# Fast Design of Audio Processing Algorithms by Interactive Parameter Exploration

Zijian Xu, MSc Thesis
Signal Processing Systems group, Eindhoven University of Technology
e-mail: z.xu@student.tue.nl

***Abstract*—Current hearing aid algorithms contain more than a hundred tuning parameters, which bring challenges for users to find perceptually pleasing settings in such huge design space. In this project, an interactive approach is designed which allows the customer to change parameter values on-the-fly while he listens to the effect of the output signal. The search space is limited by projecting the full parameter space onto a 2-dimensional exploration pad. In this way, the user is able to change the algorithm settings by moving his finger (or mouse) over the tablet. An efficient many-to-2-dimensional parameter projection function has been designed in this project. In addition, user preference has been learned based on a probit model for binary observations in Gaussian processes (GPs) as well to simplify the tuning process.**

***Index Terms*—Bayesian optimization, Gaussian process, Parameter compression, Adaptive resolution, Hearing aid**

## DEFINITIONS

### A. Abbreviations

| | |
|---|---|
| CDF | Cumulative Distribution Function |
| GP | Gaussian Process |
| IPE | Interactive Parameter Exploration |
| LLE | Locally Linear Embedding |
| MDS | Multidimensional Scaling |
| PCA | Principal Components Analysis |
| PDF | Probability Density Function |
| UI | User Interface |

### B. Conventions

| | |
|---|---|
| $\mathcal{D}$ | Observation pair sets |
| $k$ | Kernel function of covariance |
| $p$ | Exponential scale factor |
| $\boldsymbol{x}$ | Sample sets |
| $\theta, \boldsymbol{\theta}$ | Hyperparameters and hyperparameter vector |
| $\boldsymbol{\chi}$ | Sample space |
| $\Sigma$ | Covariance matrix |
| $\phi$ | Probability Density Function |
| $\Phi$ | Cumulative Distribution Function |

## I. INTRODUCTION

HEARING impairment is the most frequent sensory deficit in human populations, affecting more than 250 million people in the world [1]. A relatively small proportion of adults with hearing impairment seek help for their hearing problems and use hearing aids. Several studies have shown that a large proportion of people who could benefit from hearing aids do not have them [2]. Although the adoption rate (a percent of people with admitted hearing loss who own hearing aids) in the United States reached 24.6% in 2008 [3], there is still high market potential for hearing aids. In another aspect, some defects of hearing aids (e.g. volume adjustment [4]) dissatisfy the users and are needed to be solved. Also, current hearing aid algorithms contain more than a hundred tuning parameters. As a user, the patient does not expect to tune hundreds of parameters manually. In this thesis, an Interactive Parameter Exploration (IPE) system is developed to help design of hearing aid algorithms which is able to analyze the users' preference and make preference recommendation according to hearing defect characteristic of every user. The block diagram of the hearing aid system is shown in Fig. 1. In this diagram, we can see that Oracle/user will hear results provided by DSP system. If he is not satisfied with the results, parameters can be set during interaction with UI (user interface). At here, Bayesian optimization based on Gaussian Process (GP) for parameter recommendation has been implemented and a dimensionality reduction algorithm has been used for helping system to find the mapping for 2D to high-dimensional conversion. Also, a zooming module is introduced, which can provide high resolution for the area where users are more likely finding their best settings.Then in adaptation procedure, his preference will be learned and DSP settings will be changed. In this report, the algorithms of interactive parameter exploration are going to be discussed and the work is divided into four sections – Bayesian optimization, parameter compression, adaptive resolution and performance evaluation. The structure of the report is as follows. In the next section we briefly review the method of Bayesian optimization by pairwise comparison in GP and consider the extension of this approach in audio processing system. Different algorithms for parameter compression related to this work are considered in Section III. Section IV discusses adaptive resolution algorithm, and then an introduction for performance evaluation as well as the results with different test cases are introduced in Section V. Section VI concludes and future works are also discussed in this section.

As shown in Fig. 2, our work provides a friendly GUI for users. Users can either find their best settings by moving fingers on the tablet or using a pairwise comparison way to get sample from system recommendation. During tuning procedure, if there are some samples they are interested in, they can click "Record" button and save the positions of the samples. Also, whether choosing zooming mode can be

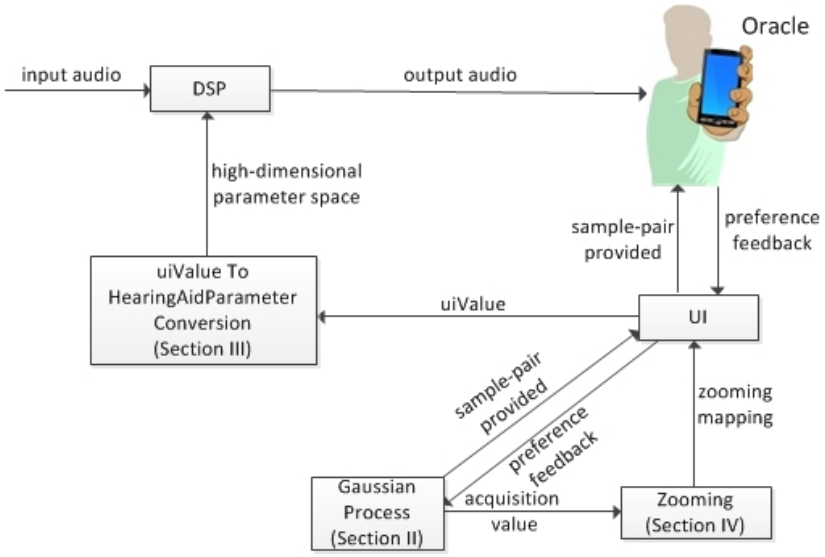



Fig. 1: Framework for the hearing aid system. More explanation can be found in Section I.

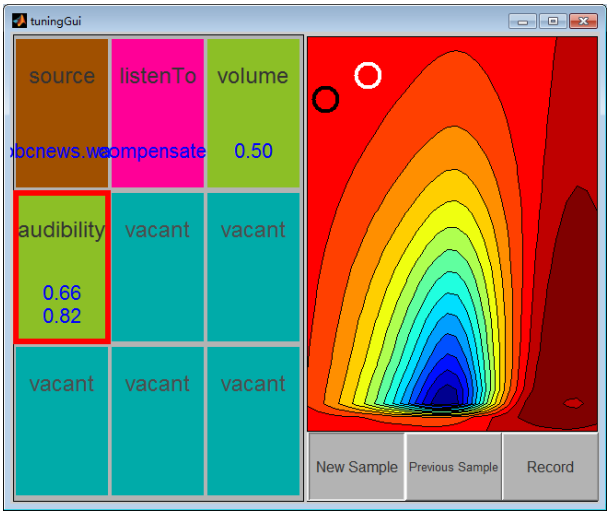



Fig. 2: GUI for interactive parameter exploration. In the GUI, users can find their preference settings manually or by clicking buttons underneath to get the samples provided by Gaussian Process. Also, adaptive resolution algorithm has been embedded in the system and users can choose "zooming" mode.

determined by users.

## II. Bayesian Optimization for Parameter Exploration

This section provides an introduction of parameter exploration based on Bayesian optimization. The reason for introducing this is users are always hard to find their preference on a vast 2D tablet without any clue. Simply choosing samples randomly just wastes their patience and time. We want a algorithm which can find users' best configurations quickly (the process should access the best settings quickly during iterations). Before the detailed discussion, the algorithms of Bayesian optimization in this work as well as a demo are provided.

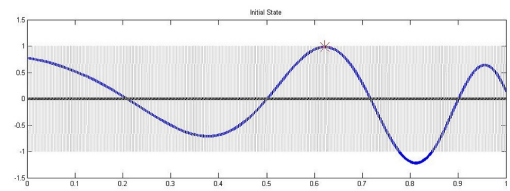
### A. Algorithms for Bayesian Optimization

A simple one dimensional example is provided in Fig. 3, of which the target function is $y = 0.7732 - \sin(x) - \frac{x}{3} - \sin(12x^2)$ as denoted as solid blue curve. The star is the target location where the function is maximal we want to reach, and the red dot indicates the best value we have acquired. The solid line connects the comparison pair in this iteration while the dashed line indicates all past comparisons. By an acquisition function, the best sample (maximal location of the acquisition functions) for the next iteration has been denoted. The shadow represents the variance around the mean curve (black) of a Gaussian Process and the variance of a sample represents the uncertainty of the sample. At the beginning of iterations, mean is zero and variance is one in the whole interval. It should be mentioned that in the initialization stage, the acquisition function cannot be evaluated. From the figure we can see that the variance decreases during iterations and this algorithms can reached the maximum value quickly.

