	Noisy spiral with 1 rotation: sound Noisy spiral with 2 rotations: sound Gaussian in 5d: sound Mixture of 2d rectangular uniform distribution and 5d Gaussian: sound
Description:	GNG Process Monitoring Sonification examples
Duration:	5 s

Table 8.16: Sound examples for GNG Process Monitoring Sonifications for different data distributions whose plots are shown in Figure 8.33.

The next example is a sonification for the noisy spiral dataset shown in Figure 8.33, (b). A remarkable difference can be perceived during the first half second of the sound. The spectral mean grows rapidly, indicating that at the beginning, the net is two-dimensional to approximate the data. With increasing network size, the GNG adapts to the finer one-dimensional curve and edges between adjacent braces of the spiral are deleted. This causes a decrease of the spectral centroid. Finally, as in the last example, overfitting begins. The dynamical evolution of spectrum is typical for situations where a higher-dimensional data distribution is build up from lower dimensional pieces.

Very different from the latter sound is the sonification of GNG growth for a 5d Gaussian, shown in Figure 8.33, (c). The number of edges per neuron grows much faster than in the previous two-dimensional examples. However, it can be heard, that directly after the beginning of the sonification, neurons with few edges are very rare. This is typical for data that is distributed rather homogeneously in a high-dimensional space. As with the other examples, edges are getting deleted due to overfitting which leads to a growth of acoustic energy in the low-frequency part at the end.

The final sound example for the dataset in Figure 8.33, (d) sounds similar to the previous one. In contrast to that, however, the sound level at the lower modes does not dissappear in the beginning. Carefully comparing both examples, it is possible to follow the lower pitched plateau. This part of the spectrum is caused by the neurons in the 2d distribution.

8.6.6 Conclusion

The GNGS model provides a technique for browsing a high-dimensional dataset in an intuitive way by probing a scatter plot with the mouse pointer. The neural network in this case is helpful in two ways: as a means to approximate the given data by a model of reduced complexity, and by providing a process which helps the user to connect sound with its meaning. The current implementation uses only the number of edges per neuron to determine the sound, so that many other observables like density estimates, coordinates, edge ages etc. remain unused. Such variables may be integrated in future extensions of the GNGS model.

The extension of GNG Process Sonification showed that adaptation processes can be sonified by using iterated adaptation and sonification cycles using a modified probing model. Using GNGS, the evolution of network connectivity becomes audible as temporal evolution of spectrum. This enables the listener to perceive dynamical patterns (e.g. timbre plateaus) which may help to understand or explain the data at hand.

While the presented GNG Process Sonification monitored the acoustic properties of the GNG during its growth, another possibility is to monitor its changes during the variation of a resolution parameter. In the current implementation, inputs are generated by sampling from the dataset and adding some noise $\eta \sim \mathcal{N}\{0, \sigma^2 \boldsymbol{I}_d\}$. The parameter σ controls the resolution and may be referred to as a temperature parameter. An interesting sonification might be to monitor the sound changes during a decay of σ with constant network size. Such *Annealing sonifications* will be considered in ongoing work

In the GNGs model, the element of energy propagation along edges is an important new building block that may be used also in other sonifications as for instance for rendering sonifications for Self-Organizing Maps.

8.7 Discussion

The presented sonification models are examples for the framework of Model-Based Sonification. Each model focuses on a different aspect of the data and of human listening capabilities. All sonifications use superposition of independent sound pieces to compose a complex sound. This potentially makes these algorithms suitable for parallelization.

The mathematical details of the models may give the impression that it is difficult to guide an untrained user of the sonification models. However, for the user it is only important to understand the qualitative behavior of the model. This “model in mind” assists the interpretation of the sound and mathematical details are unnecessary for using a model, same as we do not need to understand acoustics to use sound clues in our environment.

The presented models have in common that in principle the same information can also be given by visual means. For instance in the PCS model (Section 8.4), the distance of data points from the principal curve may be plotted in bar plots over the projection index as proposed by Meinicke [Mei00]. The Particle Trajectory Sonification Model, however, is an example of a sonification model where no visualization is found that imparts the whole information. Although pitch clusters are also observed in the spectrograms, the shape of the potential function is in a way “holistically” encoded into the sound. To the author such sonifications seem to be most promising as they offer a chance to detect structures that are difficult to visualize.

All models presented use a browsing or triggering interaction. This is due to the limited expressiveness of the available input devices (mouse, keyboard) and due to the fact, that the sonifications are currently rendered offline. Interactive real-time sonification models are seen to offer a broad field of possibilities to experience dynamical information and will be focused in ongoing research.

All sonification models have in common that they provide additional information about the data without disrupting the view on the presented visualization. However, so far visualization and sonification have barely been coupled. Using a dynamic (animated) visual display and a time-synchronous sonification can potentially improve the correlation from sonic elements with elements in the visualization. If for example each data point would flash in the scatter plot during the playback of its respective sound element or a marker would move along the curve in the PCS model during playback of the sonification, this would help the user enormously in relating the sound to the data.

Finally, the models introduced in this thesis represent the first basic demonstrations for the framework of Model-Based Sonification. The author hopes that after some evolution appropriate models for many analysis tasks will exist and that sonification models will be found that enable researchers to improve the detection of interesting patterns in datasets.

Chapter 9

Extensions

In this chapter two extensions to Parameter Mapping Sonification will be presented which do not fit into the previous chapters. The first section will present how additional auditory clues can be integrated into Sonic Scatter Plots rendered by Parameter Mapping Sonification that explain the sound by using sound and spoken text. The second section presents *Multidimensional Perceptual Scaling*, a Parameter Mapping Sonification that avoids the burden of extensive parameter tuning by using MDS and PCA (see Section 2.3) as a plug-in to process the data prior to sound rendering.

9.1 Elements of Sonic Scatter Plots

In Section 2.2.1, the basic elements of graphing data in scatter plots and their function have been discussed. In this section, methods will be proposed to integrate analogous auditory counterparts to these elements into Sonic Scatter Plots. Furthermore, sound examples will be given.

Box – Acoustic Frame: The frame box in a plot limits the display area. In sonification, the intrinsic dimension is 1, the dimension of time. Therefore an analogous limiting can be achieved by adding two auditory marker sounds that mark the beginning and the end of the sonification. Decaying sine pulses at 500 Hz and 200 ms duration are well suited. In order to avoid temporal masking, the level should be about -20 dB compared to the peak level of the sonification or less. A problem that may arise is that superposition of the initial marker with the sonification may impair perception of acoustic features during the beginning of the sonification. The following solution is proposed: two markers are played at $t_0 = 0$ and $t_1 = t_b$ and data sonification starts at time $2t_b$. Suited values are $t_b = 300$ ms. From the sequence of two tones the listener expects the onset of the sonification. This marks the begin of the sonification without masking it. An alternative way of supplying a frame is to use spoken words like “go” and “stop” as markers.

Tick Marks: Acoustic tick marks can be added to the sonification by using short percussive ’tick’ sounds at equally spaced points in time between begin and end. Sampled “claves” sounds are well suited. Their level should be adjusted so that they can just be heard but do not disturb. They are perceived as a rhythmic background stream and allow to orient the events relative to a given time grid.

Legend and Value Range: Each acoustic attribute has a specified value range. Both the assignment from data variables to attributes and the range can be presented in an auditory legend composed as follows: in a sequence, the spoken variable label is followed by three tones which only vary in the applied attribute and have mean values for other attributes. The tones display the sound for minimal, mean and maximal attribute value.

Symbols – Acoustic Events: In a scatter plot, each data point is represented by a symbol. The acoustic analogue is an sonic event. Their acoustic structure can be determined by several attributes as discussed in Section 3.7.4.

Lines - Dynamic Sound Variables: to represent continuous variations in sound, dynamic sound attributes have to be used. These are attributes which can be modified in continuously playing sound. For example frequency is a dynamic variable, while attack time is not. The sound is played during the whole sonification while one (or more) of his attributes are continuously varied according to the associated variable(s). Suitable attributes are frequency, amplitude, frequency modulation index, brightness, roughness or filter frequencies.

Annotations: Symbolic labels can be integrated into a sonic scatter plot by using speech or event markers. Speech markers directly "say" the label which is annotated to a sound and they are played simultaneously to the event or after the sonic event has finished.

Features: As an analogue to elements in graphical plots like trend lines or markers for min/max values of a function plot, in sonic scatter plots auditory marker events can be introduced. To stand out of the sonification, it is recommended to use another timbre or a totally different instrument. For instance samples from real percussive instruments or speech sounds would be suitable marker sounds in a sonic scatter that uses flute tones to represent data points.

In visual plots, the eyes may selectively attend the legend, the tick marks or the shape of the plotted functions. In auditory display, annotation markers and tick sounds may disturb the listener depending on the task at hand. Therefore an interface is helpful that allows the user to mute these additional sounds or at least to control their level. Playback of the auditory legend and value range sound may be suppressed and only played at the users request.

Sound examples for such enhanced Parameter Mapping Sonifications are compiled in Table 9.1.

File/Track 33:	Iris dataset:	raw sonification ,	with frame markers ,	Auditory Legend
	Cancer dataset:	raw sonification ,	with frame markers ,	Auditory Legend
Description:	Auditory Legend, Frame and Tick marks			

Table 9.1: Sound examples for Auditory Legends and Sonic Scatter Plots.

9.2 Perceptual Parameter Mapping

One of the main problems with Parameter Mapping Sonifications is the necessity to specify a mapping. Additionally, for interpreting the sound, one has to refer back to the mapping to associate sound attributes to data attributes. In this section, such mapping problems are avoided by combining Parameter Mapping Sonification with a data preprocessing that automatically transforms data records to suitable attribute vectors for sonification. Instrument design and parameter range optimization thus only need to be done once in this approach, and then the sonification operates on arbitrary datasets without any further tuning of parameters. Since the preprocessing maps a high-dimensional dataset on perceptual attributes, it shall be called *Multidimensional*

Perceptual Scaling (MPS). Some requirements are that the MPS shall be invariant on rotations of the dataset (resp. any orthogonal transformation), on translations of the mean and that the sound should help to distinguish groupings in the data.

The first step for the definition of an MPS Auditory Display is to select an instrument, resp. a sound event generator with parameters $\theta = (\theta_1, \theta_2, \dots, \theta_q)$. For the examples in this section, an hybrid FM-synthesis/Additive-Synthesis model is used, given by

$$\begin{aligned} s^L(t) &= s(t-t_o)\sqrt{1-\alpha} && \text{(left channel)} \\ s^R(t) &= s(t-t_o)\sqrt{\alpha} && \text{(right channel) with} \\ s(t) &= A \cdot (w(t,t_a,t_d)\sin(\omega_c t + I\sin(m\omega_c t)) + Bw(t,0,t_d)\sin(3\omega_c t)\exp(-kt)) \end{aligned}$$

with a triangle amplitude envelope

$$w(t,t_a,t_d) = \begin{cases} 0 < t \le t_a & : \ t/t_a \\ t_a < t < t_a + t_d & : \ (t_d + t_a - t)/t_d \\ \text{else} & : \ 0 \end{cases} \tag{9.1}$$

The parameter vector is $\theta = (t_o, A, t_a, t_d, \omega_c, I, B, \alpha)$. t_o is the onset of the event in the sonification, A the amplitude, t_a the attack time, t_d the decay time, ω_c the carrier frequency of FM-synthesis, I the modulation index, B the relative strength of the attack part and α is used for intensity panning, so that the sound can be assigned between the left ($\alpha = 0$) and right ($\alpha = 1$) audio channel.

The next step is to select nonlinear mappings so that a linear change of an parameter is mapped to a (more or less) linear perceptual change. For instance, instead of controlling the amplitude A the level L is controlled and A is computed by $A = 10^{L/10}$, also pitch P is used instead of frequency $\omega_c = 2\pi 2^P$.

The next step is to select a mean parameter vector and appropriate ranges along the attribute scales, so that attributes changes can be discerned whatever the other attribute values might be. This requires a subjective optimization. My choices are

Variable:	L	t_a	t_d	P	I	B	α
Mean:	28	0.05	0.25	8.3	10	1	0.5
Range:	12	0.05	0.2	0.3	10	1	0.5

so that attributes take the values in the interval $[\text{Mean} - \text{Range}, \text{Mean} + \text{Range}]$.

Now the parameters are ordered according to their perceptual salience. Again this is a subjective ordering. Here my choice is $(t_o, P, L, \alpha, I, t_d, t_a, B)$.

For MPS now Multidimensional Scaling (see 2.3.4)is used to compute a one-dimensional representation of the dataset. The MDS index is mapped to the time index. PCA is applied to the dataset and the projection indices are mapped to the other elements in the parameter vector in such a way that the projection index on the first principal axis maps to the second element of the parameter vector and so on.

File/Track 34:	FM instrument:	Auditory Legend
	Iris Dataset:	MPS (PCA) , MPS (MDS/PCA)
	Cancer dataset:	MPS (PCA) , MPS (MDS/PCA)
	7d tetrahedron:	MPS (PCA) , MPS (MDS/PCA)
	10d noisy circle	MPS (PCA) , MPS (MDS/PCA)
Description:	MPS Sonifications using MDS and PCA	

Table 9.2: Sound examples for MPS Sonifications

Sound Examples for MPS are compiled in Table 9.2. A much simpler alternative is to omit the MDS step and just use the PCA transformed dataset to map the attributes. Then the projection index on the first eigenvector determines the onset of the sound events. The examples show that MPS sonification allows to perceive groupings in the data and even the structure of the circle dataset can be understood from the sound. The Iris dataset shows, that the MDS delivered projection indices related to the projection on the first principal axis of the dataset. The tetrahedron dataset shows that MPS allows to resolve all clusters. The sound of the events within a cluster are very similar in terms of their channel, brightness and level.

The advantage of assigning an independent projection index for the event onset is plausible since the organization of sound in time is the most dominating attribute. However, alternative to MDS, other one-dimensional projections may be used, like the 1d SOM or the principal curve. In contrast to PCS, here no information about the curve is given and no interpretation of the sound as motion along a curve is possible.

Chapter 10

Applications

In this chapter, some applications of sonification for data mining, pattern recognition and process monitoring will be presented in order to demonstrate the practical use of sonification. Application of sonification to concrete data mining problems is necessary for two reasons: firstly only real-world applications can show that sonification is not just a nice idea, but also a technique that can really improve or facilitate data exploration. Secondly, by the exchange and cooperation with domain experts and users which are (at the beginning) not familiar with sonification, there is much to learn about the demands on auditory display, the needs of the users and the way in which listeners use the sound.

During the last three years many different applications were considered by the author and not all of them are reported in this thesis. For the application of sonification on option trading the reader is kindly referred to the Diploma thesis of T. Thomas [Tho01] whom the author advised. Audification was applied to monitor the dynamical behavior of recurrent neural networks in the Diploma thesis of R. Haschke [Has99][1].

In the following Sections, three applications of sonification will be presented: (i) Monitoring of transcripts from psychotherapeutic protocols, (ii) Analysis of EEG data, and (iii) Sonification for exploration of cell biological images.