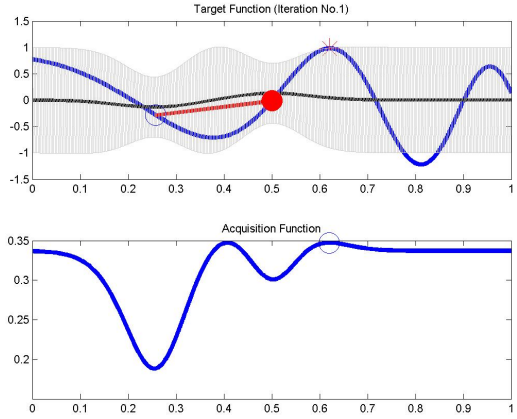
### B. Gaussian process

This subsection presents an introduction of GP mostly based on [5]. A Gaussian process is a natural way of defining prior distributions over functions of one or more input variables [6]. It has been widely used in many areas such as kriging, ARMA (autoregressive moving average) models, Kalman filters, and radial basis function networks [7]. With $f : \boldsymbol{\chi} \to \mathbb{R}$, GP can be defined by the property that any finite set of $N$ points $\{\mathbf{x}_n \in \boldsymbol{\chi}\}_{n=1}^{N}$ induces a multivariate Gaussian distribution on $\mathbb{R}^N$ [8]. It can be completely specified by its mean function, $m$ and covariance kernel (used to calculate the covariance between samples), $k$:

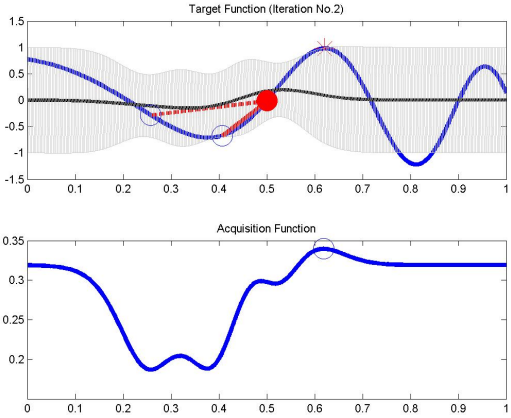
$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')), \tag{1}$$



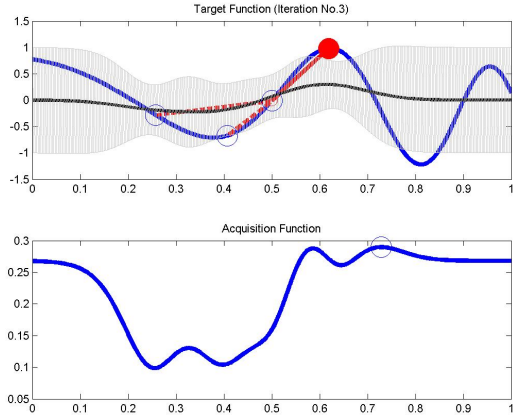

(a) Initialization




(b) First iteration




(c) Second iteration




(d) Third iteration

Fig. 3: 1-D example in the first 3 iterations. Further explanation is provided in Section II-A.

where $\mathbf{x}'$ is any samples in the N-point set. Since we do not have any prior knowledge about the position of the optimum value, we choose the prior mean as the zero function $m(\mathbf{x}) = 0$. Now the definition of the covariance kernel $k$ is left for determination. By adding hyperparameters $\theta$ in isotropic model, the covariance function becomes

$$k(\mathbf{x}, \mathbf{x}') = \exp(-\frac{1}{2\theta^2}\|\mathbf{x} - \mathbf{x}'\|^2). \quad (2)$$

The squared exponential function is the degenerate isotropic model for $\theta = 1$. For anisotropic models, the kernel function can be represented as

$$k(\mathbf{x}, \mathbf{x}') = \exp(-\frac{1}{2}(\mathbf{x} - \mathbf{x}')^T \mathrm{diag}(\boldsymbol{\theta})^{-2}(\mathbf{x} - \mathbf{x}')), \quad (3)$$

where $\mathrm{diag}(\boldsymbol{\theta})$ is the matrix which only contains diagonal elements in matrix $\boldsymbol{\theta}$.

### C. Probit model for binary observations

In this work, pairwise comparison is chosen for GP evaluation. The reason for choosing as such rather than absolute ratings is that pairwise preferences are often more accurate than absolute ratings for human judgment [5]. In addition, the judgment of a person is not always static but may fluctuate even for two similar inputs at different times. So the users' decisions should be discussed under Gaussian distribution. Bonilla et al. [9], Chu and Ghahramani [10], Brochu et al. [11] and Nielsen et al. [12] did some similar work related to GP preference elicitation by pairwise comparison. In [12], Nielsen et al. proposed a specific sparse extension of the classical pairwise likelihood using the pseudo-input formulation. It is useful for problems with large amount comparison pairs, while our work is not the case. The idea of this paper is similar with the paper written by Chu and Ghahramani [10]. Bonilla et al. [9] and Brochu et al. [11] provide two main references in this work, while Bonilla et al. [9] focus on preference recommendation based on prior knowledge from different sources. In another paper, Nielsen [13] proposes another approach which asks the user to express to what extent he prefers the dominating option on a 3 step scale. Then the likelihood function can be acquired by sigmoid functions according to the feedback of the user. But his work is a kind of absolute ratings algorithm and we tend to use GP method based on the work of Bonilla.

Based on the set $\{\mathbf{x}_n \in \boldsymbol{\chi}\}_{n=1}^N$, a set of $M$ observed

pairwise preference relations on the instances is denoted as

$$\mathcal{D} = \{\mathbf{r}_k \succ \mathbf{c}_k : k = 1, \ldots, M\}, \tag{4}$$

where $\mathbf{r}_k, \mathbf{c}_k \in \boldsymbol{\chi}$, and $\mathbf{r}_k \succ \mathbf{c}_k$ means the instance $\mathbf{r}_k$ is preferred to $\mathbf{c}_k$ [10]. The feedback given by user can be written as:

$$g(\mathbf{r}_k, \mathbf{c}_k) = \begin{cases} 1, & \mathbf{r}_k \succ \mathbf{c}_k \\ 0, & \mathbf{r}_k \prec \mathbf{c}_k \end{cases}. \tag{5}$$

It is convinced that users may sway their preferences, especially for two samples with similar performance. So we can take such a noise term into consideration, the probability that item $\mathbf{r}_k$ is preferred to item $\mathbf{c}_k$ is given by:

$$\begin{aligned} P\big(\mathbf{r}_k \succ \mathbf{c}_k | f(\mathbf{r}_k), f(\mathbf{c}_k)\big) &= P\big(g(\mathbf{r}_k, \mathbf{c}_k) = 1 | f(\mathbf{r}_k), f(\mathbf{c}_k)\big) \\ &= \Phi\Big(\frac{f(\mathbf{r}_k) - f(\mathbf{c}_k)}{\sigma_{\text{noise}}}\Big), \end{aligned} \tag{6}$$

where the noise term is Gaussian: $\varepsilon \sim \mathcal{N}(0, \sigma^2_{\text{noise}})$ and $\Phi(\cdot)$ is the CDF of the standard normal distribution (proof is provided in Appendix). Then we have the likelihood function

$$P(\mathcal{D}|\mathbf{f}) = \prod_{k=1}^{M} P\big(\mathbf{r}_k \succ \mathbf{c}_k | f(\mathbf{r}_k), f(\mathbf{c}_k)\big), \tag{7}$$

where $\mathbf{f} = \{f(\mathbf{x}_1), f(\mathbf{x}_2), \ldots, f(\mathbf{x}_n)\}$, $n$ is the number of samples which occur in the pairwise comparison list.