10.1 Sonification of Psychotherapeutic Protocols

The goal of this application is to develop a sonification system to assist the identification of key moments in transcripts from psychoanalytic sessions. The term *key moment* refers to segments of a session that are regarded as clinically important. These moments are seen as turning points or a breakthrough in the course of a psychoanalytic therapy.

This application has been developed in cooperation with E. Mergenthaler from the University of Ulm. In the cooperation the constraint was given that only verbatim transcripts of sessions were available as data. This means that relevant acoustic information about the patient's state (like prosody or speech velocity) and visual information (like mimics, gestures, body postures) or any other physiological observables like sweating or palpitation can not be used. However, the developed auditory display allows a later integration of such observables if they should become available.

E. Mergenthaler [Mer96, MB99] developed a model for the explanation of the patient's state from verbatim protocols and a theory for the temporal evolution of the patient's state in the course

[1] A sound example is available at `http://www.TechFak.Uni-Bielefeld.DE/ags/ni/projects/theory/oscinet/frequenz.html`.

of a session or treatment which is called the therapeutic cycle model. Mergenthaler defined a set of variables which can be computed from the transcripts. Two important variables are *emotion* and *abstraction*. Emotion is considered a central aspect of psychotherapies. Due to the constraints, the study is restricted to the indicators of emotion that can be observed in transcripts. Measuring the density of emotional words provides the variable of emotional tone. Technically this is done by comparing the transcript word-wise with a lexicon of emotional words. Emotion is regarded as essential for the key moments. However, in therapeutic sessions there are many situations with highly emotional tone but which lack insight. For that purpose the variable of abstraction has been added. Abstraction in verbal utterances can be seen as the central mechanism for the construction of new structures. Technically, abstraction is measured from text by using both a lexicon of abstract nouns (e.g. ’freedom’, ’friendship’) or by a suffix analysis, because abstract nouns show specific endings like ’-ness’, ’-ity’.

For further data analysis, the dataset is grouped in blocks of N words and the number of emotional words e_i and abstract words a_i in the i-th block is counted. Typical values are $N = 150$. Transforming the series $\{e_i\}$ (resp. $\{a_i\}$) to mean 0 and variance 1 yields for each i a two-dimensional vector (e'_i, a'_i). Mergenthaler now identified four prototypical patterns:

(-,-) Relaxing: (little emotion, little abstraction). In the relaxing state, patients talk about material that is not manifestly connected to their central symptoms. They talk in a more descriptive rather than reflective manner.

(-,+) Reflecting: (little emotion, much abstraction). In the reflecting state, patients present abstract topics without intervening emotions. This may be seen as intellectualizing and be interpreted as a defense strategy.

(+,-) Experiencing: (much emotion, little abstraction). In the experiencing state, patients may bring up conflictual themes and experience them emotionally.

(+,+) Connecting: (much emotion, much abstraction). In the connecting state, patients have access to conflictual themes and are able to reflect them. This may mark the clinically important key moment.

The therapeutic cycle is thus as shown in Figure 10.1. In the course of the session the state changes from relaxing, experiencing, connecting over reflecting to relaxing.

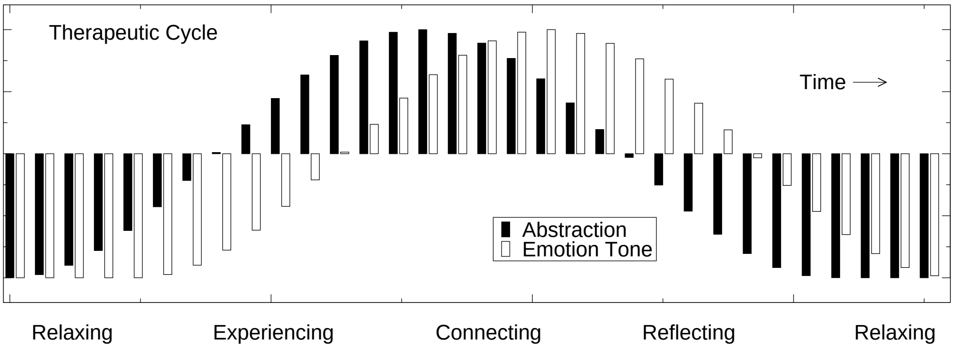


Figure 10.1: Prototypical cycle of emotion-abstraction patterns within the course of a psychotherapy session. After Mergenthaler [Mer96].

However, further analysis undertaken by Mergenthaler indicates that additional variables may also be used: the referential activity (RA) which is defined as the function of connecting non-verbal experience with language. Here a computer scored RA (CRA) variable is given. The word length is also considered as an attribute that might be related to the state of the patient. Furthermore, Mergenthaler separated emotional tone into two variables, one indicating emotionally positive and another for emotionally negative tones.

10.1.1 Data

There are datasets available for single therapeutic sessions. A typical session lasts about 1 hour and a dataset contains about 8500 words. Each word is characterized by the following attributes:

Attribute	Description	Values
hour	index of the session	$\mathbb{N}$
minute	time [minutes] since begin of session	$\mathbb{N}^+$
wordnr	number of the word within the session	$\mathbb{N}^+$
abstract	flag if word shows abstraction	$\{0, 1\}$
emopos	flag if word shows positive emotion	$\{0, 1\}$
emoneg	flag if word shows negative emotion	$\{0, 1\}$
refacth	flag if word shows high referential activity	$\{0, 1\}$
refactl	flag if word shows low referential activity	$\{0, 1\}$
speaker	identify speaker	1=patient, 2=therapist
wordlen	length of word in characters	Integer

10.1.2 Sonifications

The applications of sonification in this domain are twofold:

Monitoring: Sonification can be used to monitor the sessions of a patient in order to facilitate the selection of a session for further analysis.

Analysis: Sonification can be used to explore the dynamical evolution of a therapeutic session. The motivation is to uncover undetected patterns in the dynamical evolution of the session.

Three sonification approaches will be presented in this section. They may all be used for monitoring as well as the exploratory analysis of single therapeutic sessions.

10.1.2.1 Monitoring Transcript Sonification

The first approach is a word-by-word acoustic presentation of the data. As a first step, a time stamp is estimated for all words of a dataset by assuming an equal duration for all words spoken within one minute. The dataset is sonified by superimposing sound events for each word and storing the result in a sound vector. If none of the markers (abstract, emotion, CRA+, CRA-) are set, no sound is added for this word. The marker sounds for a word are spatialized according to the speaker, presenting patient markers on the left and therapist markers on the right audio channel in the stereophonic setup. A session of about 1 hour is temporally compressed to about 10 s. This high compression allows to compare different sessions by listening to the sonifications in a sequence. This first approach for data sonification allows to perceive the total frequency of emotional, abstract and referential words, the turn-taking (change of speakers) and also the speed of the dialog.

Sonification Design The selection of the marker sounds as well as the mapping are a subjective choice. Some reasons for the choices made will be given. All marker sounds are chosen to be percussive sounds with a short duration of about 250 ms. This reduces the overlapping and mutual masking which could reduce the temporal resolution. To facilitate timbre recognition, only one synthesized sound is used (a decaying sine pulse) and for all other markers sampled percussive musical instruments are used:

- Abstract words are represented by a ’ting’ sound. A sampled cymbal sound (sound example cymbal , Track 35) is used.
- Emotional words are represented by an exponentially decaying sine pulse whose pitch corresponds to the polarity of emotional tone: 523 Hz (C9) for positive emotions and 261.6 Hz (C8) for negative emotions.
- Words that indicate referential activity are marked by a sampled percussive claves sound (sound example claves , Track 35). This sound can be played simultaneously with the emotional and abstract sound without deteriorating their perception. The pitch of the clave sound is manipulated by resampling the sound vector with a resampling factor R. Lower/slower clave sound thus corresponds to low referential activity. $R = 1$ is used for words indicating high RA. $R = 0.5$ is used for words indicating low RA.

All three sound markers are positioned in stereo listening space according to the speaker: words from the patient (therapist) are presented on the left (right) audio channel.

Table 10.1 lists sonifications for four sessions using this sonification approach. The first listening impression is that all sessions are very much alike. However, after some practice, distinctions between the sessions become audible. In Session 12, at the end a phase with relative few emotional or abstract words can be perceived. That is right after the therapist has a higher activity. In Section 24, the therapist becomes more active in the middle of the session using more abstract words.

File/Track 36:	Session 11 , Session 12 , Session 24 , Session 26
Description:	Monitoring session transcript
Duration:	15 s

Table 10.1: Sound examples for Session Transcript Monitor Sonification

10.1.2.2 Therapeutic Cycle Sonification with AIBs

The analysis of Mergenthaler showed that the deviation of emotional tone and abstract tone from their mean value throughout the whole session is suitable for explaining the patient’s state. These deviations are rather small compared to the mean and therefore are overheard in the monitoring sonification presented above. The following sonification aims at using the deviation more explicitly to determine the sound. Another critical point in the approach above is, that the large number of CRA words dominated the auditory display. This problem worsens with increasing time compression. It can be avoided by applying Auditory Information Buckets to reduce the number of markers (for AIBs see Section 17).

The following sonification uses 5 AIBs to reduce the number of acoustic elements in each stream. An AIB is used for each of the attributes (abstraction, positive emotion, negative emotion, referential activity high and low). Records are put into an AIB if they fulfill the AIB condition, that is the case here if the respective attribute is set. So, one record may contribute to several AIBs. The AIB threshold is computed from the statistics of the session in such a way that each bucket flushes a constant number of times N during the session. Thus the AIB events mark the $1/N$-quantiles of the integrated counts. The AIB sonification is the superposition of marker sounds identical to the attribute sounds introduced in Section 10.1.2.1, but now pitch and amplitude are determined by local averages for the attributes. They are computed by kernel regression on the independent variable 'word number' using a Gaussian kernel with bandwidth σ_k.

Sonification Design The sonification consists of three auditory streams: the abstract, emotional and CRA stream. The streams are separated perceptually by using different timbres. The abstract stream uses the sampled cymbal sound, the emotional stream uses bitonal chords from a sine pulse generator and the CRA stream uses a sampled claves sound as in the Monitoring Transcript Sonification above. A marker sound event is played whenever the respective AIB threshold is exceeded. Deviation of the smoothed attribute from the mean value is used to compute the amplitude of the marker sound by

$$10 \log_{10} \text{amplitude} = \text{map}(\text{attribute} - \text{mean}), [0, \sqrt{\text{Var}(\text{attribute})}], [0, 20]) \ . \qquad (10.1)$$

The pitch is modified by the sign of the deviation using

$$\log_2(\text{frequency}) = \text{map}(\text{sgn}(\text{attribute} - \text{mean}), [-1, 1], [P_1, P_2]). \qquad (10.2)$$

where the pitch values P_1 and P_2 are chosen so that sounds are obtained that are easy to discriminate but similar enough to be recognized as belonging to the same class of sounds.

In phases of high density of emotional words, the AIB sonification for emotional words occurs at a higher rate and the pitch is higher to express a positive deviation from the mean value of the attribute. Again, all acoustic elements are spatialized between left and right channel depending on the speakers. Since the AIBs include words of the patient as well as the therapist, the stereo position is a continuous variable. The higher the fraction of words spoken by the patient, the closer the sound is located to the left audio channel and vice versa.

Figure 10.2 shows a smoothed dataset using kernel regression on the word number with a bandwidth of $\sigma_k = 300$.

Two different time compressions are useful for the sonifications:

- With $T_{\text{tot}} \approx 5$ s, different states of the session are easily discerned and a quick overview of the session can be obtained.
- Using $T_{\text{tot}} \approx 15$ s, more details of the session become audible. A good heuristics is to take about 8 buckets per second and a bandwidth of 300 words or more. This means to smooth attributes within a window of about 2 minutes of real-time in the session.

File/Track 37:	Session 12 (10 s) , Session 24 (10 s) , session 12 (5 s) , session 24 (5 s)
Description:	AIB sonification of session transcripts, sessions 12, 24
Duration:	10 s, resp. 5 s

Table 10.2: Sound examples for Cycle Sonifications using AIB streams for the attributes abstraction, emotion and CRA.

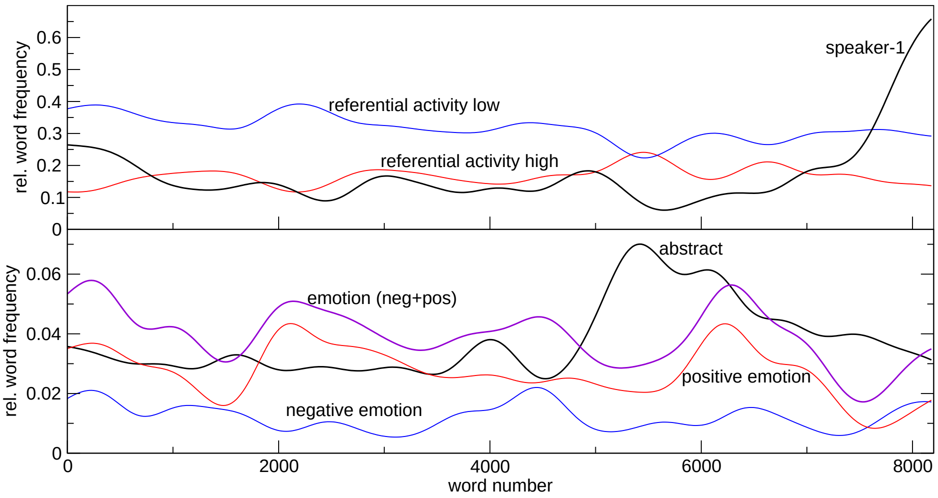


Figure 10.2: Plot of a session dataset using kernel regression to smooth the variables.

10.1.2.3 Symbolic Data Display

In the following sonification, occurrences of prototype patterns of the therapeutic cycle model are identified. This is done by considering the time-variant state of the session (e'_i, a'_i) as a 2d vector. The prototypes are located on a circle as shown in Figure 10.3.

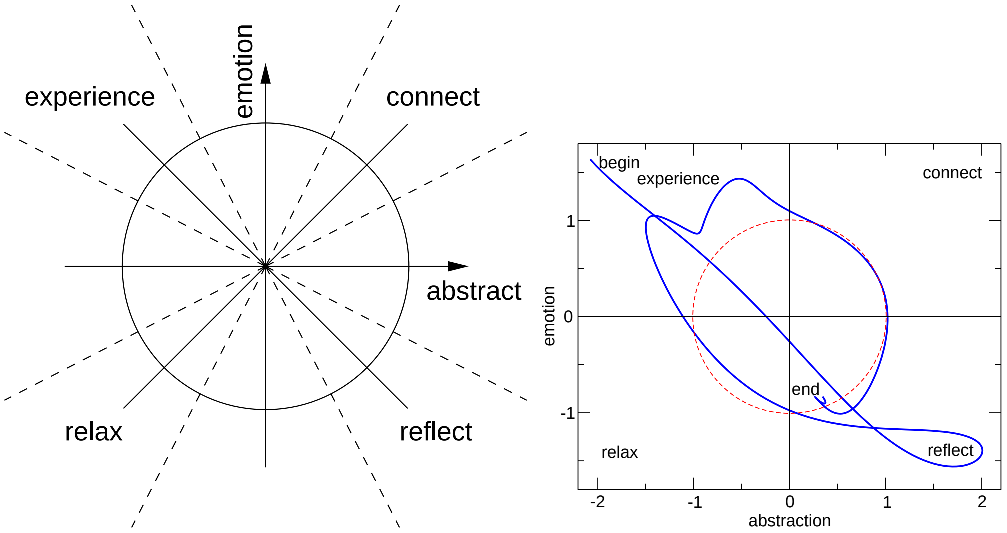


Figure 10.3: Prototypical Patterns in abstract/emotional feature space. The mean value is shifted to the origin and axis are in units or the standard deviations. The right plot shows the state trajectory for Session 24.

The sonification basically uses the same auditory streams as in the previous display. As an extension, the four spoken words ’connect’, ’relax’, ’abstract’ and ’reflect’ are now used to represent the state of the session. The spatialization of the speech marker is done in the same manner as the other streams. The sound level of the spoken markers now correspond to “the clearness” of the state: the closer the state is to the bisectors of the coordinate system, and the higher the distance is to the origin, the louder the spoken word is. These sonifications roughly allow to follow

File/Track 38:	Session 11 , Session 12 , Session 24 , Session 26
Description:	AIB sonification of session transcripts using speech markers.
Duration:	10 s

Table 10.3: Sound examples for Cycle Sonifications using AIB streams and spoken markers.

the development of a session and they also make it possible to connect sonic evolutions with a symbolic meaning.

10.1.3 Discussion

In the given datasets, time is an inherent attribute. As all the sonification models in Chapter 8 apply a different meaning to the time axis, and as in this case the aim is to gain insight into the dynamical evolution of the process, no sonification model is used here, but instead a new sonification method is developed in order to render the sound.

Firstly, a direct sonification of transcript datasets was presented. These sonifications failed to provide the relevant information according to the Therapeutic Cycle model due to the too large density of events. This problem was solved in the second approach by using Auditory Information Buckets, which further allowed to use pitch and level as data-driven features. In these sonifications, “acoustic event waves” were perceived which help to follow the progress of a session. This sonification provides a means to monitor a large number of protocols very quickly. Apart from that, it offers the chance to learn the typical rhythmical pattern of a session. The last sonification extended the second approach by integrating speech markers into the auditory display. The markers facilitate the interpretation of the auditory patterns w.r.t. the therapeutic cycle model.

10.2 Sonifications for EEG Data Analysis

Electroencephalography (EEG) evolved over the last decades as a technique able to provide researchers with physiological data of brain activity. In the project “cortical representation and processing of language” of the Sonderforschungsbereich SFB 360[2], the aim is a neurophysiological analysis of the functional behavior of participating neuronal assemblies during high level cognitive processes with a focus on comprehension of spoken language. Applied techniques are analysis of event related potentials (ERP) and coherence analysis [NL99]. As a rather new approach, sonification of EEG data is now considered as a means of assisting and accelerating data inspection, pattern classification and exploratory data analysis.

In this section, three sonification techniques are developed which provide the researcher with a means for the inspection of short-time Fourier transform spectra from the measurements. Spectral Mapping Sonification (Section 10.2.2) allows frequency-selective browsing of EEG data, Distance Matrix Sonification (Section 10.2.3) allows to follow the time-variant distance matrix of spectral vectors. Finally, Differential Sonification (Section 10.2.4) allows for fast spectrally and spatially resolved comparison of datasets for one subject under different conditions.

[2] `http://www.sfb360.uni-bielefeld.de/`

10.2.1 Experiment and Data

This work was based on an experiment performed at the Brain Research Institute, University of Vienna [MWR99, WR96]. In the current project and in collaboration with P. Meinicke, data mining techniques for the EEG-analysis are investigated and applied to data from the following psycholinguistic experiment. In the experiment under consideration, 25 female participants, aged 20 to 30 years, were seated in a sound reduced chamber and asked to listen to auditorily presented stimuli, that were presented via headphones. Among the several stimuli, there are three sets which are used for the current sonifications:

(i) Spoken Language (story) with Austrian-German speaker (average duration 2 min) are played,

(ii) Pseudo Speech, consisting of auditory patterns generated by amplitude and frequency modulation using a base frequency of 200 Hz and an amplitude envelope which resembles the real spoken sentences, and

(iii) EEGr, where the EEG is recorded for 2 min during rest with open eyes.

The three conditions were chosen in order to identify patterns which emerge from the higher-level cognitive processing of speech rather than from the primary acoustical analysis of the stimuli. Different spectral bands are discerned while analyzing EEG data, which are supposed to play specific functional roles, namely the δ-band (1-4 Hz), θ-band (4-8 Hz), α_1-band (8-11 Hz), α_2-band (11-13 Hz), β_1-band (13-19 Hz) and the β_2-band (19-30 Hz)[3]. Former data analysis [MWR99, WR96] indicated that the α_1-band reflects processes of primary acoustical analysis whereas the β_1-band reflects cognitive components.

The EEG data were recorded with 19 scalp electrodes positioned according to the 10-20 positioning system, measured against the averaged signal of both ear-lobes. Prior to analysis, the signals are band-pass filtered (0.3 Hz to 35 Hz) and digitized using a 16 bit quantization and a sampling rate of $\nu_{SR} = 256$ Hz. Figure 10.4 gives an overview of the available electrodes and their positions on the scalp and further shows some typical time series for selected electrodes.

10.2.2 Spectral Mapping Sonification

The simplest approach to get an auditory representation of EEG data is to use Audification, the direct playback of the raw time series as air pressure variations (see Section 3.7). The main problem with Audification is that the resulting sound is very noisy and it is difficult to control playback speed and pitch independent from each other. Some EEG data audifications are illustrated in Table 10.4. For 3 subjects and all three conditions, two channels (Fp1 and channel T5) are audified. Playing the Fp1 Audification on the left and the T5 Audification on the right audio channel allows comparison of the temporal evolution of both signals and to detect time-dependent correlations.

Obviously, the Audifications sound very noisy. But in all this noise it is possible to perceive some strong low-frequency bursts. Some of these are artefacts that are caused by muscle activities. Since the sound amplitude is normalized to the available quantization range, different Audifications have a different level. Here, it is crucial to eliminate the outliers before using Audification. However, Audification is a quick method in locating such outliers manually as they peak

[3] The upper limits deviate from the common usage (e.g. δ=1-3 Hz, θ=4-7 Hz). The change was made to avoid gaps in the spectrum between the bands.

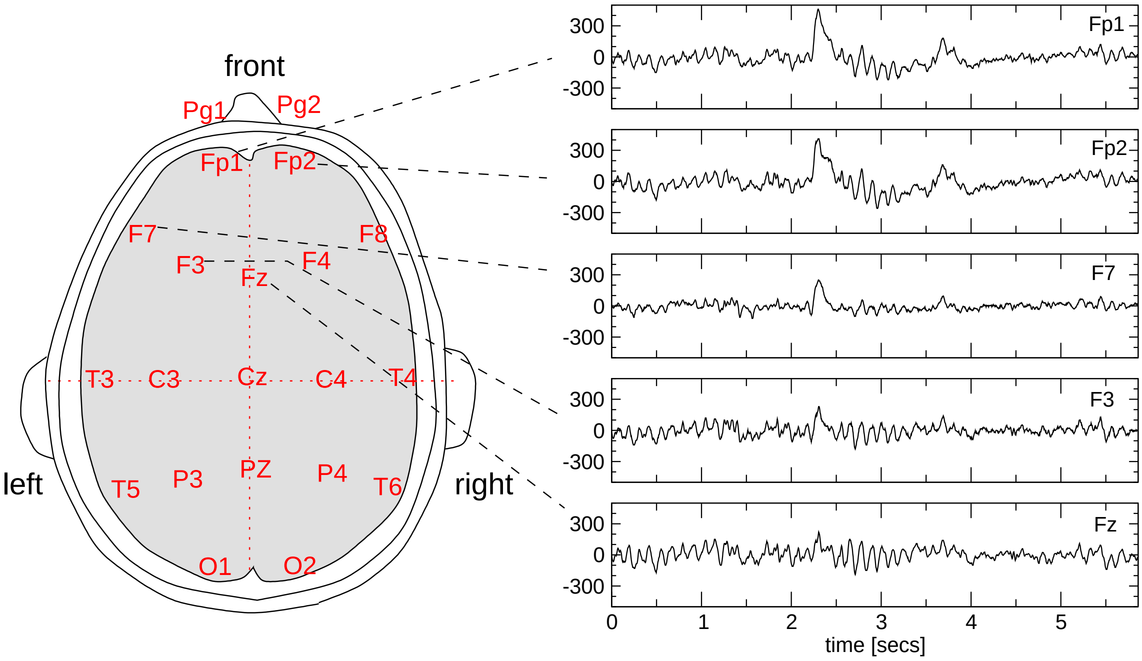


Figure 10.4: Illustration of the available EEG electrodes and their position on the scalp. Some typical measurements are shown. Left picture after [Bir99].

File/Track 39:	S1 EEGr: Fp1 , T5 , Fp1/T5 , speech: Fp1 , T5 , Fp1/T5 pseudo: Fp1/T5
	S2 EEGr: Fp1 , T5 , Fp1/T5 , speech: Fp1 , T5 , Fp1/T5 pseudo: Fp1/T5
	S3 EEGr: Fp1 , T5 , Fp1/T5 , speech: Fp1 , T5 , Fp1/T5 pseudo: Fp1/T5
Description:	Audification of EEG Data for 2 electrodes (Fp1 and T5) for subjects S1–S3.
Duration:	≈ 1 s

Table 10.4: Audification of EEG data (2 electrodes, 3 subjects and all 3 conditions. Besides the dominating noise, pitched spikes can be heard.

above the average noise level. But apart from these dominating sound elements, in some Audifications specific pitched patterns are perceived, see example for subject S3, speech, channel T5 in Table 10.4. Here, three pitched bursts occur with decreasing pitch. These pitches correspond to activity within the α_1-band. Besides this, interesting acoustic features can be perceived in in the example for subject S1, EEGr, channel Fp1 in Table 10.4: four short pitched events with a down-chirp pitch curve. The pitch corresponds to peaks within the β_1-band. Such features are easier to detect from the sound than from the extremely noisy time series data and even in spectrogram plots, these chirps are hardly visible (see Figure 10.5).

The main problem of Audification is, that independent control over spectrum and duration is limited. Therefore the sonification technique presented here, uses the short time Fourier transform (STFT) of the time series as starting point. Given the measurements $s_i[n]$ where $i = 0, \dots, 18$ determines the channel and n is the sample number, the STFT is computed for each channel i by

$$\tilde{s}_i[m,k] = \sum_{n=0}^{N-1} s_i[Cm+n]w[n] \exp\left(-i\frac{2\pi nk}{N}\right) \qquad (10.3)$$

where m is the frame number, C the offset between succeeding frames, $k = 0, \ldots, N/2$ the frequency index and N the window width in samples. For the following sonifications, a triangular window $w[n]$ is used. The value $|\tilde{s}_i[m,k]|$ denotes the Fourier amplitude at frequency $f(k) = k\nu_{SR}/N, \ \ k \in \{0, \ldots, N/2\}$ in the mth frame, including the time step indices $[Cm, Cm+1, \ldots, Cm+N-1]$.

A compromise between coarse frequency resolution for small N and coarse time resolution for large N must be found. For further analysis $N = 256$ is chosen corresponding to an analysis window of 1 s. Figure 10.5 shows spectrograms for the different electrodes of one subject.

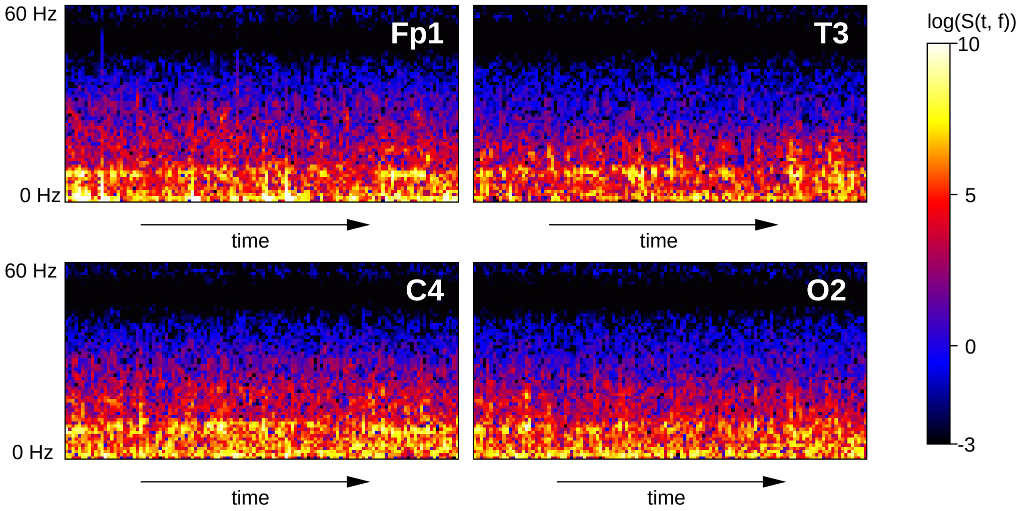


Figure 10.5: Spectrogram of measurements for 4 channels, shown for one subject and the EEGr condition for 64 s data. The color is mapped by $\log(\tilde{s}[n,k])$. Patterns of high energy can be seen in the α-bands.

After this preprocessing, 19 spectrograms are given for each time frame. Spectral Mapping Sonification now superimposes for each selected electrode i a set of N_{osc} time-variant oscillators whose frequency f_n for $n = 0, \ldots, N_{osc}-1$ is given by

$$f_n = \ln(2) \exp\left(p^o_{min} + \frac{n}{N_{osc}-1}(p^o_{max} - p^o_{min})\right) \tag{10.4}$$

where $[p^o_{min}, p^o_{max}]$ denotes the desired pitch range in octaves. Let $\tilde{s}^k_i(t)$ denote the time-variant function from interpolating the sequence $\tilde{s}_i[0,k], \ldots, \tilde{s}_i[M,k]$ such that $\tilde{s}^k_i(0) = \tilde{s}_i[0,k]$ and $\tilde{s}^k_i(T) = \tilde{s}_i[M,k]$. Then the amplitude of the kth oscillator is computed by

$$a_k(t) = \hat{a} g_\delta\left(\tilde{s}^k_i(t) / \max_{t'} \tilde{s}^k_i(t')\right) \tag{10.5}$$

where $g_\delta(\cdot)$ is a nonlinear function which suppresses all amplitudes less than a given threshold δ.