Now we can begin to deduce the posterior distribution of the latent utility function given the discrete data (Detailed proof can be found in [14]). According to Bayes' theorem,

$$P(\mathbf{f}|\mathcal{D}) \propto P(\mathbf{f}) \prod_{k=1}^{M} P\big(\mathbf{r}_k \succ \mathbf{c}_k | f(\mathbf{r}_k), f(\mathbf{c}_k)\big). \tag{8}$$

In equation(8), $\mathbf{f} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$ and

$$\boldsymbol{\Sigma} = \begin{bmatrix} k(\mathbf{x}_1, \mathbf{x}_1) & \ldots & k(\mathbf{x}_1, \mathbf{x}_n) \\ \vdots & \ddots & \vdots \\ k(\mathbf{x}_n, \mathbf{x}_1) & \ldots & k(\mathbf{x}_n, \mathbf{x}_n) \end{bmatrix}.$$

Since $P(\mathbf{f}|\mathcal{D})$ is not a Gaussian distribution anymore, for computing reason Laplace method has been used. By applying this method, the true posterior can be approximated by a Gaussian: $P(\mathbf{f}|\mathcal{D}) \approx \mathcal{N}(\mathbf{f}|\hat{\mathbf{f}}, \mathbf{A}^{-1})$, where $\hat{\mathbf{f}} = \arg\max_{\mathbf{f}} P(\mathbf{f}|\mathcal{D}) = \arg\max_{\mathbf{f}} P(\mathbf{f}) \prod_{k=1}^{M} P\big(\mathbf{r}_k \succ \mathbf{c}_k | f(\mathbf{r}_k), f(\mathbf{c}_k)\big)$ and $\mathbf{A}$ is the Hessian of the negative log-posterior evaluated at $\hat{\mathbf{f}}$ [9]. To maximize $P(\mathbf{f}|\mathcal{D})$ is equivalent to maximize the following expression:

$$\psi(\mathbf{f}) = \sum_{k=1}^{M} \ln \Phi(z_k) - \frac{1}{2}\mathbf{f}^T \boldsymbol{\Sigma}^{-1} \mathbf{f}, \tag{9}$$

where $z_k = \frac{f(\mathbf{r}_k) - f(\mathbf{c}_k)}{\sigma_{\text{noise}}}$. It is a convex problem (proof in [10]), and Newton's method is used to obtain the following iterative update:

$$\begin{aligned} \mathbf{f}^{\text{new}} &= (\boldsymbol{\Sigma}^{-1} + \mathbf{W})^{-1} \Big( \frac{\partial \ln P(\mathcal{D}|\mathbf{f})}{\partial \mathbf{f}} + \mathbf{W}\mathbf{f} \Big) \\ &= (\boldsymbol{\Sigma}^{-1} + \mathbf{W})^{-1} \Big( \sum_{k=1}^{M} \frac{\partial \big[\ln \Phi(z_k)\big]}{\partial \mathbf{f}} + \mathbf{W}\mathbf{f} \Big), \end{aligned} \tag{10}$$

with $\mathbf{W}_{ij} = \sum_{k=1}^{M} \frac{\partial^2 \big[-\ln \Phi(z_k)\big]}{\partial f(\mathbf{x}_i) \partial f(\mathbf{x}_j)}$.

The first order and second order derivatives can be written as (proof is provided in Appendix)

$$\frac{\partial \big[-\ln \Phi(z_k)\big]}{\partial f(\mathbf{x}_i)} = -\frac{s_k(\mathbf{x}_i)}{\sigma_{\text{noise}}} \frac{\mathcal{N}(z_k; 0, 1)}{\Phi(z_k)}, \tag{11}$$

$$\begin{aligned} \frac{\partial^2 \big[-\ln \Phi(z_k)\big]}{\partial f(\mathbf{x}_i) \partial f(\mathbf{x}_j)} &= \frac{s_k(\mathbf{x}_i) s_k(\mathbf{x}_j)}{\sigma^2_{\text{noise}}} \\ &\quad \cdot \Big( \frac{\mathcal{N}^2(z_k; 0, 1)}{\Phi^2(z_k)} + \frac{z_k \mathcal{N}(z_k; 0, 1)}{\Phi(z_k)} \Big), \end{aligned} \tag{12}$$

where $s_k(\mathbf{x}) = \begin{cases} 1, & \mathbf{x} = \mathbf{r}_k \\ -1, & \mathbf{x} = \mathbf{c}_k \\ 0, & \text{otherwise} \end{cases}$.

Once we have found the maximum posterior $\hat{\mathbf{f}}$ by using the above iteration we can show that:

$$P(\mathbf{f}|\mathcal{D}) \approx \mathcal{N}(\mathbf{f}|\hat{\mathbf{f}}, (\boldsymbol{\Sigma}^{-1} + \mathbf{W})^{-1}). \tag{13}$$

According to the previous results, the predictive distribution for an unseen sample $f_*(\mathbf{x}_*)$ can be derived as (proof is provided in Appendix)

$$\begin{aligned} P(f_*|\mathcal{D}) &= \int P(f_*|\mathbf{f}) P(\mathbf{f}|\mathcal{D}) \mathrm{d}\mathbf{f} \\ &= \mathcal{N}(\mathbf{k}^T \boldsymbol{\Sigma}^{-1} \hat{\mathbf{f}}, k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}^T (\boldsymbol{\Sigma} + \mathbf{W}^{-1})^{-1} \mathbf{k}), \end{aligned} \tag{14}$$

where $\begin{bmatrix} \mathbf{f} \\ f_* \end{bmatrix} \sim \mathcal{N}\Big( \begin{bmatrix} \mathbf{0} \\ 0 \end{bmatrix}, \begin{bmatrix} \boldsymbol{\Sigma} & \mathbf{k} \\ \mathbf{k}^T & k(\mathbf{x}_*, \mathbf{x}_*) \end{bmatrix} \Big)$, $\mathbf{k} = [k(\mathbf{x}_1, \mathbf{x}_*), k(\mathbf{x}_2, \mathbf{x}_*), \ldots, k(\mathbf{x}_M, \mathbf{x}_*)]^T$. This predictive distribution is what we want to get by doing such complex mathematical computing, which can be used by a criterion. During this procedure, the sample which is most interested by users will be found and will be explored during next iteration. The criteria will be discussed in the following section.

### *D. Acquisition functions*

In many cases, measurements for target utility functions are limited due to the high cost of measurements or inconvenience for users. With such limitations, every sample for measurement should be as efficient as possible and acquisition functions are used for providing information for next sampling. Based on the definition of GP, the acquisition function denoted by $a : \boldsymbol{\chi} \to \mathbb{R}^+$, determines which sample should be evaluated next via an optimization [8]

$$\mathbf{x}_{\text{next}} = \arg\max_{\boldsymbol{\chi}} a(\mathbf{x}).$$

The decision represents an automatic trade-off between exploration (where the utility function is very uncertain) and exploitation (trying values of x where the utility function is expected to be high) [5]. For interested readers, some traditional acquisition functions have been discussed in the paper [8].

In this work, the acquisition function is chosen to be expected improvement [15]. The improvement function is defined as

$$I(\mathbf{x}) = \max\{0, f(\mathbf{x}) - f(\mathbf{x}^+)\}, \tag{15}$$

where $\mathbf{x}^+$ is the best current value. The new sample is found by maximizing the expected improvement:

$$\mathbf{x} = \arg\max_{\mathbf{x}} EI(\max\{0, f(\mathbf{x}) - f(\mathbf{x}^+)\}|\mathcal{D}). \tag{16}$$

For the utility function value of every sample $f(\mathbf{x}) \sim \mathcal{N}\big(\mu(\mathbf{x}), \sigma^2(\mathbf{x})\big)$, the expected improvement can be calculated over the integral (detailed explanation please refer to [5]):

$$\begin{aligned} EI(\mathbf{x}) &= \int_0^\infty I \cdot \mathcal{N}(\mu(\mathbf{x}) - f(\mathbf{x}^+), \sigma^2(\mathbf{x}))\mathrm{d}I \\ &= \begin{cases} \big(\mu(\mathbf{x}) - f(\mathbf{x}^+)\big)\Phi(Z) + \sigma(\mathbf{x})\phi(Z), & \sigma(\mathbf{x}) > 0 \\ 0, & \sigma(\mathbf{x}) = 0 \end{cases} \end{aligned} \tag{17}$$

where $Z = \frac{\mu(\mathbf{x}) - f(\mathbf{x}^+)}{\sigma(\mathbf{x})}$ and $\phi(\cdot)$ is the PDF of the standard normal distribution. For the pairwise comparison situation, $\sigma(\mathbf{x})$ should be replaced by $\sigma(\mathbf{x}) + \sigma(\mathbf{x}^+) + \sigma_{\text{noise}}$ (neglect $\sigma_{\text{noise}}$ if $\sigma_{\text{noise}} << \sigma(\mathbf{x}) + \sigma(\mathbf{x}^+)$).

DIRECT, an algorithm for finding the global minimum of a multivariate function subject to simple bounds [16], is implemented for the acquisition function optimization in this work. Comparing with most competing methods, it is an efficient algorithm which converges in fewer function evaluations than others [16].

### *E. Sampling with Kushner's criterion [17]*

In previous subsection we have discussed about expected improvement criterion which focuses on sampling the point with the largest expected improvement over the current minimum estimate. Another criterion proposed by Kushner which samples the point with the highest probability of lying below the current minimum estimate [18]. During the project, we tried to make full use of the information given by Gaussian Process and proposed the next sample

$$\begin{aligned} \mathbf{x}_{\text{next}} &= \arg\max_{\mathbf{x}} p(f(\mathbf{x}) \leq f(\mathbf{x}^+)|\mathcal{D}) \\ &= \Phi\left(\frac{\mathbf{E}[f(\mathbf{x}^+)] - \mathbf{E}[f(\mathbf{x})]}{\sqrt{\text{var}[f(\mathbf{x}^+)] + \text{var}[f(\mathbf{x})] - 2\text{cov}[f(\mathbf{x}^+), f(\mathbf{x})]}}\right). \end{aligned} \tag{18}$$

After experiments, we found that this criterion works well in absolute rating systems but not in pairwise comparison system. Because for each iteration, the system generates a new sample and make a comparison with previous best sample. The information we can receive is really limited and the largest value of previous formula will always located around the best sample we have ever explored. Then we cannot receive good recommendation for next trial.

### *F. Summary*

In this section, we have introduced Bayesian Optimization for Parameter Exploration. During each iteration, users are asked to determine which sample is better over two given samples. Based on their feedback, the posterior distribution of GP will be updated. Then acquisition functions are used for providing users another sample for next iteration. For more details, readers can refer to Algorithm 1.

**Algorithm 1** Preference Learning with Gaussian Process

1: Initialize Gaussian Process as in (1) and set the iteration times;
2: **for** each iteration **do**
3: Present a present pair to the user (randomly generate 2 samples in the first iteration);
4: Receive feedback from the user as in (5);
5: Update posterior distribution as in (10);
6: Find the maximum value of acquisition function as in (17);
7: Generate next comparison pair;
8: **end for**

## III. Parameter Compression

When users are not satisfied with the results provided by recommendation procedure, they may need to tune parameters manually. But as we discussed before, there are tens of parameters needed to be modified which bring huge burden for users. For users they should not need to care about the high-dimensional data. Then we need to find a mapping which connects high-dimensional and 2D data. In our idea, dimensionality reduction (manifold learning) should be implemented which projects high dimensional data to a 2D space. Based on the high-to-low projection, we can easily find the high-dimensional samples if 2D data is provided with inverse operation.

Dimensionality reduction is a significant problem across a wide variety of information processing fields including pattern recognition, data compression, machine learning and database navigation [19]. A lot of algorithms for dimensionality reduction have been proposed. In this section, some classical algorithms as well as some new proposed ideas for dimensionality reduction (like PCA, LLE, ISOMAP) are briefly introduced and a novel approach for parameter projection will be introduced later.

For the whole hearing aid system, all works are based on Bayesian theory. For DSP block, there are 17 parameters needed to be tuned and the data samples of these parameters is given. In this case, we should project samples with high probability rather than simply project all 17-dimensional space on the screen of a pad. In this work, two main issues are faced:

- Given a $Q$-dimensional parameter (in this work, $Q = 17$) $\boldsymbol{\theta} = [\theta_1, \ldots, \theta_Q]$, with given data samples, find the best projection which can project $\boldsymbol{\theta}$ on a 2-dimensional pad for exploration;
- Preserve small perceptual changes for small distances on the pad.

### *A. Related work of dimensionality reduction algorithms*

There are a lot of papers which provide introduction of dimensionality reduction. For example, many traditional methods as well as their advantages and disadvantages are listed in [20]. Other than [20], there is some literature discussing probabilistic models for dimensionality reduction [21], [22], [23] and [24]. Lawrence in [21] proposed Maximum Entropy Unfolding (MEU) to project data and his work is similar

with maximum variance unfolding algorithm (MVU, which maximizes the total latent variance of a data set under local distance constraints). But this work maximizes the product of eigenvalues of Kernel matrix while MVU maximizes the sum of eigenvalues. Tipping and Bishop in [22] proposed PPCA which focuses on building a probabilistic model based on standards PCA. Target (low dimension) and observed (high dimension) data are treated as Gaussian distributions and a noise term is induced in this model. Collins et al. [23] can be treated as an extension of the work of Tipping and Bishop. In this paper, Michael Collins et al. propose a general way for dealing with exponential family and build a generalization model of PCA. In [25] and [26], two complex algorithms are proposed which are based on Rank Priors and neural network. In one interesting reference [24], the authors in this work proposed a new definition of distance based on posterior distribution based on Sammons mapping. During the literature review it became clear that most of the dimensionality reduction algorithms are used for machine learning, which pursuit high classification precision and is not very suitable for our work. In our work, the most vital issue is small perceptual changes preservation while the latest literature cannot solve this problem. An traditional algorithm, ISOMAP, works well and it will be discussed in the next subsection.

### *B. ISOMAP*

In order to preserve small perceptual changes for small distances, ISOMAP is a good choice for the first step as we discussed before. A short introduction of ISOMAP (demo is shown in Fig. 4) is provided in this section based on [19].

Isomap is a nonlinear generalization of classical MDS (introduction of MDS can also be found in [19]). In ISOMAP, Euclidean distances are replaced by geodesic distances. The geodesic distances represent the shortest paths along the curved surface of the manifold measured as if the surface were flat. ISOMAP then applies MDS to find a low-dimensional mapping that preserves these pairwise distances.

Isomap algorithm proceeds in three steps:

- Find the neighbours of each data point in high-dimensional data space by k-Nearest Neighbor algorithm;
- Compute the geodesic pairwise distances between all points by Dijkstra's algorithm [27];
- Embed the data via MDS so as to preserve these distances.

In this figure we find the drawback of methods based on PCA, another traditional and useful algorithm, is obvious: the topology of data will be verified after transform and Gaussian model is not suitable for every data set.

### *C. Parameter compression for hearing-aid tuning*

For our parameter compression, there are three procedures needed to be done. First, apply ISOMAP and then use "coarse quantization" proposed by us. Finally, implement regularization method. After ISOMAP projection, data from high-dimension is projected into 2D continuous plane. The rest problem is how we can interpolate the rest of data in 2D space. For continuous plane, it is not easy to deal with this interpolation work (as shown in Fig. 4d). So an efficient way for this is project these data to a discrete plane and then the rest of data points are easy to be interpolated by their 4 neighbours. Coarse quantization, a quantization algorithm, is implemented in order to quantify the samples from 2D continuous space to 2D discrete space. First, an automatic quantification is shown in Fig. 5. A straight line (slope equals 1) is used to sweep along the whole continuous plane from top to bottom. When the first sample is found, put it in the $(1,1)$ in the new discrete plane. After that, search for the next two samples, then put it in the $(2,1)$ and $(1,2)$ in the new discrete plane based on their geometry relation. Sweep all the plane as such until all the samples are put in the discrete plane. This procedure is called coarse quantization.

Taking the Swiss Roll dataset as the example, now we can discuss our algorithm which is shown in Fig. 6. In order to illustrate the idea, each sample is connected with their four nearest neighbours in Fig. 6a. It is easy to quantify the data by coarse quantization and the results are shown in Fig. 6b and Fig. 6c. Some of the geometrical relationship is lost after coarse quantization. In this case, we propose a regularization algorithm to recover such relations. Denote $\overrightarrow{\mathbf{r}_i}$ as the coordinate of the $i$-th sample in continuous plane while $\overrightarrow{\mathbf{k}_i}$ as coordinates of the $i$-th sample in the discrete plane, and denote $adj(i)$ as 4 nearest adjacent neighbours of sample $i$. Based on this, we define a cost function as

$$Cost(K) = \sum_i d^2(\overrightarrow{\mathbf{r}_i}, \overrightarrow{\mathbf{k}_i}) + \frac{\alpha}{2} \sum_i \sum_{j \in adj(i)} d^2(\overrightarrow{\mathbf{k}_i}, \overrightarrow{\mathbf{k}_j}), \quad (19)$$

where $d(\cdot,\cdot)$ is the distance between two samples and $K = \{\overrightarrow{\mathbf{k}_1}, \overrightarrow{\mathbf{k}_2}, \ldots, \overrightarrow{\mathbf{k}_n}\}$, $n$ is the number of samples. Then the results will be acquired by finding the optimization value

$$K_{opt} = \arg\min_K Cost(K). \quad (20)$$

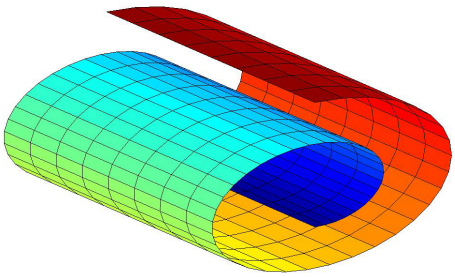
The computational complexity is $\mathcal{O}(2^n)$ in (20), which means it is impossible to calculate the cost function for each case. A greedy algorithm is taken in this work which is based on the coarse quantization. It is obvious that if we want to swap between sample $i$ and sample $j$, we do not need to calculate formula (19) for twice, we just need to calculate the difference of the cost function $\Delta Cost(i,j)$ between after swapping and before swapping with $i$ and $j$. Then we have

$$\begin{aligned}\Delta Cost(i,j) = {} & d^2(\overrightarrow{\mathbf{r}_i}, \overrightarrow{\mathbf{k}_i}) + d^2(\overrightarrow{\mathbf{r}_j}, \overrightarrow{\mathbf{k}_j}) \\ & - d^2(\overrightarrow{\mathbf{r}_i}, \overrightarrow{\mathbf{k}_j}) - d^2(\overrightarrow{\mathbf{r}_j}, \overrightarrow{\mathbf{k}_i}) \\ & + \alpha \Big[ \sum_{p \in adj(i)} d^2(\overrightarrow{\mathbf{k}_i}, \overrightarrow{\mathbf{k}_p}) + \sum_{q \in adj(j)} d^2(\overrightarrow{\mathbf{k}_j}, \overrightarrow{\mathbf{k}_q}) \\ & - \sum_{q \in adj(j)} d^2(\overrightarrow{\mathbf{k}_i}, \overrightarrow{\mathbf{k}_q}) - \sum_{p \in adj(i)} d^2(\overrightarrow{\mathbf{k}_j}, \overrightarrow{\mathbf{k}_p}) \Big]. \end{aligned} \quad (21)$$