Only very few parameters need to be specified for sonification, namely the duration per frame T_f, the pitch range $[p^o_{min}, p^o_{max}]$ and an EEG frequency range to be used. With this sonification, the activity in a specific spectral band can be monitored. Assume, we are interested in the α-band from 8 Hz to 13 Hz. As the window width is 1 s, the frequency resolution is 1 Hz and thus 6 frequency cells are within the selected range. Thus six time-variant oscillators are created whose amplitudes are determined by the corresponding Fourier series coefficients. Suited time compressions are about 50, allowing to monitor 50 s of experimental data in 1 s. If more than one channel is of interest, the sonifications of chosen channels can be superimposed. To compare different regions on the scalp, each channel can be assigned to the left or right audio channel. Some example sonifications to illustrate the sound are presented in Table 10.5.

File/Track 40:	$\delta = 0.4$, [0-30 Hz] left: T3,T5 right: T4 T6.	S1	S2	S3	S4	S5	S6
	$\delta = 0$, [0-30 Hz] left: T3,T5 right: T4 T6.	S1	S2	S3	S4	S5	S6
	$\delta = 0$, [0-30 Hz] left: F7,F3,T3,C3,T5,P3. right: F4,F8,C4,T4,P4,T6.	S1	S2	S3	S4	S5	S6
Description:	Spectral Mapping Sonification of EEG data						
Duration:	$\approx$ 4 s						

Table 10.5: Sound examples for Spectral Mapping Sonifications of EEG data, for the conditions pseudo speech and speech in a sequence with a short gap in between (2.5 s in real-time).

The threshold δ allows to control the complexity of the sonification. With larger values of δ most of the signal energy is cut off and only the spectral peaks contribute to the sound. Correlations between different bands can be perceived as pitch patterns, e.g. one may often observe high pitches following some low pitched sounds. Since this sonification technique enables to listen to the data at various timescales including real-time, it may even be a useful technique for monitoring brain activity parallel to an acoustic representation of the stimulus.

Unfortunately, the listener of these sonifications can not distinguish the contributions of different channels. This limitation can be partially overcome by using different timbres to represent the contributions of different electrodes. However, this would also reduce the spectral resolution since complex timbre is always spread in spectrum.

Summarizing, Spectral Mapping Sonification is a technique for monitoring EEG data selectively in frequency and it allows the direct comparison of two sets of channels.

It might be applied to detect outliers, or to get a rough idea about how a condition influences the data by listening to the data for different conditions in a sequence.

10.2.3 Distance Matrix Sonification

Whereas the previous sonification was concerned with allowing the user to follow the spectral activation within the brain, now sonification focuses on a less direct and more abstract observable: the synchronization of different brain areas as a function of time. It is an open research question how information is processed in the brain and how the processing of different parts is related. A working hypothesis is that electrodes having a similar spectral activation profile over time are concerned with similar information processing. Such information can be expressed in form of a time-dependent distance matrix $\mathbf{D}$ with entries

$$D_{ij}[m] = \|\hat{\hat{\mathbf{s}}}_i[m] - \hat{\hat{\mathbf{s}}}_j[m]\| \; . \qquad (10.6)$$

The elements D_{ij} contain the Euclidean distance between the normalized spectral vectors of channel i and j in the mth window, beginning at sample $C \cdot m$ with time $t = Cm/\nu_{SR}$. Figure 10.6 shows some distance matrices for succeeding time frames. Small entries in $\mathbf{D}$ indicate similar activity in the corresponding channels given by row and column index. Usually a high similarity is expected for electrodes with a small topological distance on the scalp. Topological distances between electrodes have been measured by a Polhemus tracker [Pol].

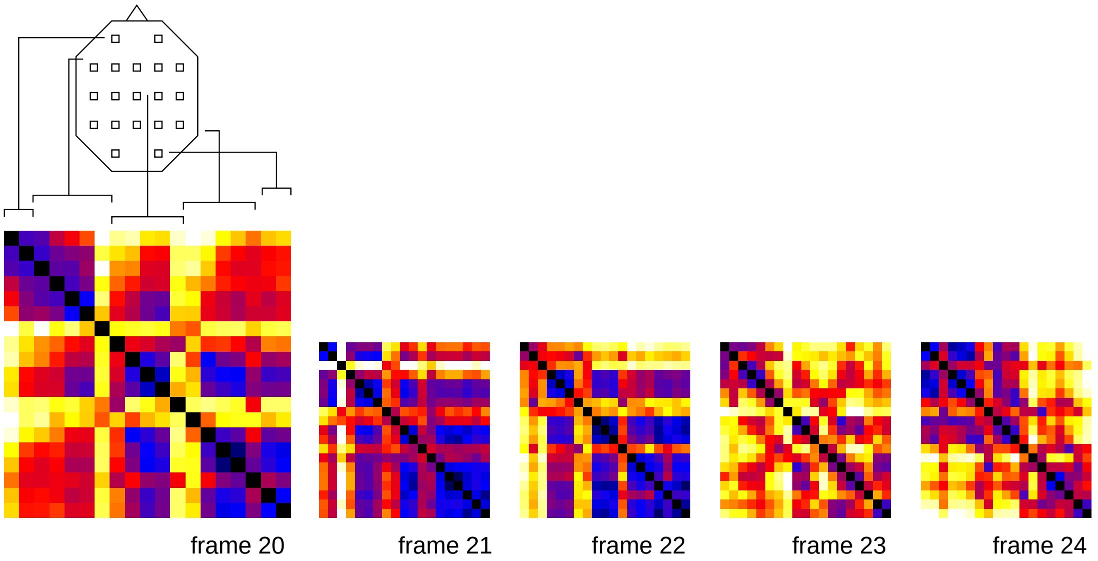


Figure 10.6: 19×19-distance matrix for spectral vectors for 5 succeeding frames. The channels are ordered as depicted from left to right. Small values are represented by dark colors, indicating similar activity between the respective channels.

For sonification, the topological distance between electrodes is used to drive the pitch of auditory grains which are superimposed into the sound vector at the appropriate onset. The similarity $\exp(-D_{ij}[n])$ is used to drive the level of these grains. Thus loud and high pitched contributions indicate long-range couplings. However, such couplings are even more interesting, if the electrodes carry significant energy. Thus, $(\|\tilde{\mathbf{s}}_i\| \cdot \|\tilde{\mathbf{s}}_j\|)$ is mapped to the duration of the grains so that durations between 20 ms and 200 ms are obtained for the available range of inputs. Hereby correlations between channels having few energy automatically do not dominate the sound as they last much shorter than the terms with channels of higher energy. In addition, a clipping is done so that events with a too low intensity product are removed. This accelerates the computation and makes it easier to discern the dominating couplings. As a final ingredient, sound spatialization is used to give a rough indication in which part of the brain the coupling takes place: if both electrodes are located on one side of the scalp the sound is played on the respective audio channel, couplings between different hemispheres are represented by tones played from the center.

Sound examples for distance matrix sonifications are compiled in Table 10.6. The sonifications indicate that in the speech condition more long range couplings occur than in the pseudo speech condition. Distance Matrix Sonification allows to inspect the range and strength of couplings between different channels, although it is not possible to conclude from the sounds on the source of the couplings.

File/Track 41:	frequency range 8–20 Hz. subjects: S1 S2 S3 S4 S5 S6
Description:	Distance Matrix Sonification for concatenated datasets for the pseudo speech and speech condition. A noise burst separates the two parts
Duration:	$\approx$ 5 s

Table 10.6: Sound examples for Distance Matrix Sonification of spectral vectors. High-pitched loud tones indicate frequency selective couplings of long range.

10.2.4 Differential Sonification

The following EEG data sonification allows the comparison of data recorded for one subject under different conditions in order to accelerate the detection of channels and frequency bands along which the conditions cause systematic differences. In contrast to the previous sonifications, here the time axis is used to distinguish the location of the electrodes, scanning the brain from the frontal side to the occipital electrodes. A basic question while comparing EEG data from different conditions is, what channels have a different activation and at what frequencies. For the comparison, for each condition α , each channel i and each frequency band k, the time sequence of Fourier coefficients $|\tilde{s}^i_\alpha[j,k]|,\ j = 1, \dots, N_{i,\alpha}$ is used, which is obtained from the STFT as described above. The mean

$$\mu^i_{\alpha,k} = \frac{1}{N_{i,\alpha}} \sum_j |\tilde{s}^i_\alpha[j,k]| \tag{10.7}$$

and the standard deviation

$$\sigma^i_{\alpha,k} = \sqrt{\frac{1}{N_{i,\alpha}-1} \sum_j (|\tilde{s}^i_\alpha[j,k]| - \mu^i_{\alpha,k})^2} \tag{10.8}$$

is computed. Assuming both sequences to be independent samples from the same distribution, the random variable

$$\tilde{t} = \frac{1}{\sigma^i_{\alpha,\beta,k}} (\mu^i_{\alpha,k} - \mu^i_{\beta,k}) \tag{10.9}$$

with

$$\begin{aligned} \sigma^i_{\alpha,\beta,k} &= \sqrt{K((N_{i,\alpha}-1)(\sigma^i_{\alpha,k})^2 + (N_{i,\beta}-1)(\sigma^i_{\beta,k})^2)} \\ K &= \frac{1}{\nu}\left(\frac{1}{N_{i,\alpha}} + \frac{1}{N_{i,\beta}}\right) \end{aligned}$$

is student-t distributed with $\nu = N_{i,\alpha} + N_{i,\beta} - 2$ degrees of freedom. With increasing values of $\tilde{t}$, it gets more significant that the distributions for the condition α and β differ. $\tilde{t}$ is thus used within the sonification to decide, if a sonic marker for frequency band k and channel i contributes to the sonification and at what level.

The sonification for a comparison of EEG data for the conditions α and β consists of a sequence of sonic events whose structure and meaning is given as follows:

- **Time Ordering:** the sonification can be regarded as a scanning from the frontal side to the occipital side. To increase the utility of the time axis, electrodes from the left to the right side are separated within each row as shown in Figure 10.7.

- **Spatialization:** comparison results concerning electrodes from the left (resp. right) side of the brain are presented on the left (resp. right) audio channel.

- **Spectral Mapping:** the frequency band center frequency is a monotonous function of the initial pitch of the sonic marker. Thus changes within the β-band result in high-pitched events, changes within the δ-band in low-pitched markers. Equal musical intervals (e.g. quint) are used as spectral spacing between neighbored bands.

- **Spectral Motion:** comparison results indicate either increase or decrease of activation. To monitor these qualitative states, frequency drifts (chirps) within the markers are used. Although this may not be the most intuitive mapping (an energy increase would be associated with an increase of level instead of pitch), this assignment leads to a clearer perception of the difference.

- **Event Level:** comparison results with $\tilde{t}$ not exceeding a threshold $\tilde{t}_{min}$ are suppressed, allowing to reduce the complexity of the sonifications so that only the most significant changes are audible. The level of the sonic markers increases with the values of $\tilde{t}$.

- **Marker Sounds:** for sound generation, exponentially decaying sine functions are superimposed. The frequency is driven linearly from its initial to its final value.

- **Acoustic Axes/Labels:** For sonifications longer than 1 s, an uprising arpeggio[4] is played beforehand to present all center frequencies used for the frequency bands. To facilitate the spatial assignment of perceived events, a marker sound is played on each new row to be started. This is particularly useful in the case that only few markers are mixed, for instance if a high threshold of $\tilde{t}$ is used.

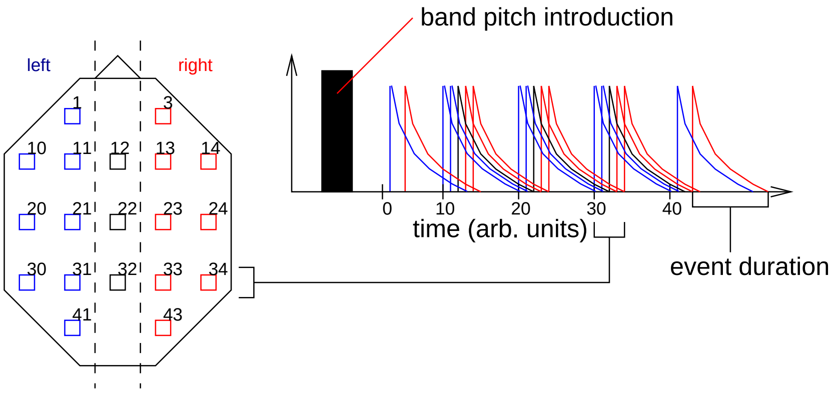


Figure 10.7: Illustration of temporal and spatial organization of acoustic elements in Differential Sonification. The plot shows the amplitude envelope of the events for all electrodes.

Figure 10.8 shows a spectrogram for a typical differential sonification. Here, the up/down-chirp can be seen. The corresponding sound example is Ex-DiffSon-1 (Track 42).

A set of differential sonifications for 6 subjects and the conditions pseudospeech/speech and EEGr/speech are compiled in Table 10.7.

For the untrained listener, it may be an easier start to begin with the examples of 4 s duration for learning to interprete the sound. However, after some training, shorter sonifications should be preferred as they allow to scan more subjects in less time and to keep them in short-term auditory memory as ’auditory gestalts’. It can be perceived from the sonifications that in the speech condition more activity can be found mainly in the lower frequency bands (α_1, α_2) and mainly in the occipital sector of the brain. The sonifications also show that brain activity is higher during speech perception throughout the whole brain than under the EEGr condition.

Differential Sonifications are an example for using sound in a more abstract way: time is given a different meaning than in the data. This sonification can be extended in many ways, for

[4] Def.: a chord played one note at a time

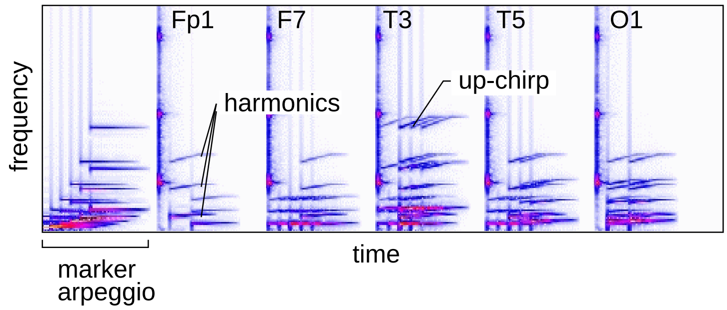


Figure 10.8: Spectrogram of a EEG Comparison Sonification for conditions pseudospeech versus speech. Frequency Drifts are seen here as spectral motion.

File/Track 43:	pseudospeech – speech (1 s):	subject S1 , S2 , S3 , S4 , S5 , S6
	same with higher threshold (1 s):	subject S1 , S2 , S3 , S4 , S5 , S6
	speech – EEGr (1 s):	subject S1 , S2 , S3 , S4 , S5 , S6
	pseudospeech – speech (4 s):	subject S1 , S2 , S3 , S4 , S4 , S6
Description:	Differential Sonification of EEG Data	
Duration:	1 s and 4 s	

Table 10.7: Sound Examples for Differential Sonification.

instance by using marker events of higher timbral and temporal complexity. The analysis problem at hand determines how to connect the data to the properties of the sound.