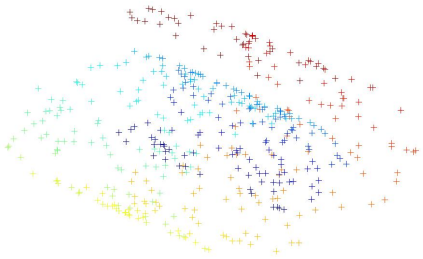
For each iteration, find

$$(i_{\text{best}}, j_{\text{best}}) = \arg\max_{(i,j)} \Delta Cost(i,j),$$

and swap them to continue next iteration until the local optimization solution is found. The computational complexity

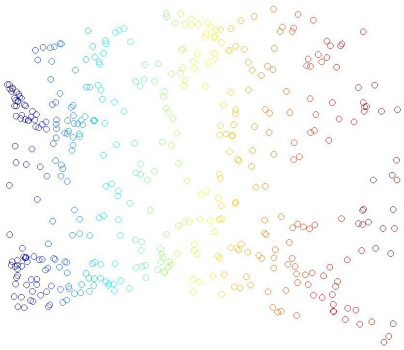


(a) Swiss Roll dataset



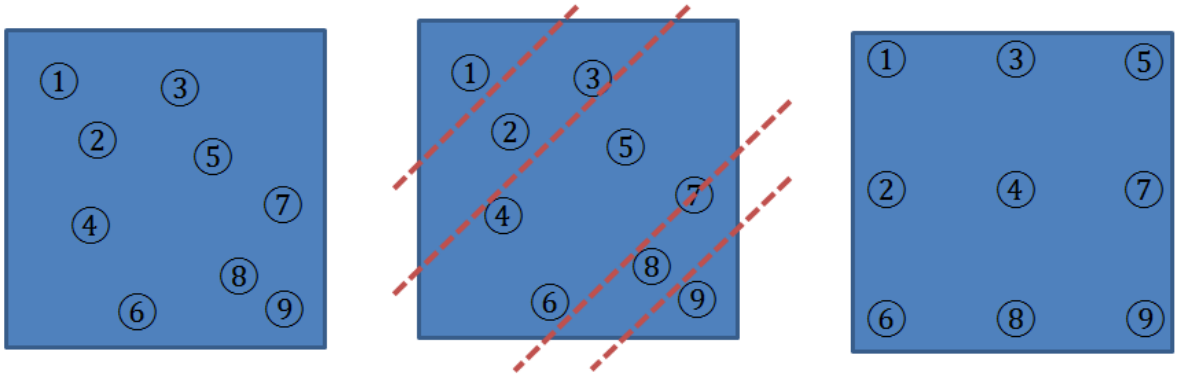
(b) Samples from dataset

(c) PCA results



(d) ISOMAP results

Fig. 4: Demo for 3D to 2D projection. For the classical testing dataset Swiss Roll (which is an popular reference 3D dataset for testing dimensionality reduction algorithms), ISOMAP can successfully project 3D dataset to 2D plane while PCA provides a bad result. In addition, in this demo ISOMAP preserves the geodesic distance.




(a) Original projection after ISOMAP (b) Sweep in continuous 2D plane (c) Results after coarse quantization

Fig. 5: Schematic diagram for coarse quantization procedure. A straight line is used to sweep along the whole continuous plane from top to bottom to quantify the 2D data samples.

is $\mathcal{O}(n^2)$ for each iteration and the results for Swiss Roll dataset can be found in Fig. 6d and Fig. 6e. Lots of the local information has been recovered and most of the samples locate near to neighbours. Also it should be emphasized that a suitable parameter $\alpha$ is needed to balance the distortion of quantization and local information.

## IV. Adaptive Resolution (Zooming)

According to Gaussian Process and Bayesian learning, we can evaluate the acquisition function in two-dimensional interval. We can believe that users are more probably able to find the parameters in the area with high acquisition function value. In this case, it is natural that users would like to have a better resolution in such high value area than other parts. The term "adaptive resolution" stands for the resolution on the pad which highly depends on the acquisition function. In this section, an algorithm for adaptive resolution will be discussed and our purpose is design a projection which can project space with high acquisition value to a larger space. To begin with, an one-dimensional example will be given in order to clarify the idea of this work.

### A. One-dimensional example for adaptive resolution

The one-dimensional example is shown in Fig. 7. Our target is to design a projection which has high resolution in the interval with high acquisition value. Here a simple projection in the interval of $[0, 1]$ is proposed:

$$a' = \frac{1}{Y_1}\int_0^a f(x)\mathrm{d}x, \tag{22}$$

where $Y_1$ is normalization factor and $Y_1 = \int_0^1 f(x)\mathrm{d}x$.

On the $x'$ axis of this figure, there are different resolution for different interval. For example, there is a high acquisition function value in the interval of $[0.2, 0.3]$, and a low acquisition function value in the interval of $[0.9, 1]$. As a result, the resolution in the interval of $[0.2, 0.3]$ is significantly higher than $[0.9, 1]$.

Now, this intuitive idea will be applied to the two-dimensional case.

### B. Two-dimensional algorithm for adaptive resolution

The idea for two-dimensional space $[0, 1]^2$ is borrowed from the one-dimensional example. For a sample $(a, b)$ $(a, b \in [0, 1])$, we propose an adaptive resolution projection $\mathbb{P}$:

$$\begin{cases} a' = \dfrac{1}{Y_2}\displaystyle\int_0^1\int_0^a f^p(x, y)\mathrm{d}x\mathrm{d}y, \\ b' = \dfrac{1}{Y_2}\displaystyle\int_0^b\int_0^1 f^p(x, y)\mathrm{d}x\mathrm{d}y, \end{cases} \tag{23}$$

where $Y_2$ is normalization factor and $Y_2 = \int_0^1\int_0^1 f^p(x, y)\mathrm{d}x\mathrm{d}y$ and $f(x, y) > 0, \forall(x, y) \in [0, 1]^2$. $p$ is the exponential factor and $p > 0$. Larger $p$ corresponds to higher degree of zooming.

The integrations can be replaced by sums for implementation. With the projection $\mathbb{P}$, there are three important properties (brief proof is provided in Appendix):

- With equation (23), $(a', b') \in [0, 1]^2, \forall(a, b) \in [0, 1]^2$;
- Projection $\mathbb{P}$ is a bijection;
- For a line segment $L_1 : (a_1, b_1), (a_2, b_2)$, the new line segment $L_2 : (a'_1, b'_1), (a'_2, b'_2)$ will be received after projection $\mathbb{P}$. In the route from $L_1$ to $L_2$, $f(x, y) > 1$. Then we have $\|L_1\| < \|L_2\|$.

With these three properties, we can say that this projection is suitable for our needs. First a sample from $[0, 1]^2$ space will received its projection in the same space, and different samples have different projections. Also, the most interesting property for our work is samples from high acquisition function value will acquire higher contrast space and users can easily tune parameters in such projection space.

In Fig. 8, we have clearly demonstrated our work. In the figures, blue regions stand for areas with low acquisition value, while dark red stands for highest acquisition value in the whole space. Comparing with this two images, clearly that blue zones have been compressed while red zones have been enlarged, which successfully fulfill our requirements.

## V. Performance Evaluation

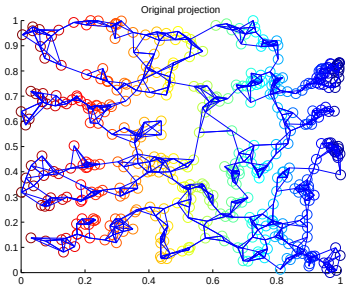
In this section, we will evaluate the performance of our system. Instead of evaluate the whole system together, we have verified our work in a separate way. First, GP will be tested and then our dimensionality algorithm will be evaluated.

### A. Comparison between exploration with pairwise comparison and randomly sampling

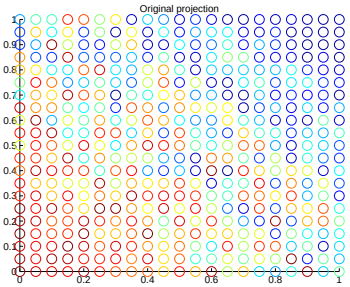
After the introduction of GP, we would like to show the superiority of its iteration times. Here a comparison will be proposed between GP and randomly sampling. Three test functions are used during this experiment which are Brainin (three global minimum positions), Goldstein (one global minimum position) and Six-hump Camel Back (two global minimum positions). Our target is to find one of their global minimal positions by using GP and randomly sampling. If the sample they choose is near the target minimal position (the distance is less than 0.05 in this experiment), then we treat the minimum has been found. In order to "help" randomly sampling, new picked samples will have a distance larger than 0.05 with all previous samples. For both GP and randomly sampling, our experiments will run 50 times for each target function and the results have been shown in Table I.

In the table, success means the method find target best value within 15 iterations this time and average means the average number of iterations in all successful sampling. With the comparison, it is obvious that GP plays far better than randomly sampling, which shows it is quite necessary to provide recommendation pairs for user to find their best configurations. Otherwise users can hardly find their preference settings within 15 trials.

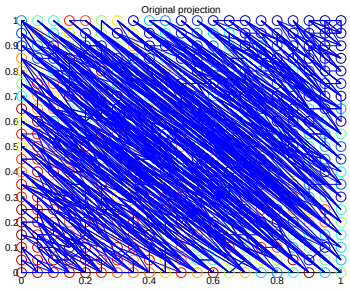
Also, another experiment has shown in Fig. 9. For each of these three test functions, evaluations have been taken for 20 times. Based on the results we get, the error of mean and standard variance have been shown in the figure. Via this experiment, we can say that our method is fulfill our requirements because it can converge to zero-error quickly.



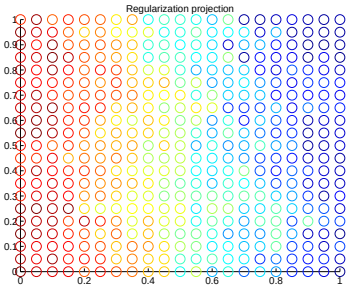

(a) Original projection after ISOMAP with connected neighbours



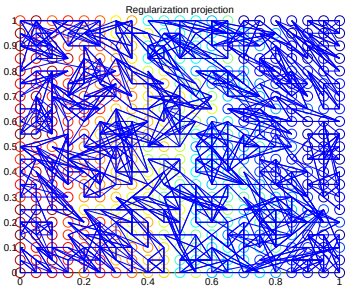

(b) Results after coarse quantization




(c) Results after coarse quantization with connected neighbours




(d) Results after regularization




(e) Results after regularization with connected neighbours

Fig. 6: Results with regularization for Swiss Roll dataset.

| | **Brainin** | | **Goldstein** | | **Six-hump Camel Back** | |
|---|---|---|---|---|---|---|
| | *Random* | *GP* | *Random* | *GP* | *Random* | *GP* |
| **Success Rate** | 28% | 78% | 14% | 82% | 18% | 98% |
| **Average (success)** | 8.2 | 8.5 | 9.4 | 6.0 | 7.0 | 7.6 |

TABLE I: Comparison between Gaussian process and randomly sampling.

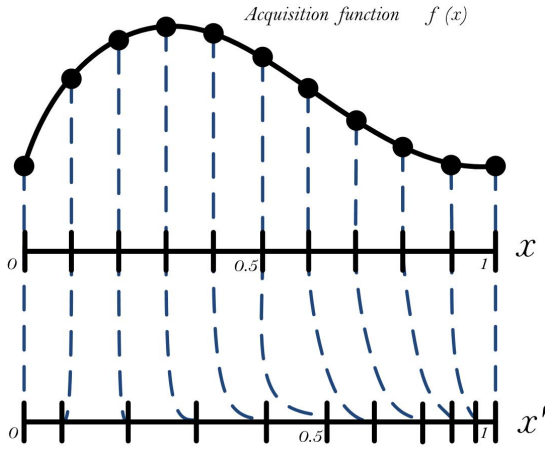
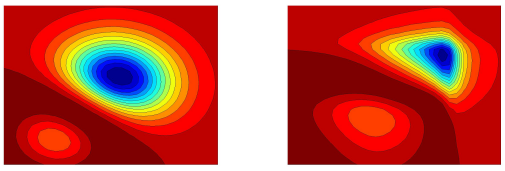



Fig. 7: One-dimensional example for adaptive resolution. After projection, it is easy to find the length of the interval with high acquisition value is larger than lower acquisition value interval.

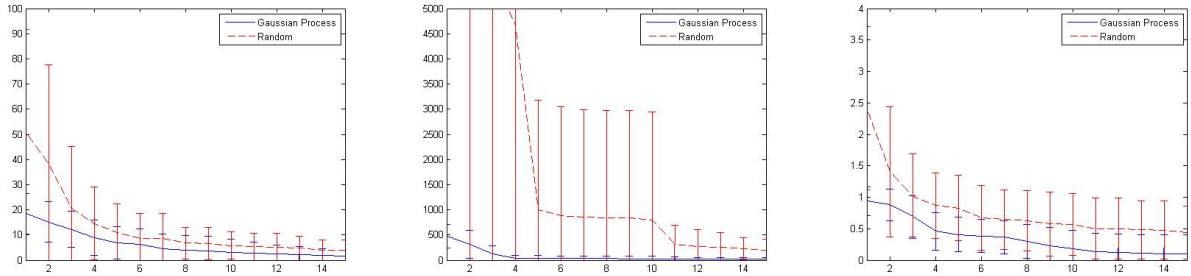


(a) Figure without zooming (b) Figure with zooming

Fig. 8: Comparison between figures with and without zooming. By applying zooming function, users have higher possibility to find samples in areas with high acquisition value.

### *B. Comparison of dimensionality reduction*

Dimensionality reduction algorithms are always used in the domain of machine learning, and there are many ways to evaluate their performance via accuracy, etc. But they are not suitable for our work. Because of this reason, we proposed an novel method for evaluation. In our work, we are most curious about the property of gradual change as we have mentioned in the problem statement. By using the same dataset for different algorithms, moving with same small distance (0.001 on both horizontal and vertical direction) on the 2D pad, we can evaluate the distance changes on original high dimensional datasets. In this evaluation, we use two dataset: typical audiogram vectors proposed in [28] and Swiss roll dataset. Other than ISOMAP and PCA, we also include LLE [29]and MDS [30], two other traditional dimensionality reduction algorithms, into comparison. For each algorithms, two cases: with regularization and without regularization are both considered in our experiments. Also, in order to make a clearly comparison, all the results has been divided by the




(a) Branin Function (b) Goldstein Function (c) Six-hump Camel Back function

Fig. 9: The evolution of error for the estimate of the optimum on the test functions. The plot shows the error against the number of iterations. The solid blue line is our method while the dashed red line is a baseline method in which each point is selected randomly.

| | *ISOMAP* | *PCA* | *LLE* | *MDS* |
|---|---|---|---|---|
| **with REG** | 1 | 1.015 | 1.008 | 1.036 |
| **without REG** | 1.163 | 1.087 | 1.036 | 1.099 |

TABLE II: Comparison between 4 different dimensionality reduction algorithms on audiogram dataset. The result by applying ISOMAP with REG has the lowest distance change in the high dimensional space.

| | *ISOMAP* | *PCA* | *LLE* | *MDS* |
|---|---|---|---|---|
| **with REG** | 1 | 1.119 | 1.102 | 1.107 |
| **without REG** | 1.123 | 1.158 | 1.156 | 1.184 |

TABLE III: Comparison between 4 different dimensionality reduction algorithms on Swiss Roll dataset. The result by applying ISOMAP with REG has the lowest distance change in the high dimensional space.

results received from ISOMAP with REG. In every test case, 1000 samples are used for distance calculation. All the results have been shown in Table II and Table III.

From the results we can find our ISOMAP with REG plays well, especially in Swiss Roll dataset. The main reason why ISOMAP is not significantly better than others is the audiogram dataset is too small (only 60 samples) and it is really hard to learn the manifold by such small dataset. But at Swiss Roll, we can clearly find the superiority of ISOMAP. Also, it should be mentioned that regularization improves the results for every manifold learning algorithm.

### C. Evaluation of zooming

As we have seen in the previous section, Zooming module is hard to evaluate quantitatively. But Fig. 8 in previous section can be referred as a result of this work.