10.2.5 Conclusion

EEG data are a particularly interesting type of data for the application of sonification since they consist of multiple time series. This kind of multi-channel data contains a lot of noise which complicates the automatic detection of patterns so that an exploratory data analysis can profit very much from the humans' highly-developed auditory skills in signal/noise separation.

Sonification can be applied in data analysis for different tasks. For a first data screening, Audification may be used, which allows to detect outliers and rhythmical and pitched patterns in the raw signals. Spectral Mapping Sonification allows the researcher to contribute his own listening skills to investigate frequency-specific patterns of different EEG channels. Apart from that the data are monitored at a high time-resolution. Distance Matrix Sonification transforms the data into a sound that allows to detect long-range couplings of brain activity with high temporal resolution. Finally, Differential Sonification enables researchers to scan large databases of EEG data recordings in a very condensed way. The trained listener can conclude roughly which regions, in which hemisphere and at which frequencies are affected by the condition under investigation.

Only very few of the possibilities of sonification for the analysis of EEG data have been addressed so far. One particular point of interest for ongoing work is to use an acoustic representation of features of the stimuli played simultaneously to the sonifications. Simultaneous playback is supposed to facilitate the relation between stimulus and brain activity as a response to the stimulus so that hopefully characteristic dynamic patterns can be detected.

10.3 Sonification of Multi-Channel Image Data

In this section, sonification is applied to a common data type found in many fields of science, namely a stack of 2d images. Such data occurs for instance in geography where different maps of a region give different types of information, in medical research (e.g. multispectral MRI) and in micro-biology for instance. For the practical application of sonification, several questions arise: how important is the pleasantness of sounds, how long should sonifications last and how much can sound increase users' performance. In this section, data from the domain of Multi-Parameter Fluorescence Microscopy is used in a cooperation with T. W. Nattkemper and W. Schubert. The image data result from Multi-Parameter Fluorescence Microscopy experiments. Tissue probes with biological cells are recorded iteratively stained with different markers, which leads to fluorescence of certain cells in the probe under each condition. In the resulting stack of grey-level 2d images, locations or the cells remain constant. The collaborative work is concerned with enabling automatic detection of cells in such image stacks. In a consequence facing an increasing amount of available data, one finds techniques to detect patterns within the data become more and more important.

Figure 10.9 shows a typical image obtained from the image acquisition system as well as a series of images in the image stack of a small region. For the exploration of such image stacks, a

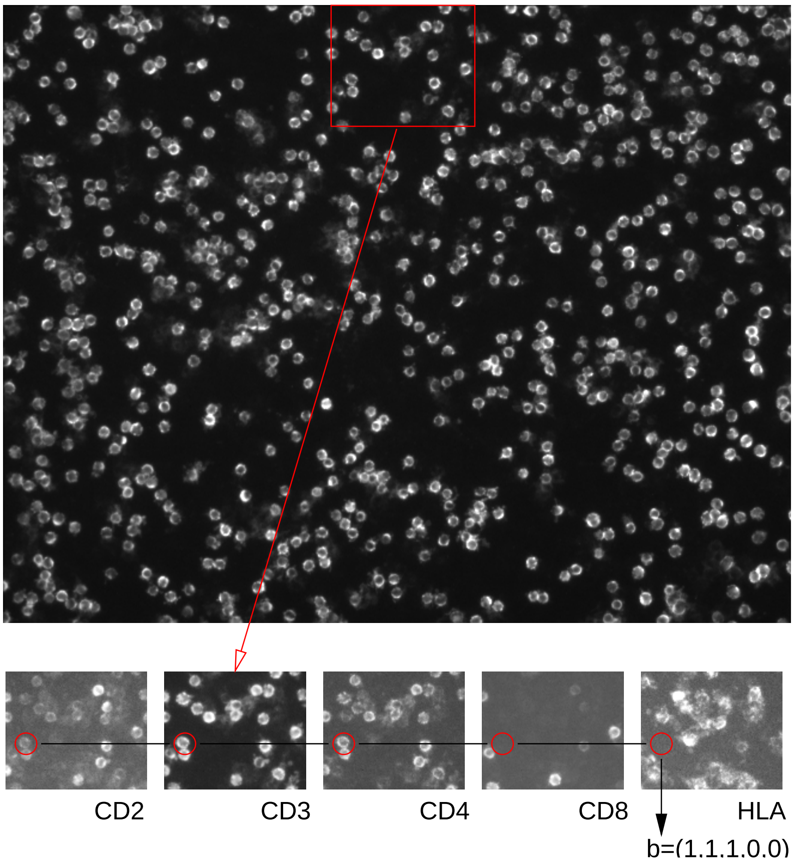


Figure 10.9: Fluorescence Micrographs from a blood sample for 5 different markers. For one example corresponding cells are highlighted. The cell can be characterized by a binary vector $\mathbf{b}$ obtained from binarizing the average fluorescence intensity.

straight forward technique would be to plot the stack of images on a grid. However, to recognize the patterns, the eyes have to sequentially attend the corresponding cell images which leads to a large work-load for the visual search in the image. This can be facilitated by marking all cells. As an alternative, up to three images can be presented simultaneously by assigning the grey-level images onto different color channels, e.g. to the RGB or HSI color space [GBC81]. While this color-mapping technique preserves the visual inspection of the cell images, it is limited to three colors or frequency channels. In contrast, our ears are sensitive to a much larger number of frequencies all of which are processed in parallel. Therefore, we employed the color-based channel coding strategy *in the auditory domain* leading to a new sonification strategy aiming to assist the perception of correlations in image data stacks by utilizing the multi-channel processing capabilities of our auditory system simultaneously to visual images [HNSR00].

10.3.1 Auditory Maps

In a printed stack of images or maps, information can only be accessed sequentially as the eyes need to attend the corresponding points on the individual images one by one. The access to information can be strongly enhanced by using interactive visualizations within the computer because such data displays can be dynamic. They can e.g. use additional windows to summarize the image stack data at the mouse pointer in a Chernoff face or histogram. However, in general these visualizations require extra space on the display and increase its complexity, eventually overloading it and thus making comprehension more complicating. Auditory maps [MMBG94] offer a method to integrate additional information into the user interface by using the auditory channel, distributing the perceptual load more evenly among the available senses. As a second important benefit, for the visual channel this entails less disruption of the view of the underlying image.

Interactivity is essential for accessing auditory maps, since usually only limited parts or regions of a map are focused at a time. Possible modes of acoustic map interactions are

- **Browsing:** a pointer device (e.g. mouse) is used to access information from the map. Two modes can be distinguished here. *Ambient sound* may arise from the pointer location in the image while the pointer is moved, or information may be retrieved by *active probing* of the image, e.g. by clicking a button at a certain location to trigger the acoustic presentation of properties in the map. The **Aura** is the region around the pointer which maximally contributes to the sound. It is often chosen by the user according to his needs. In this case, Browsing cells uses only a small region the size of a single cell.

- **Summarizing:** Sonification can be used to summarize information of all patterns within a larger region, or within the whole image stack. For this purpose either a path or a region of the image is marked in the image prior to rendering the sound.

In this section, the focus will be put on techniques for browsing an image stack. Sonifications to summarize the content of a region can be derived from a suitable composition of browsing sonifications. Since the choice of sonification technique depends on the analysis task at hand, details about the sound generation are reported after taking a closer look at the application at hand and its specific needs.

10.3.2 Multi-Parameter Fluorescence Microscopy

In our collaboration, Multi-Parameter Fluorescence Microscopy data of immunofluorescently labeled lymphocytes have to be analyzed. One experimental dataset consists of N intensity images of the sample, each of them recorded with a different marker. As a result of a specific immunolabeling technique [Sch92, Sch97], in each image different subsets of the lymphocytes appear with high intensity values, indicating the existence of a specific cell surface protein. Because the positions of the cells are not affected by the labeling process, the N fluorescence signals of a cell can be traced through the image stack at constant coordinates as shown in Figure 10.9.

So far, the evaluation of these data is performed in two steps: in a first step the images are segmented into cells and background or environmental tissue. As a result of this step, a set of vectors $\mathbf{v}_i$ is available, each describing one cell center coordinate. In a second step, the image information through the stack is reduced to a binary marker vector $\mathbf{b}_i$ for each cell i, whose element b_{ij} is 1 if the cell at $\mathbf{v}_i$ is fluorescent in image j of the stack. To visualize the results of this second step, histograms of the absolute frequencies of the *marker combination patterns* [Nat01] are generated. The x-coordinate of the histogram bar for a pattern $\mathbf{b}$ is computed by $x_b = \sum_j 2^{jb_j}$. A major drawback of this visualization is the fact that histogram bars of similar patterns are not shown close together. To measure the similarity, here the Hamming distance

$$d_H(\mathbf{b}_i, \mathbf{b}_k) = \sum_{j=1}^{N} |(b_{ij} - b_{kj})| \tag{10.10}$$

is used, indicating that no further knowledge about the markers is available.

When evaluating the data, the biomedical expert is mainly interested in

- finding unusual patterns specifically those correlating with a disease under investigation
- detecting already known patterns which are known to be shared by certain diseases
- detecting deviations from known patterns correlated to a specific biological phenomenon.

Thus, on the basis of the binary markers, sonification should assist in comparing, identifying and distinguishing patterns. From the perspective of cell biology, however, more information than the marker combination vector can potentially be of interest. So far this is not considered due to the lack of suitable exploration techniques. The spatial distribution of markers on the cell membrane is neglected as well as the local distribution of marker patterns around a selected cell.

10.3.3 Sonification Design

The developed sonifications are acoustic presentations of binary marker vectors. Out of the interests of the biomedical expert, some requirements on the sound representing a pattern can be formulated:

(a) Cells with identical combination patterns should be perceived as identical sounds. This is trivial for sonifications using the binary marker vector directly for sound generation, but it is an essential requirement for sonifications that base directly on the cell image data.

(b) Cell patterns with similar patterns in terms of their Hamming distance should be perceived as similar.

(c) Cell pattern sonification must be consistent to extensions. This is crucial for later extensions of the marker library, in order to keep the learning effort small for the addition of markers.

In addition to these requirements there are further demands from the practical side:

(d) To enable quick browsing and comparison, the sound should not last longer than about 2 s.

(e) The sound patterns have to be easy to memorize. This can be achieved by using sounds that are familiar to human listeners like speech sounds or musical sounds.

(f) sonifications should not be intrusive, and they should be pleasant as they are potentially heard very often.

Different techniques for sonification are available, like Parameter Mapping, Earcons, Auditory Icons or Model-Based Sonification. Parameter Mapping would use a set of acoustic attributes and compute their values by a mapping of the components of the binary marker vector. The main problem here is, that different acoustic attributes are not independent (missing orthogonality) and that they have a different perceptual strength: e.g. pitch perception will dominate over attack perception. An alternative approach is to use a superposition of acoustical elements to generate a auditory scene for each single pattern. In this approach, for each single marker, one or more acoustic events are rendered using any sonification technique, thus the sonification of a pattern is given by a combination of all events. This organization of auditory display is referred to as **Auditory Scene Generation** [Bar94]. Examples for auditory scenes are a symphony orchestra or any other situation where several sound sources come together.

To fulfill (f), the domain of musical sounds with diatonic scaling seemed appropriate. In addition, musical sounds are much easier to synthesize compared with environmental sounds or speech sounds needed for other scenes. Besides that, people are usually good at remembering musical phrases (short melodies) and distinguishing rhythmic and harmonic patterns. The sonifications presented first only use a single auditory stream and organize sound events of a single source in time.

Rhythmical Sonifications:
A binary marker vector is presented as a chain of tones, playing one tone per marker after another. Thus the onset is determined by the marker index while the pitch is determined by the data value. Using a percussive sound like a ”conga drum”, a rhythmic pattern emerges for each cell. Sound examples are available in Table 10.8. This approach is in agreement with (c), as it can easily be extended by adding new markers at the end. However, the pattern dimension that can be presented is limited because a time interval of about 100 ms is required between events to allow their resolution. With a shorter time interval, the perceptual resolution is exceeded. Using fixed 100 ms per event, this sonification is limited to less than 20 markers before (d) is violated.

Harmonic Sonifications:
In this sonification a marker is only characterized by its pitch. This makes it possible to play several markers at one time, as humans perceive sets of tones played parallel as a musical chord. This offers the whole range of harmonical forms to distinguish patterns. A proper selection of marker pitches is required to avoid dissonant harmonic clusters. Such chords are difficult to discern for untrained listeners. However, this sonification works well even with larger marker libraries if the marker vector $\mathbf{b}$ is only sparsely set. A suitable alternative for large marker libraries is to split the sonification into a sequence of two or more chords using the same strategy. Then, a tone can be used repeatedly for different markers. Some sound examples for a division into two chords will

be given below. As an extension, the level of the tones can be driven by the actual intensity of the fluorescence, corresponding to the concentration of protein on the considered cell. Listen to the sound examples in Table 10.8.

Melodic Sonifications:
Combining rhythmic and harmonic sonifications results in melodic sonifications. Markers now determine both the onset and the pitch. Unset markers remain silent leading to pauses in the motive. These sonifications have shown to be superior to the harmonic and rhythmical sonification concerning memory and comparison, according to our subjective experiences.

There are many other ways to get melodic sonifications. An interesting variant is to divide the sonification into three chunks: the first plays an arpeggio (an uprising chain) of the tones for all activated markers, the second chunk plays all sounds for markers whose assignment is ambiguous. The last chunk plays all off-markers.

The melodic sonifications proved most useful for the task of comparing cells. They have been optimized using synthetically generated cell images. Then they were tested with real data from Multi-Parameter Fluorescence Microscopy Experiments.

Multitimbral Sonifications:
This sonification uses a mixture of instruments, onsets and rhythmic patterns to represent the markers. Each marker is characterized by a musical motive, consisting either of a single tone, two tones played together with determined pitch and interval or a rhythmical sequence. The musical instruments vibraphone, violin, electric base and a plucked guitar are used and for each instrument only 2–3 different motives are configured. The motives are chosen to be localized within two quarter-note of a bar lasting 1 s. Some of these marker pattern sonifications are collected in Table 10.8.

10.3.4 Examples

For the following sonifications, a stack of 10 synthetically rendered images are used, shown in Figure 10.10. These images were rendered with a cell image generator developed by T. Twellmann and T. W. Nattkemper [Nat01]. An average distance between cell centers of 20 pixels and a cell diameter of 15 pixels has been chosen in order to avoid partial occlusions. The user interface for the browsing is an interactive tool developed in Neo/NST [Rit00]. A screenshot is shown in Figure 10.11. Here the mouse pointer is used to probe the image stack.

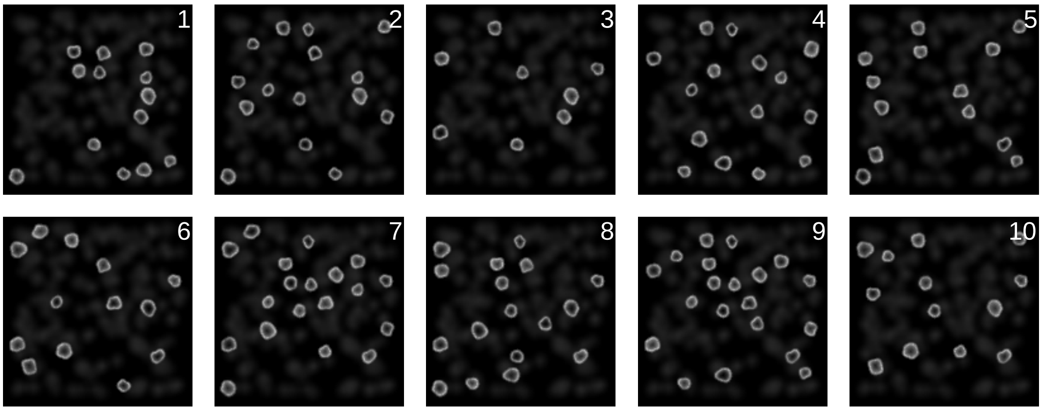
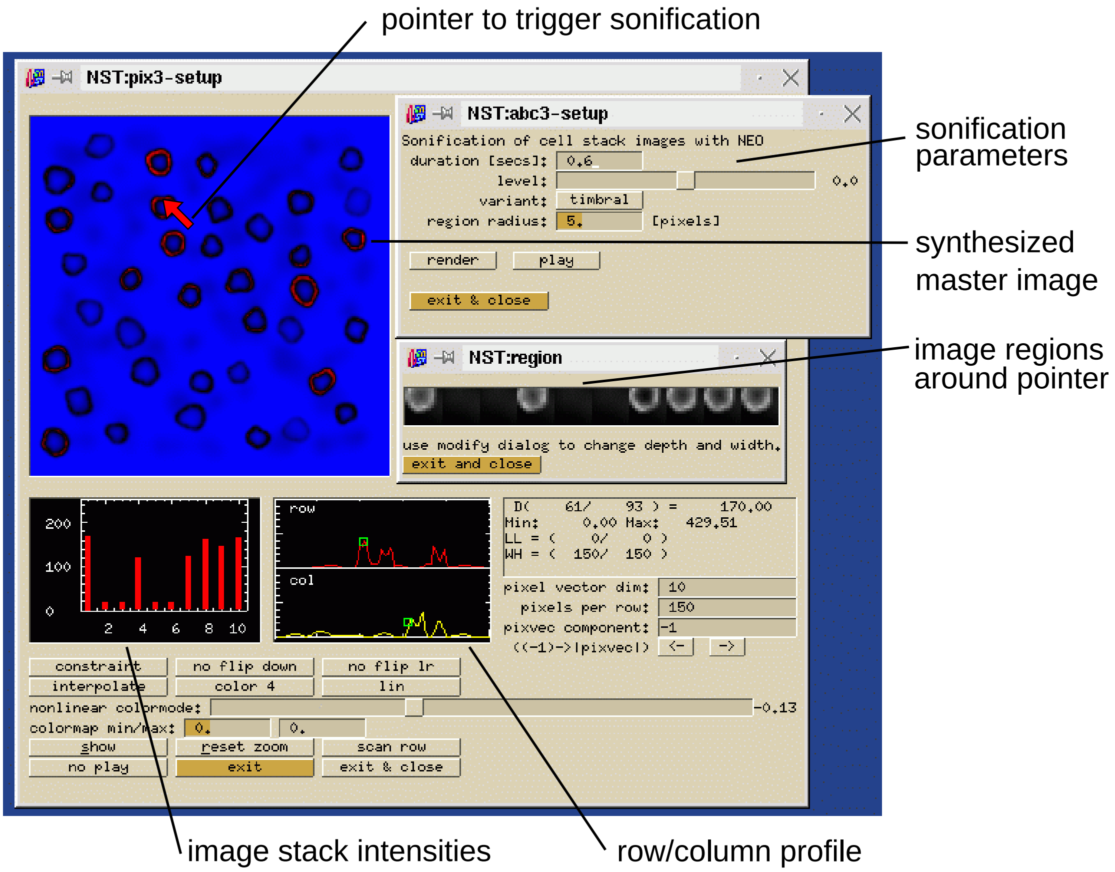


Figure 10.10: Synthetically rendered images used to exemplify cell pattern sonification.



```
Sonification of cell stack images with NEO
 duration [secs]: 0.6
           level:                     0.0
         variant: timbral
   region radius: 5.       [pixels]

use modify dialog to change depth and width.

 D(    61/    93 ) =      170.00
Min:     0.00 Max:    429.51
LL = (    0/     0 )
WH = (  150/   150 )

pixel vector dim: 10
  pixels per row: 150
pixvec component: -1
 ((-1)->|pixvec|)

nonlinear colormode:   -0.13
colormap min/max: 0.   0.
```

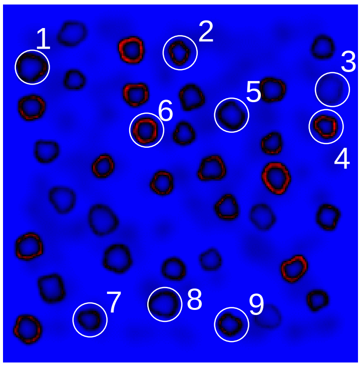
Figure 10.11: Graphical User Interface for exploration of cell stack patterns

For visualization, a synthesized image is shown in which all detected cells are visible. Either the maximum intensity throughout the image stack or the average of all images is used for this. In the latter case, the cells intensity is almost proportional to the number of active markers as shown in Table 10.8.

10.3.5 Conclusion

Sonification of Multi-Channel Image Data was presented as a method to inspect high-dimensional data by listening to their sonifications. In contrast to the visualizations which are limited to three color channels, sonifications allow to find correlations in data of much higher dimensionality. As a second benefit, the sonifications provide this information without disrupting the view onto the image.

To test the described sonifications strategies, some synthesized cell stacks were browsed by the author and T. Nattkemper either with or without the support by the sonifications The experiences from these testings show that sonification indeed supports the recognition of frequent patterns, and especially simplifies the quest for identical patterns. Furthermore these testings indicated that pleasant sounds are much preferred and that musical tones are in that regard preferred to the rhythmical patterns. In the beginning, longer sonification times are preferred. After some training, however, short sonifications are accepted as they accelerate the examination-process. This indicates that the duration should be adjustable also for the end-users. In addition, sonifica-



b_1= (0,0,0,0,0,1,1,1,0,1)
b_2= (0,1,0,1,0,0,1,1,1,0)
b_3= (0,0,0,1,0,0,0,0,0,0)
b_4= (0,0,1,0,1,1,1,1,1,1)
b_5= (0,0,0,1,0,0,1,0,1,0)
b_6= (1,0,0,1,0,0,1,1,1,1)
b_7= (0,0,0,1,0,0,0,1,1,0)
b_8= (0,0,0,1,0,0,0,1,1,0)
b_9= (1,1,0,1,0,1,0,0,0,0)

File/Track 44:	Rhythmical:	b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8	b_9
Track 45:	Harmonic:	b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8	b_9
Track 46:	Harmonic/(2 chords):	b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8	b_9
Track 47:	Melodic:	b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8	b_9
Track 48:	Multitimbral:	b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8	b_9
Description:	Marker Pattern Sonifications									
Duration:	$\approx$ 1.4 s									

Table 10.8: Sonification of cell fluorescence patterns. Sound examples for rhythmic, harmonic, melodic and multitimbral sonifications for 9 synthetic cells in the image above are given.

tions are less intrusive if they do not dominate the listening space. Like the sound of a keystroke or mouse click which is present on every action but does not impair concentration, soft sonifications are better compatible with an exploratory work-flow. For the fast scanning of cell images, the harmonic and melodic patterns seem to be suited best. Compromises between these prototypes, like the harmonic sonification with two chords are possible and may be chosen, if there is some knowledge available about the marker. For example, tones of related markers could be grouped in one chord.

These qualitative results provide some guidelines for further sound-refinements. Of course, psychophysical experiments are substantially required to assess the use of sonification and to compare different strategies.

For a real-world application in routine laboratory work, the biologist, resp. biomedical experts have to learn to understand the sonification and relate the sound to the marker patterns. For this it is necessary to establish standard sonification procedures, which allow the users to get used to sonifications. Of course, such a standard has to include ways to extend the sonifications without changing the previously learned patterns. The sonifications presented here can be extended in various ways:

(i) to give an acoustic summary of all patterns within the whole image stack, a sonification can be composed as a sequence of sonic marker combination patterns, playing the most frequent patterns with an amplitude mapped from their frequency. Playing this sequence in descending order of the frequency, a characteristic score results for each image stack. This extension makes use of the learned pattern representations.

(ii) Information about the cell size, its contour and fluorescence composition along the contour can be included in the sonifications by driving the acoustic structure of marker events.

(iii) The marker library can be extended by adding further events resp. tones.

The application of sonification methods for browsing marker combination patterns from real image stacks by the biomedical expert will be the next steps in this collaboration.

10.4 Conclusion

The presented applications show how sonification can contribute to enhance practical data inspection. A rigorous empirical testing of the developed methods, however, was out of the scope of this work. A general conclusion is that so far not sufficient standard techniques exist to be used for concrete sonification problems and therefore each application demands an individual making of sonifications which incorporate the specificities of the domain. Also, these specialized sonification are highly adapted to the task at hand like detection, comparison, searching or summarizing.

The sonification models presented in Chapter 8 did not come to application in the three cases. The reason for this is that sonification models work independently from the meaning of the features in a dataset. However, in concrete application researchers pay attention to the features and prefer to have sonic attributes which are more directly correlated to the features. Another reason is, that both in the EEG data and the psychotherapeutic protocols, time is an inherent parameter which offers itself to be used as sonification time if the interest is on the dynamic evolution of data. Sonification models are thus supposed to be a better choice for application in domains where nothing or only very little is known about associations between the features. In some of the presented applications, sonification models may be a suited choice for the next steps, e.g. in the cell biological application where PCS can be applied to compare the clustering of whole datasets with binary marker combination for image stacks from different patients.

Chapter 11

Conclusion

In this thesis, the Model-Based Sonification (MBS) framework has been introduced as a new concept to develop Sonification Models that allow to render sound for high-dimensional datasets. The framework provides guidelines for the construction of sonification models that allow to interact with a virtual data-driven sounding object or environment. Sonifications according to the concept show the ability to avoid some of the problems encountered with the classical sonification approaches, namely Parameter Mapping Sonification (PMS). The key difference is that PMS addresses musical listening skills whereas MBS addresses everyday listening skills. Whereas PMS is limited to the dimensionality of the attribute vector, MBS allows for the definition of sonification models that operate on data of arbitrary dimensionality without the need for explicit dimensionality reduction. Whereas PMS requires a specification of a mapping on each sonification, sonification models require only few or no settings to be made. While for interpretation of PMS the (potentially complex) mapping must be known, in MBS the model itself provides the key for an interpretation of the sound with respect to the data. Sonification models can be furthermore designed according to a task and provide in their definition a means of interaction. As the sound thus is a consequence of the user's actions, it is less annoying. Sonification models often have also parameters which can be adjusted by the user in order to optimize the sound, but in comparison with PMS the number is greatly reduced and the meaning is easily understood as the parameters are tightly coupled to the sound generating process described by the model.

In Chapter 7 several new sonification models have been developed and exemplified. The Data Sonogram Sonification Model probes the dataset by means of an expanding excitation wave. The Particle Trajectory Sonification model uses Audification of particle trajectories in a data potential to encode properties of a dataset into sound in a holistic way. The McMC Sonification Model defines a process which explores features of a density function and summarizes the features in auditory streams. As a particular new element for the organization of sound in dynamic data display, the Auditory Information Buckets have been introduced, which provide a means of auditory zooming. The Principal Curve Sonification Model uses motion along a trajectory to define a dynamic soundscape. Properties of the curve and of the given dataset are summarized in the sonification. For the Data Crystallization Sonification Model and the Growing Neural Gas Sonification Model a dynamic growth process was used to determine the temporal evolution and acoustic properties of the sonification. Monitoring the proceeding changes of the dataset crystal or the growing neural net allows to value and compare local and global properties (in the case of data crystallization) or between initial and final properties of the achieved fit (in the case of the GNG model). In addition the GNG Sonification Model showed how hints on overfitting can be obtained by sound.

Chapter 9 presented two extensions to Parameter Mapping Sonifications. Firstly, the Auditory Legend was introduced as a means to explain the mapping by using sound. This avoids the change of media in accessing and interpreting auditory display. Secondly, the Multidimensional Perceptual Scaling Sonification was introduced as an approach that uses both MDS and PCA for preprocessing of high-dimensional datasets for PMS. The sonifications use an MDS index as time axis and the PCA components as acoustic attributes and thus give an auditory image of the dataset that allows to detect groupings from the sound.

Chapter 10 then demonstrated the utility of sonification in practical real-world domains:

- In Section 10.1, sonification was applied to data from psychotherapeutic verbatim protocols. The presented sonifications allow fast browsing of a 1 hour session within some seconds and further allow to perceive cycles of increased emotional brain activity of abstraction. For this purpose, also the Auditory Information Buckets are applied.

- In Section 10.2, EEG datasets from psycholinguistic experiments were sonified. The presented sonification technique of Spectral Mapping Sonification provides a fast overview of the data. The Distance Matrix Sonification allows to detect long-range correlations between electrodes. Differential Sonification was developed to compare datasets for a subject under different conditions.

- In Section 10.3, sonification was used to enhance the analysis of image stacks from Multi-Parameter Fluorescence Microscopy. The developed sonifications represent marker combination patterns by short musical motives which were optimized to compare, recognize and distinguish patterns.

Given a pool of available sonification techniques, the next step is to analyze how these techniques compare with visualization techniques, where they fail or where they are superior. Psychophysical experiments which measure performance of subjects under different conditions for identical problems have to be conducted.

Concerning the human-machine interface, the next steps are certainly to improve the modes of interaction with sonification models. This requires to realize real-time sonifications in order to allow an enhanced interactivity between the user and the sonification model. Audio-haptic interfaces that enable to make use of the rich expressiveness of the human hand for manipulation and exploration are seen as very promising and are subject to ongoing research. Finally, the author hopes that Model-Based Sonification inspires researchers to develop new types of sonifications and that the application of sonification models will lead to some standard models which assist the data mining work-flow.

Bibliography

[AC96] A. Axen and I. Choi. Investigating geometric data with sound. In *Proc. Int. Conf. on Auditory Display,* 1996. `http://www.santafe.edu/~icad/ICAD96/proc96/axen.htm`.

[Adr91] J. M. Adrien. The missing link: modal synthesis. In G. De Poli, A. Piccialli, and C. Roads, editors, *Representations of Musical Signals*. MIT Press, 1991.