## VI. Conclusion & Future Works

### A. Conclusion & Discussion

In this work, an experimental evaluation environment for interactive parameter exploration has been developed in MATLAB. There are three sections in this work: Bayesian optimization for parameter exploration, parameter compression and adaptive resolution. We have imported Gaussian Process with pairwise comparison into hearing-aid system, and researched on dimensionality reduction algorithms for our special requirements. For these two sections, we have evaluated them by experiments. Besides, a zooming function proposed by us has been added in order to satisfy users.

### B. Future works

- Implement algorithms on embedded platform.
- Optimization based on entropy minimization.

## Acknowledgment

After finishing all the paper work for this 32-week graduation project, I have to admit that this is a really challenging work which contains using Gaussian Process for best configurations recommendation, parameter compression for projection, zooming for adaptive resolution. Nearly every topic is a totally new domain and there is hardly to find any reference provided by others. In such difficult situation, my supervisor, Professor Bert de Vries, provides me lots of help in many aspects. He is not an armchair theorist, but on the contrary, he always brings me inspirations and helps me get out of various dilemmas. Also I should thank to the members in my graduation committee, they care about my work and provides me useful suggestions in the mid-term presentation. This topic contains lots of mathematical work, which is quite difficult for an engineering student. But luckily, my friends, Nabil Bouhouch and Kaidi Yang, who have strong mathematical background, help me overcome many troubles.

Thank you all!

## References

[1] C. Mathers, A. Smith, and M. Concha, "Global burden of hearing loss in the year 2000," *Global burden of Disease*, vol. 18, 2000.

[2] L. Knudsen, M. Öberg, C. Nielsen, G. Naylor, and S. Kramer, "Factors influencing help seeking, hearing aid uptake, hearing aid use and satisfaction with hearing aids: A review of the literature," *Trends in amplification*, vol. 14, no. 3, pp. 127–154, 2010.

[3] S. Kochkin, "Marketrak viii: 25-year trends in the hearing health market," *Hearing Review*, vol. 16, no. 11, pp. 12–31, 2009.

[4] ——, "Marketrak viii: Consumer satisfaction with hearing aids is slowly increasing," *The Hearing Journal*, vol. 63, no. 1, pp. 19–20, 2010.

[5] E. Brochu, V. Cora, and N. De Freitas, "A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning," *arXiv preprint arXiv:1012.2599*, 2010.

[6] R. M. Neal, "Monte carlo implementation of gaussian process models for bayesian regression and classification," *arXiv preprint physics/9701026*, 1997.

[7] C. M. Bishop *et al.*, *Pattern recognition and machine learning*. springer New York, 2006, vol. 4, no. 4.

[8] J. Snoek, H. Larochelle, and R. Adams, "Practical bayesian optimization of machine learning algorithms," *arXiv preprint arXiv:1206.2944*, 2012.

[9] E. Bonilla, S. Guo, and S. Sanner, "Gaussian process preference elicitation," 2010.

[10] W. Chu and Z. Ghahramani, "Preference learning with gaussian processes," in *Proceedings of the 22nd international conference on Machine learning*. ACM, 2005, pp. 137–144.

[11] E. Brochu, N. De Freitas, and A. Ghosh, "Active preference learning with discrete choice data," *Advances in neural information processing systems*, vol. 20, pp. 409–416, 2007.

[12] J. B. Nielsen, B. S. Jensen, and J. Larsen, "Pseudo inputs for pairwise learning with gaussian processes," in *Machine Learning for Signal Processing (MLSP), 2012 IEEE International Workshop on*. IEEE, 2012, pp. 1–6.

[13] J. B. Nielsen, "Preference based personalization of hearing aids," Master's thesis, IMM-M.Sc.-2010-61, 2010, Technical University of Denmark (DTU), Kgs. Lyngby, Denmark, 2010.

[14] C. E. Rasmussen and C. K. Williams, *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006, vol. 1.

[15] D. R. Jones, M. Schonlau, and W. J. Welch, "Efficient global optimization of expensive black-box functions," *Journal of Global optimization*, vol. 13, no. 4, pp. 455–492, 1998.

[16] D. R. Jones, C. D. Perttunen, and B. E. Stuckman, "Lipschitzian optimization without the lipschitz constant," *Journal of Optimization Theory and Applications*, vol. 79, no. 1, pp. 157–181, 1993.

[17] H. J. Kushner, "A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise," *Journal of Basic Engineering*, vol. 86, p. 97, 1964.

[18] I. M. Park, M. Nassar, and M. Park, "Active bayesian optimization: Minimizing minimizer entropy," *arXiv preprint arXiv:1202.2143*, 2012.

[19] A. Ghodsi, "Dimensionality reduction a short tutorial," *Department of Statistics and Actuarial Science, Univ. of Waterloo, Ontario, Canada*, 2006.

[20] L. Van der Maaten, E. Postma, and H. Van den Herik, "Dimensionality reduction: A comparative review," *Journal of Machine Learning Research*, vol. 10, pp. 1–41, 2009.

[21] N. D. Lawrence, "Probabilistic spectral dimensionality reduction," 2010.

[22] M. E. Tipping and C. M. Bishop, "Probabilistic principal component analysis," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 61, no. 3, pp. 611–622, 1999.

[23] M. Collins, S. Dasgupta, and R. E. Schapire, "A generalization of principal component analysis to the exponential family," in *NIPS 2001*, 2001.

[24] P. T. Kontkanen, J. M. A. Lahtinen, P. J. Myllymäki, T. V. Silander, H. R. Tirri, K. A. Valtonen *et al.*, "Visualization method and visualization system," Mar. 29 2005, uS Patent 6,873,325.

[25] A. Geiger, R. Urtasun, and T. Darrell, "Rank priors for continuous non-linear dimensionality reduction," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 880–887.

[26] Z. Meng and Y.-H. Pao, "Visualization and self-organization of multidimensional data through equalized orthogonal mapping," *Neural Networks, IEEE Transactions on*, vol. 11, no. 4, pp. 1031–1038, 2000.

[27] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische mathematik*, vol. 1, no. 1, pp. 269–271, 1959.

[28] N. Bisgaard, M. S. Vlaming, and M. Dahlquist, "Standard audiograms for the iec 60118-15 measurement procedure," *Trends in Amplification*, vol. 14, no. 2, pp. 113–120, 2010.

[29] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[30] T. F. Cox and M. A. Cox, *Multidimensional scaling*. CRC Press, 2001, vol. 88.


## APPENDIX A
## FORMULA DEDUCTION

### A. Proof of equation(6)

*Proof.*

$$\begin{aligned}
&P\big(\mathbf{r}_k \succ \mathbf{c}_k | f(\mathbf{r}_k), f(\mathbf{c}_k)\big) \\
&\quad = P\big(f(\mathbf{r}_k) - f(\mathbf{c}_k) > \varepsilon\big) \\
&\quad = P\Big(\frac{f(\mathbf{r}_k) - f(\mathbf{c}_k)}{\sigma_{\text{noise}}} > \varepsilon'\Big) \ \big(\varepsilon' \sim \mathcal{N}(0,1)\big) \\
&\quad = \int_{-\infty}^{+\infty} \mathbb{I}\Big[\varepsilon' < \frac{f(\mathbf{r}_k) - f(\mathbf{c}_k)}{\sigma_{\text{noise}}}\Big] \cdot \mathcal{N}(\varepsilon'; 0, 1)\, \mathrm{d}\varepsilon' \qquad \text{(A.1)} \\
&\quad = \int_{-\infty}^{\frac{f(\mathbf{r}_k) - f(\mathbf{c}_k)}{\sigma_{\text{noise}}}} \mathcal{N}(\varepsilon'; 0, 1)\, \mathrm{d}\varepsilon' \\
&\quad = \Phi\Big(\frac{f(\mathbf{r}_k) - f(\mathbf{c}_k)}{\sigma_{\text{noise}}}\Big),
\end{aligned}$$

where $\mathbb{I}[\cdot]$ is an indicator function that is 1 if the condition $[\cdot]$ is true and 0 otherwise, and $\Phi(\cdot)$ is the CDF of the standard normal distribution. □