[Ass60] American Standard Association. American standard acoustical terminology. American Standard Association, 1960.

[AV97] J. L. Alty and P. Vickers. The CAILTIN Auralization System: Hierarchical leitmotif design as a clue to program comprehension. In *Proc. Int. Conf. Auditory Display (ICAD '97)*. ICAD, ICAD, 1997. `http://www.icad.org/websiteV2.0/Conferences/ICAD97/abstr97.html`.

[Bar94] R. Bargar. Pattern and Reference in Auditory Display. In Kramer, editor, *Auditory Display*. Addison-Wesley, 1994.

[Bar97] S. Barrass. *Auditory Information Design*. PhD thesis, Australian National University, 1997.

[BCS$^+$01] M. Barra, T. Cillo, A. De Santis, T. Matlock, U. F. Petrillo, A. Negro, V. Scarano, and P. P. Maglio. Personal Webmelody: Customized sonification of web servers. In Zacharov, Hiipakka and Takala, editors, *Proc. of the 7th Int. Conf. on Auditory Display*, pages 1–9. ICAD, Laboratory of Acoustics and Audio Signal Processing, Helsinki University of Technology, 2001.

[BdH00] M. P. Bussemakers and A. de Haan. When it sounds like a duck and it looks like a dog... Auditory Icons vs. Earcons in Multimedia Environments. In P. R. Cook, editor, *Proc. Int. Conf. on Auditory Display*, pages 184–189. ICAD, Int. Community for Auditory Display, 2000. `http://www.icad.org/websiteV2.0/Conferences/ICAD2000/ICAD2000.html`.

[Bir99] N. Birbaumer. *Biologische Psychologie*. Springer, 4. edition, 1999.

[Bis95] C. M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, Oxford, 1995.

[Bla74] J. Blauert. *Räumliches Hören*. Hirzel Verlag, Stuttgart, 1974.

[Bly82] S. Bly. *Sound and Computer Information Presentation*. PhD thesis, University of California, Davis, 1982.

[BMGC98] T. L. Bonebright, N. E. Miner, T. E. Goldsmith and T. P. Caudell. Data collection and analysis techniques for evaluating the perceptual qualities of auditory stimuli. In *Proc. ICAD '98*. ICAD, British Computer Society, 1998. `http://www.ewic.org.uk/ewic/workshop/list.cfm`.

[Bon98] T. L. Bonebright. Perceptual structure of everyday sounds: A multidimensional scaling approach. In Hiipakka, Zacharov, and Takala, editors, *Proc. 7th Int. Conf. on Auditory Display*, pages 73–78. ICAD, Laboratory of Acoustics and Audio Signal Processing and the Telecommuniations Software and Multimedia Laboratory, Helsinki University, 1998.

[BPG00] M. Ballora, B. Pennycook and L. Glass. Sonification of Heart Rate Variability Data. In P. R. Cook, editor, *Proc. Int. Conf. on Auditory Display*, pages 184–189. ICAD, Int. Community for Auditory Display, 2000. `http://www.icad.org/websiteV2.0/Conferences/ICAD2000/ICAD2000.html`.

[BR92] J. D. Banfield and A. E. Raftery. Ice floe identification in satellite images using mathematical morphology and clustering about principal curves. *Journal of the American Statistical Association*, 87:7–16, 1992.

[Bre90] A. Bregman. *Auditory Scene Analysis: The Perceptual Organization of Sound*. MIT Press, Cambrigde Massachusetts, 1990.

[Bre96] S. A. Brewster. A sonically enhanced interface toolkit. In *Proc. Int Conf. Auditory Display (ICAD '96)*. ICAD, 1996. `http://www.santafe.edu/˜icad/ICAD96/proc96/INDEX.HTM`.

[Bre98] S. A. Brewster. Sonically-Enhanced Drag and Drop. In *Proc. of ICAD '98*. ICAD, ICAD, 1998. `http://www.icad.org/websiteV2.0/Conferences/ICAD98/icad98programme.html`.

[BRK96] S. A. Brewster, V.-P. Raty, and A. Kortekangas. Earcons as a method of providing navigational cues in a menu hierarchy. In *Proc. of HCI '96*, pages 167–183, Imperial College, London, UK, 1996. Springer.

[BS74] Bergmann and Schaefer. *Lehrbuch der Experimentalphysik*. Walter de Gruyter, Berlin, 1974.

[BS98] J. Bruske and G. Sommer. Intrinsic dimensionality estimation with optimally topology preserving maps. *IEEE Trans. of Pattern Analysis and Machine Intelligence*, 20(5):572–575, 1998.

[BWE94] S. A. Brewster, P. C. Wright, and A. D. N. Edwards. A detailed investigation into the effectiveness of earcons. In G. Kramer, editor, *Auditory Display*, pages 471–498. ICAD, Addison Wesley, 1994.

[CC94] D. R. Cox and M. A. A. Cox. *Multidimensional Scaling*. Chapman & Hall, London, 1994.

[Cho73] J. Chowning. The synthesis of complex spectra by means of frequency modulation. *Journal of the Audio Engineering Society*, 21(7):526–534, 1973.

[Cle94] W. S. Cleveland. *The Elements of Graphing Data*. Hobart Press, Summit, NJ, 1994.

[CM94] V. Cherkassky and F. Mulier. Self-organizing networks for nonparametric regression. In *From Statistics to Neural Networks*, pages 188–212, 1994.

[CMM74] J. M. Chambers, M. V. Mathews, and F. R. Moore. Auditory data inspection. Technical report, AT&T Bell Laboratories, 1974.

[CN95] C. Cruz-Neira. *Virtual Reality Based on Multiple Projection Screens: The CAVE and its Applications to Computational Science and Engineering*. PhD thesis, University of Illinois at Chicago, Chicago, 1995. `http://www.evl.uic.edu/research/vrdev.html`.

[Coo99] P. R. Cook, editor. *Music, Cognition, and Computerized Sound*. MIT Press, Cambridge, Massachusetts, 1999.

[DF84] P. Diaconis and D. Freedman. Asymptotics of graphical projection pursuit. *Annals of Statistics*, 12:793–815, 1984.

[dTSS86] S. H. C. du Toit, A. G. W. Steyn, and R. H. Stumpf. *Graphical Exploratory Data Analysis*. Springer-Verlag, New York, 1986.

[ea92] K. W. Brodlie et al., editor. *Scientific Visualization*. Springer-Verlag, Berlin, 1992.

[Eck98] G. Eckel. A spatial auditory display for the cyberstage. In *Proc. ICAD '98*. British Computer Society, 1998. `http://www.icad.org/websiteV2.0/Conferences/ICAD98/icad98programme.html`.

[EOS92] E. Erwin, K. Obermayer, and K. Schulten. Self-organizing maps: ordering, convergence properties and energy functions. *Biological Cybernetics*, 67:47–55, 1992.

[Fis36] R. A. Fisher. The use of multiple measurements in taxonomic problems. *Annual Eugenics*, 7(Part II):179–188, 1936.

[Fis99] R. A. Fisher. UCI repository of maschine learning databases. `ftp://ftp.ics.uci.edu/pub/machine-learning-databases/iris`, 1999.

[FK94] T. Fitch and G. Kramer. Sonifying the body electric: Superiority of an auditory over a visual display in a complex multivariate system. In Kramer, editor, *Auditory Display*. Addison-Wesley, 1994.

[FM98] M. Fernström and C. McNamara. After direct manipulation - Direct Sonification. In *Proc. ICAD '98*. British Computer Society, 1998.

[FO71] K. Fukunaga and D. R. Olsen. An algorithm for finding intrinsic dimensionality of data. *IEEE Trans. Computers*, 20(2):176–183, 1971.

[For46] T. W. Forbes. Auditory signals for instrument flying. *J. Aeronautical Soc.*, pages 255–258, May 1946.

[Fri95] B. Fritzke. A growing neural gas network learns topologies. In G. Tesauro, D. Touretzky, and T. Leen, editors, *Advances in Neural Information Processing Systems*, volume 7, pages 625–632. The MIT Press, 1995.

[FT74] J. H. Friedman and J. W. Tukey. A projection pursuit algorithm for exploratory data analysis. *IEEE Transactions on Computers*, 23:881–890, 1974.

[Gab47] D. Gabor. Acoustical quanta and the theory of hearing. *Nature*, 159(4044):591–594, 1947.

[Gar98] William G. Gardner. Reverberation algorithms. In M. Kahrs and K. Brandenburg, editors, *Applications of Digital Signal Processing to Audio and Acoustics*, chapter 3, pages 85–130. Kluwer Academic, 1998.

[Gav89] W. W. Gaver. The SonicFinder: An interface that uses Auditory Icons. *Human-Computer Interaction*, 4:67–94, 1989.

[Gav93a] W. W. Gaver. How do we hear in the world? Explorations in ecological acoustics. *Ecological Psychology*, 5(4):285–313, 1993.

[Gav93b] W. W. Gaver. What in the world do we hear? An ecological approach to auditory source perception. *Ecological Psychology*, 5(1):1–29, 1993.

[Gav94] W. W. Gaver. Using and creating auditory icons. In G. Kramer, editor, *Auditory Display*, pages 417–446. ICAD, Addison-Wesley, 1994.

[GBC81] C. Gawbay, G. Brugal, and C. Choquet. Application of colored image analysis to bone marrow cell recognition. *Analyt. Quant. Cytol.*, 4:272, 1981.

[GCSR95] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin. *Bayesian Data Analysis*. Chapman & Hall, 1995.

[GRS96] W. R. Gilks, S. Richardson, and D. J. Spiegelhalter. *Markov Chain Monte Carlo in Practice*. Chapman & Hall, 1996.

[GS90] W. W. Gaver and R. Smith. Auditory icons in large-scale collaborative environments. In R. Baecker, J. Grudin, W. Buxton, and S. Greenberg, editors, *Readings in Human-Computer Interaction: Toward the Year 2000*, pages 564–569, NY, 1990. Morgan-Kaufman.

[GSO91] W. W. Gaver, R. B. Smith, and T. O'Shea. Effective sounds in complex systems: the aRKola simulation. In *Proc. of CHI*. New York: ACM, 1991.

[Has99] R. Haschke. Oszillationen in rekurrenten neuronalen Netzen. Master's thesis, Bielefeld University, Technical Faculty, 1999.

[Hay94] C. Hayward. Listening to the earth sing. In G. Kramer, editor, *Auditory Display*, pages 369–404, 1994.

[Her00] D. J. Hermes. Synthesis of the sounds produced by rolling balls. Technical report, IPO, Center for User-System Interaction, 2000.

[HHR01] T. Hermann, M. H. Hansen, and H. Ritter. Sonification of Markov chain Monte Carlo simulations. In Hiipakka, Zacharov, Takala, editors, *Proc. of 7th Int. Conf. on Auditory Display*, pages 208–216, Helsinki University of Technology, 2001. ICAD, Laboratory of Acoustics and Audio Signal Processing and the Telecommunications Software and Multimedia Laboratory. `http://www.acoustics.hut.fi/icad2001/proceedings/index.htm`.

[HKR02] T. Hermann, J. Krause, and H. Ritter. Real-time control of sonification models with an audio-haptic interface. In *Proc. of the Int. Conf. on Auditory Display*, pages 81–86. Int. Community for Auditory Display, 2002. submitted.

[HMR00] T. Hermann, P. Meinicke, and H. Ritter. Principal curve sonification. In P. R Cook, editor, *Proc. of the Int. Conf. on Auditory Display*, pages 81–86. Int. Community for Auditory Display, 2000.

[HNR02] T. Hermann, C. Nölker, and H. Ritter. Hand postures for sonification control. In Wachsmuth and Sowa, editors, *Proc. Int. Gesture Workshop GW2001*. Springer, 2002. accepted.

[HNSR00] T. Hermann, T. W. Nattkemper, W. Schubert, and H. Ritter. Sonification of multichannel image data. In V. Falavar, editor, *Proc. of the Mathematical and Engineering Techniques in Medical and Biological Sciences (METMBS 2000)*, pages 745–750. CSREA Press, 2000.

[HR99] T. Hermann and H. Ritter. Listen to your data: Model-Based Sonification for data analysis. In G. E. Lasker, editor, *Advances in intelligent computing and multimedia systems, Baden-Baden, Germany*, pages 189–194. Int. Inst. for Advanced Studies in System research and cybernetics, 1999.

[HR01] M. H. Hansen and B. Rubin. Babble online: Applying statistics and design to sonify the internet. In Zacharov, Hiipakka, and Takala, editors, *Proc. of the 7th Int. Conf. on Auditory Display*, pages 10–15. ICAD, Laboratory of Acoustics and Audio Signal Processing, Helsinki University of Technology, 2001.

[HS89] T. Hastie and W. Stuetzle. Principal curves. *Journal of the American Statistical Association*, 84:502–516, 1989.

[Hub85] P. J. Huber. Projection pursuit (with discussion). *Annals of Statistics*, 13:435–525, 1985.

[Huo99] J. Huopaniemi. *Virtual acoustics and 3-D sound in multimedia signal processing*. PhD thesis, Helsinki University of Technology, Faculty of Electrical and Communications Engineering, Laboratory of Acoustics and Audio Signal Processing, 1999. `http://www.acoustics.hut.fi/˜ruba/pubs/\#theses`.

[ICA] ICAD. International community of auditory display. `http://www.icad.org`.

[JD88] A. K. Jain and R. C. Dubes. *Algorithms for Clustering Data*. Prentice-Hall, Englewood Cliffs, NJ, 1988.

[Jol86] I. T. Jolliffe. *Principal Component Analysis*. Springer, New York, 1986.

[KB97] H. Klock and J. Buhmann. Data visualization by multimensional scaling: A deterministic annealing approach. Technical report, Institute of Computer Science, University of Bonn, 1997.

[KB98] M. Kahrs and K. Brandenburg, editors. *Applications of Digital Signal Processing to Audio and Acoustics*. Kluver Academic Publishers, 1998.

[KE91] G. Kramer and S. Ellison. Audification: The use of sound to display multivariate data. In *The Proceedings of the International Computer Music Conference*, pages 214–221, San Fransisco, 1991. ICMA, CA:ICMA.

[KG97] M. W. Krueger and D. Gilden. Knowwhere:tm an audio/spatial interface for blind people. In *Proc. Int Conf. Auditory Display (ICAD '97)*. ICAD, ICAD, 1997. `http://www.icad.org/websiteV2.0/Conferences/ICAD97/abstr97.html`.

[KKLZ00] B. Kégl, A. Krzyzak, T. Linder, and K. Zeger. Learning and design of principal curves. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 22(3):281–297, 2000.

[Koh82] T. Kohonen. Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43:56–69, 1982.

[Kra92] G. Kramer. Sonification system using auditory beacons as references for comparison and orientation in data. United States Patent 5,371,854, 9 1992.

[Kra94a] G. Kramer, editor. *Auditory Display - Sonification, Audification, and Auditory Interfaces*. Addison-Wesley, 1994.