### B. Proof of equation(11) & (12)

*Proof.*

$$\begin{aligned}
\frac{\partial\big[-\ln\Phi(z_k)\big]}{\partial f(\mathbf{x}_i)} &= -\frac{1}{\Phi(z_k)} \cdot \frac{\partial \Phi(z_k)}{\partial f(\mathbf{x}_i)} \\
&= -\frac{\mathcal{N}(z_k; 0, 1)}{\Phi(z_k)} \cdot \frac{\partial z_k}{\partial f(\mathbf{x}_i)} \qquad \text{(A.2)} \\
&= -\frac{s_k(\mathbf{x}_i)}{\sigma_{\text{noise}}} \frac{\mathcal{N}(z_k; 0, 1)}{\Phi(z_k)}.
\end{aligned}$$

$$\begin{aligned}
\frac{\partial^2\big[-\ln\Phi(z_k)\big]}{\partial f(\mathbf{x}_i)\partial f(\mathbf{x}_j)} &= \frac{\partial}{\partial f(\mathbf{x}_j)}\Big\{\frac{\partial\big[-\ln\Phi(z_k)\big]}{\partial f(\mathbf{x}_i)}\Big\} \\
&= -\frac{s_k(\mathbf{x}_i)}{\sigma_{\text{noise}}} \cdot \frac{\partial}{\partial f(\mathbf{x}_j)}\Big[\frac{\mathcal{N}(z_k; 0, 1)}{\Phi(z_k)}\Big] \\
&= -\frac{s_k(\mathbf{x}_i)}{\sigma_{\text{noise}}} \cdot \Bigg[\frac{\Phi(z_k) \cdot \frac{\partial \mathcal{N}(z_k; 0, 1)}{\partial f(\mathbf{x}_j)}}{\Phi^2(z_k)} \\
&\qquad - \frac{\mathcal{N}(z_k; 0, 1) \cdot \frac{\partial \Phi(z_k)}{\partial f(\mathbf{x}_j)}}{\Phi^2(z_k)}\Bigg] \\
&= \frac{s_k(\mathbf{x}_i) s_k(\mathbf{x}_j)}{\sigma_{\text{noise}}^2} \\
&\qquad \cdot \Big(\frac{\mathcal{N}^2(z_k; 0, 1)}{\Phi^2(z_k)} + \frac{z_k \mathcal{N}(z_k; 0, 1)}{\Phi(z_k)}\Big) \qquad \text{(A.3)}
\end{aligned}$$

□

### C. Proof of equation(14)

*Proof.* Since we have

$$\begin{bmatrix} \mathbf{f} \\ f_* \end{bmatrix} \sim \mathcal{N}\Bigg(\begin{bmatrix} \mathbf{0} \\ 0 \end{bmatrix}, \begin{bmatrix} \boldsymbol{\Sigma} & \mathbf{k} \\ \mathbf{k}^T & k(\mathbf{x}_*, \mathbf{x}_*) \end{bmatrix}\Bigg),$$

the conditional Gaussian distribution

$$P(f_* | \mathbf{f}) = \mathcal{N}\big(f_* | \mathbf{k}^T \boldsymbol{\Sigma}^{-1} \mathbf{f}, k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}^T \boldsymbol{\Sigma}^{-1} \mathbf{k}\big) \qquad \text{(A.4)}$$

In order to simplify the representation of formula, denote $P(f_*|\mathbf{f}) \sim \mathcal{N}(\mu_*, \Sigma_*)$ and $P(\mathbf{f}|\mathcal{D}) \sim \mathcal{N}(\boldsymbol{\mu}_f, \boldsymbol{\Sigma}_f)$. Then we have

$$P(f_*|\mathbf{f})P(\mathbf{f}|\mathcal{D}) \propto \exp\Big\{ -\frac{1}{2}\big[(\mathbf{f}-\boldsymbol{\mu}_f)^T\boldsymbol{\Sigma}_f^{-1}(\mathbf{f}-\boldsymbol{\mu}_f)\big] + \big[(f_*-\mu_*)^T\Sigma_*^{-1}(f_*-\mu_*)\big]\Big\}. \tag{A.5}$$

Let $\mathbf{x} = [\mathbf{x}_1, x_2]^T = [\mathbf{f}, f_* - \mu_*]^T$, then

$$\big[(\mathbf{f}-\boldsymbol{\mu}_f)^T\boldsymbol{\Sigma}_f^{-1}(\mathbf{f}-\boldsymbol{\mu}_f)\big] + \big[(f_*-\mu)^T\Sigma_*^{-1}(f_*-\mu)\big] \propto (\mathbf{x}-\boldsymbol{\mu}_x)^T \begin{bmatrix} \boldsymbol{\Sigma}_f^{-1} & 0 \\ 0 & \Sigma_*^{-1} \end{bmatrix} (\mathbf{x}-\boldsymbol{\mu}_x). \tag{A.6}$$

So we have

$$\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x) = \mathcal{N}\left( \begin{bmatrix} \boldsymbol{\mu}_f \\ 0 \end{bmatrix}, \begin{bmatrix} \boldsymbol{\Sigma}_f & 0 \\ 0 & \Sigma_* \end{bmatrix} \right).$$

Now we need to design a projection $\mathbf{P}$ which satisfy

$$\mathbf{P}\cdot\mathbf{x} = \begin{bmatrix} \mathbf{f} \\ f_* \end{bmatrix}, \tag{A.7}$$

and the solution is

$$\mathbf{P} = \begin{bmatrix} \mathbf{I} & 0 \\ \mu_*\mathbf{f}^{-1} & 1 \end{bmatrix}. \tag{A.8}$$

According to this,

$$\mathbf{P}\cdot\boldsymbol{\mu}_x = \begin{bmatrix} \boldsymbol{\mu}_f \\ \mu_*\mathbf{f}^{-1}\boldsymbol{\mu}_f, \end{bmatrix}, \tag{A.9}$$

and

$$\mathbf{P}\boldsymbol{\Sigma}_x\mathbf{P}^T = \begin{bmatrix} \cdots & \cdots \\ \cdots & \mu_*\mathbf{f}^{-1}\boldsymbol{\Sigma}_x(\mu_*\mathbf{f}^{-1})^T + \Sigma_* \end{bmatrix}. \tag{A.10}$$

Till now we have derived that

$$\begin{aligned} P(f_*|\mathcal{D}) &\sim \mathcal{N}(\mu_*\mathbf{f}^{-1}\boldsymbol{\mu}_f, \mu_*\mathbf{f}^{-1}\boldsymbol{\Sigma}_f(\mu_*\mathbf{f}^{-1})^T + \Sigma_*) \\ &= \mathcal{N}(\mathbf{k}^T\boldsymbol{\Sigma}^{-1}\hat{\mathbf{f}}, \mathbf{k}^T\boldsymbol{\Sigma}^{-1}(\mathbf{W}+\boldsymbol{\Sigma}^{-1})^{-1}\boldsymbol{\Sigma}^{-1}\mathbf{k} \\ &\qquad + k(\mathbf{x}_*,\mathbf{x}_*) - \mathbf{k}^T\boldsymbol{\Sigma}^{-1}\mathbf{k}) \\ &= \mathcal{N}\Big(\mathbf{k}^T\boldsymbol{\Sigma}^{-1}\hat{\mathbf{f}}, k(\mathbf{x}_*,\mathbf{x}_*) \\ &\qquad - \mathbf{k}^T\big[\boldsymbol{\Sigma}^{-1} - \boldsymbol{\Sigma}^{-1}(\mathbf{W}+\boldsymbol{\Sigma}^{-1})^{-1}\boldsymbol{\Sigma}^{-1}\big]\mathbf{k}\Big). \end{aligned} \tag{A.11}$$

According to Matrix Inversion Lemma, it is easy to prove that $\boldsymbol{\Sigma}^{-1} - \boldsymbol{\Sigma}^{-1}(\mathbf{W}+\boldsymbol{\Sigma}^{-1})^{-1}\boldsymbol{\Sigma}^{-1} = (\boldsymbol{\Sigma}+\mathbf{W}^{-1})^{-1}$. □

### *D. Brief proof of properties in Section "Adaptive Resolution (Zooming)"*

*Proof.*

1. Since $0 \le \int_0^1\int_0^a f^p(x,y)\mathrm{d}x\mathrm{d}y \le \int_0^1\int_0^1 f^p(x,y)\mathrm{d}x\mathrm{d}y$, it is obvious that $0 \le a' \le 1$. The conclusion for $b'$ can be derived similar with $a'$.

2. It is obvious that projection $\mathbf{P}$ is a surjection, and the rest we need to prove is that $\mathbf{P}$ is a injection. Since $F(x) = \int_0^1\int_0^x f^p(x,y)\mathrm{d}x\mathrm{d}y$ is a monotone increasing function, then for $F(x) = F(x')$, there must be $x = x'$. The conclusion for $G(y) = \int_0^y\int_0^1 f^p(x,y)\mathrm{d}x\mathrm{d}y$ can be derived similar with $F(x)$.

3. Let $a_2' > a_1'$ and $b_2' > b_1'$ w.l.o.g., we have $a_2' - a_1' = \int_0^1\int_{a_1}^{a_2} f^p(x,y)\mathrm{d}x\mathrm{d}y > \int_0^1\int_{a_1}^{a_2} 1\ \mathrm{d}x\mathrm{d}y = a_2 - a_1$. Also we have $b_2' - b_1' > b_2 - b_1$, then $\|L_2\| = \sqrt{(a_2'-a_1')^2 + (b_2'-b_1')^2} > \sqrt{(a_2-a_1)^2+(b_2-b_1)^2} = \|L_1\|$. □