[Kra94b] G. Kramer. An introduction to auditory display. In G. Kramer, editor, *Auditory Display*, pages 1–79. ICAD, Addison-Wesley, 1994.

[Kra94c] G. Kramer. Some organizing principles for representing data with sound. In G. Kramer, editor, *Auditory Display*, pages 185–222. Addison-Wesley, 1994.

[KWB+99] G. Kramer, B. Walker, T. Bonebright, P. Cook, J. Flowers, N. Miner, and J. Neuhoff. Sonification report: Status of the field and research agenda. Technical report, International Community for Auditory Display, 1999. `http://www.icad.org/websiteV2.0/References/nsf.html`.

[Lar98] J. Laroche. Time and pitch scale modification of audio signals. In M. Kahrs and K. Brandenburg, editors, *Applications of Digital Signal Processing to Audio and Acoustics*, chapter 7, pages 279–308. Kluwer Academic, 1998.

[LB98] G. Leplâtre and S. A. Brewster. An investigation of using music to provide navigation cues. In *Proc. ICAD '98*. British Computer Society, 1998. `http://www.ewic.org.uk/ewic/workshop/list.cfm`.

[LB00] G. Leplâtre and S. A. Brewster. Designing non-speech sounds to support navigation in mobile phone menus. In P. R. Cook, editor, *Proc. Int. Conf. on Auditory Display*, pages 190–199. ICAD, Int. Community for Auditory Display, 2000. `http://www.icad.org/websiteV2.0/Conferences/ICAD2000/ICAD2000.html`.

[LHJZU97] S. K. Lodha, T. Heppe, A. Joseph, and B. Zane-Ulman. MUSE: A musical data sonification toolkit. In *Proc. Int Conf. Auditory Display (ICAD '97)*. ICAD, 1997. `http://www.icad.org/websiteV2.0/Conferences/ICAD97/abstr97.html`.

[LO88] J. S. Lim and A. V. Oppenheim. *Advanced Topics in Signal Processing*. Prentice Hall, New Jersey, 1988.

[LT94] M. LeBlanc and R. Tibshirani. Adaptive principal surfaces. *Journal of the American Statistical Association*, 89(425):53–65, 1994.

[Mar98] T. M. Martinetz. Competitive hebbian learning rule forms perfectly topology preserving maps. In *Proc. of the ICANN'98*, pages 427–434, Amsterdam, 1998. Springer.

[Mas98] D. C. Massie. Wavetable sampling synthesis. In M. Kahrs and K. Brandenburg, editors, *Applications of Digital Signal Processing to Audio and Acoustics*, chapter 8, pages 311–342. Kluwer Academic, 1998.

[MB99] E. Mergenthaler and W. Bucci. Linking verbal and non-verbal representations: Computer analysis of referential activity. *Brit. Journ. of Medical Psychology*, 72:339–354, 1999.

[Mei00] P. Meinicke. *Unsupervised Learning in a Generalized Regression Framework*. PhD thesis, Bielefeld University, Germany, 2000.

[Mei01] P. B. L. Meijer. The voice – seeing with sound. `http://ourworld.compuserve.com/homepages/Peter_Meijer/voice.htm`, 2001.

[Mer96] E. Mergenthaler. Emotion-abstraction patterns in verbatim protocols: A new way of describing psychotherapeutic processes. *Journ. of Consulting and Clinical Psychology*, 64(6):1306–1315, 1996.

[MI68] P. M. Morse and K. U. Ingard. *Theoretical Acoustics*. McGraw-Hill, New York, 1968.

[MKBC94] G. Mayer-Kress, R. Bargar, and I. Choi. Musical structures in data from chaotic attractors. In G. Kramer, editor, *Auditory Display*, pages 341–367. Addison-Wesley, 1994.

[MMBG94] A. L. Papp III M. M. Blattner and E. P. Glinert. Sonic enhancement of two-dimensional graphics displays. In G. Kramer, editor, *Auditory Display*, pages 447–470. ICAD, Addison-Wesley, 1994.

[Moo90] F. R. Moore. *Elements of Computer Music*. Prentice Hall, 1990.

[MR94a] T. M. Madhyastha and D. A. Reed. A framework for sonification design. In G. Kramer, editor, *Auditory Display*, pages 267–289. ICAD, Addison-Wesley, 1994.

[MR94b] K. McCabe and A. Rangwalla. Auditory display of computational fluid dynamics data. In G. Kramer, editor, *Auditory Display*, pages 327–340, 1994.

[MR97] A. C. G. Martins and R. M. Rangayyan. Experimental evaluation of auditory display and sonification of textured images. In *Proc. Int. Conf. Auditory Display (ICAD '97)*. ICAD, ICAD, 1997. `http://www.icad.org/websiteV2.0/Conferences/ICAD97/abstr97.html`.

[MRP$^+$96] A. C. G. Martins, R. M. Rangayyan, L. A. Portela, E. Amaro, and R. A. Ruschioni. Auditory display and sonification of textured image. In *Proc. Int. Conf. Auditory Display (ICAD '96)*. ICAD, ICAD, 1996. http://www.santafe.edu/~icad/ICAD96/proc96/INDEX.HTM.

[MS91] T. M. Martinetz and K. J. Schulten. *A "neural-gas" network learns topologies*, volume 1, pages 397–402. North-Holland, Amsterdam, 1991.

[MWR99] H. M. Müller, S. Weiss, and P. Rapelsberger. EEG coherence analysis of auditory sentence processing. In *Quantitative and Topological EEG and MEG analysis*. Druckhaus Mayer Verlag GmbH Jena Erlangen, 1999.

[Nat01] T. W. Nattkemper. *A Neural Network-Based System for High Throughput Fluorescence Micrograph Evaluation*. PhD thesis, Faculty of Technology, Bielefeld University, Bielefeld, 2 2001.

[NL99] E. Niedermeyer and F. H. Lopes da Silvia, editors. *Electroencephalography: Basic principles, clinical applications, and related fields*. Lippincott Williams & Wilkins, Philadelphia, 4 edition, 1999.

[OW89] A. V. Oppenheim and A. S. Willsky. *Signale und Systeme*. VCH Verlagsgesellschaft, Weinheim, 1989.

[Pat82] R. D. Patterson. Guidelines for auditory warning systems on civil aircraft. Technical report, Civil Aviation Authority, London, 1982.

[PF54] I. Pollack and L. Ficks. Information of elementary multidimensional auditory display. *J. Acous. Soc. Amer.*, 26:155–158, 1954.

[Pol] Polhemus Corporation. 3d motion tracker. http://www.polhemus.com/home.htm.

[Pul02] Pulsoxymeter. http://www.m-ww.de/enzyklopaedie/medizingeraete/pulsoximeter.html, 2002. Device for auditory monitoring of pulse and blood oxygen level.

[RGF90] K. Rose, E. Gurewitz, and G. C. Fox. Statistical mechanics and phase transitions in clustering. *Physical Review Letters*, 65(8):945–948, 1990.

[Rip96] B. D. Ripley. *Pattern Pecognition and Neural Networks*. Cambridge University Press, 1996.

[Rit00] H. Ritter. The Graphical Simulation Toolkit Neo/NST. http://www.techfak.uni-bielefeld.de/ags/ni/projects/simulation_and_visual/neo/neo_e.html, 2000.

[RM69] J.-C. Risset and M. V. Matthews. Analysis of musical instrument tones. *Physics Today*, 22(2), 1969.

[RMS92] H. Ritter, T. Martinetz, and K. Schulten. *Neural Computation and Self-Organizing Maps. An Introduction*. Addison-Wesley, Reading, MA, 1992.

[Roa85a] C. Roads. *Granular Synthesis of Sound*, chapter 10, pages 145–159. MIT Press, Cambridge, Massachusetts, 1985.

[Roa85b] C. Roads. *A Tutorial on Nonlinear Distortion or Waveshaping Synthesis*, chapter 7, pages 83–94. MIT Press, Cambridge, Massachusetts, 1985.

[Roa96] C. Roads. *The Computer Music Tutorial*. MIT Press, Cambridge , Massachusetts, 1996.

[RPAP98] P. Roth, L. Petrucci, A. Assimacopoulos, and T. Pun. AB-Web: Active audio browser for visually impaired and blind users. In *Proc. ICAD '98*. British Computer Society, 1998. `http://www.ewic.org.uk/ewic/workshop/list.cfm`.

[RS85] C. Roads and J. Strawn, editors. *Foundations of Computer Music*. MIT Press, Cambridge, Massachusetts, 1985.

[Rub98] B. U. Rubin. Audible information design in the new york city subway system: A case study. In *Proc. ICAD '98*. British Computer Society, 1998. `http://www.icad.org/websiteV2.0/Conferences/ICAD98/icad98programme.html`.

[Sam69] J. W. Jr Sammon. A nonlinear mapping for data structure analysis. *IEEE Transactions on Computer*, 18:401–409, 1969.

[SC91] C. Scaletti and A. Craig. Using sound to extract meaning from complex data. In E. J. Farrell, editor, *Extracting Meaning from Complex Data: Processing, Display, Interaction II*, pages 207–219, 1991.

[Sca94] C. Scaletti. Sound synthesis algorithms for auditory data representations. In G. Kramer, editor, *Auditory Display*. Addison-Wesley, 1994.

[SCB98] D. F. Swayne, D. Cook, and A. Buja. Xgobi: Interactive dynamic data visualization in the x window system. *Journal of Computational and Graphical Statistics*, 7(1), 1998.

[Sch92] W. Schubert. Antigenic determinants of t-lymphozyte $\alpha\beta$ receptor and other leucocyte surface proteins as differential markers of skeletal muscle regeneration: detection of spatially and timely restricted patterns by MAM microscopy. *Eur. J. Cell Biol.*, 58:395–410, 1992.

[Sch97] W. Schubert. Molecular semiotic structures in the cellular immune system: key to dynamics and spatial patterning? In W. Zimmermann J. Parisi, S.C. Mueller, editor, *A perspective look at nonlinear media in physics, chemistry and biology, Lecture notes in physics*. Springer, Berlin, 1997.

[Sco92] D. W. Scott. *Multivariate Density Estimation - Theory, Practice, and Visualization*. Wiley & Sons, 1992.

[SF97] S. Saue and O. Kr. Fjeld. A platform for audiovisual seismic interpretation. In *Proc. Int Conf. Auditory Display (ICAD '97)*. ICAD, ICAD, 1997. `http://www.icad.org/websiteV2.0/Conferences/ICAD97/abstr97.html`.

[Sil86] B. W. Silverman. *Density Estimation for Statistics and Data Analysis*. Chapman and Hall, 1986.

[Slo85] A. Sloman. Afterthoughts on analogical representation. In R. Brachman and H. Levesque, editors, *Readings in Knowledge Representation*. CA: Morgan Kaufmann, Los Altos, 1985.

[Smi91] S. Smith. An auditory display for exploring visualization of multidimensional data. In G. Grinstein and J. Encarnacao, editors, *Workstations for Experiment*. Springer Verlag, Berlin, 1991.

[Smi92] J. O. Smith. Physical modeling using digital waveguides. *Computer Music Journal, Vol. 16, no. 4, pp. 74-91*, 1992.

[Smi97] J. O. Smith. Discrete-time modeling of acoustic systems. *CCRMA Stanford University, Online-Course*, 1997. `http://www-ccrma.stanford.edu/~jos/lumped/lumped.html`.

[Smi98] J. O. Smith. Principles of digital waveguide models of musical instruments. In M. Kahrs and K. Brandenburg, editors, *Applications of Digital Signal Processing to Audio and Acoustics*, chapter 10, pages 417–466. Kluwer Academic, 1998.

[SMS89] A. J. Smola, S. Mika, and B. Schölkopf. Quantization functionals and regularized principal manifolds. *NeuroCOLT2 27150*, 1989.

[Spe61] S. D. Speeth. Seismometer sounds. *J. Acous. Soc. Amer.*, 33:909–916, 1961.

[Tho01] T. Thomas. Sonifikation von Börsendaten. Master's thesis, Bielefeld University, 2001.

[Tie94] L. Tierney. Markov chains for exploring posterior distributions (with discussion). *the Annals of Statistics*, 22(2):1701–1727, 1994.

[Ton91] H. Tong. *Non-linear time series: a dynamical systems approach*. Oxford University Press, 1991. sunspot dataset at `http://www-personal.buseco.monash.edu.au/~hyndman/TSDL/data/sunspot.dat`.

[Tuk70] J. W. Tukey. *Exploratory Data Analysis*. Addison-Wesley, 1970. Band 1, preliminary edition.

[Tuk77] J. W. Tukey. *Exploratory Data Analysis*. Addison-Wesley, 1977.

[Ulr67] R. J. Ulrick. *Principles of Underwater Sound*. New York: McGraw-Hill, 1967.

[VD95] P. J. Verveer and R. P. W. Duin. An evaluation of intrinsic dmensionality estimators. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 17(1), 1995.

[vdDP98] K. van den Doel and D. K. Pai. The sounds of physical shapes. *Presence*, 7(4):382–395, 1998.

[vH54] H. L. F. von Helmholtz. *On the Sensations of Tone as a Physiological Basis for the Theory of Music*. Dover, New York, NY, 1954.

[WC97] J. R. Weinstein and P. R. Cook. Faust: A framework for algorithm understanding and sonification testing. In *Proc. Int Conf. Auditory Display (ICAD '97)*. ICAD, ICAD, 1997. `http://www.icad.org/websiteV2.0/Conferences/ICAD97/abstr97.html`.

[Wil94] S. M. Williams. Perceptual principles in sound grouping. In G. Kramer, editor, *Auditory Display*, pages 95–125. Addison-Wesley, 1994.

[WK96] B. N. Walker and G. Kramer. Mappings and metaphors in auditory displays: An experimental assessment. In *Proc. Int Conf. Auditory Display (ICAD '96)*. ICAD, ICAD, 1996. `http://www.santafe.edu/˜icad/ICAD96/proc96/INDEX.HTM`.

[WL96] C. M. Wilson and S. K. Lodha. Listen: A data sonification toolkit. In *Proc. Int Conf. Auditory Display (ICAD '96)*. ICAD, ICAD, 1996. `http://www.santafe.edu/˜icad/ICAD96/proc96/INDEX.HTM`.

[Wol90] W. H. Wolberg. UCI repository of maschine learning databases. `ftp://ftp.ics.uci.edu/pub/machine-learning-databases/breast-cancer-wisconsin`, 1990.

[WR96] S. Weiss and P. Rapelsberger. EEG coherence within the 13-18 hz band as a correlate of a distinct lexical organisation of concrete and abstract nouns in humans. *Neuroscience letters*, (209):17–20, 1996.

[Wri00] K. Wrightson. An introduction to acoustic ecology. *Soundscape - The Jounal of Acoustic Ecology*, 1(1):10–13, 2000.

[ZF99] E. Zwicker and H. Fastl. *Psychoacoustics*. Springer, Berlin, 2 edition, 1999.

[Zwi82] E. Zwicker. *Psychoakustik*. Springer-Verlag, Berlin, 1982.