Editor-in-Chief
Dr. Saif alZahir

SIGNAL PROCESSING (SPIJ)

AN INTERNATIONAL JOURNAL

ISSN : 1985-2339

Volume 6 • Issue 5 • December 2012
Publication Frequency: 6 Issues / Year

CSC PUBLISHERS
http://www.cscjournals.org

SIGNAL PROCESSING: AN INTERNATIONAL JOURNAL (SPIJ)

VOLUME 6, ISSUE 5, 2012

EDITED BY
DR. NABEEL TAHIR

ISSN (Online): 1985-2339

International Journal of Computer Science and Security is published both in traditional paper form and in Internet. This journal is published at the website http://www.cscjournals.org, maintained by Computer Science Journals (CSC Journals), Malaysia.

SPIJ Journal is a part of CSC Publishers
Computer Science Journals
http://www.cscjournals.org

SIGNAL PROCESSING: AN INTERNATIONAL JOURNAL (SPIJ)

Book: Volume 6, Issue 5, December 2012
Publishing Date: 31-12-2012
ISSN (Online): 1985-2339

SPIJ Journal is a part of CSC Publishers
http://www.cscjournals.org

Published in Malaysia

Typesetting: Camera-ready by author, data conversation by CSC Publishing Services – CSC Journals, Malaysia

CSC Publishers, 2012

EDITORIAL PREFACE

This is *Fifth* Issue of Volume *Six* of the Signal Processing: An International Journal (SPIJ). SPIJ is an International refereed journal for publication of current research in signal processing technologies. SPIJ publishes research papers dealing primarily with the technological aspects of signal processing (analogue and digital) in new and emerging technologies. Publications of SPIJ are beneficial for researchers, academics, scholars, advanced students, practitioners, and those seeking an update on current experience, state of the art research theories and future prospects in relation to computer science in general but specific to computer security studies. Some important topics covers by SPIJ are Signal Filtering, Signal Processing Systems, Signal Processing Technology and Signal Theory etc.

The initial efforts helped to shape the editorial policy and to sharpen the focus of the journal. Started with Volume 6, 2012, SPIJ appears with more focused issues related to signal processing studies. Besides normal publications, SPIJ intend to organized special issues on more focused topics. Each special issue will have a designated editor (editors) – either member of the editorial board or another recognized specialist in the respective field.

This journal publishes new dissertations and state of the art research to target its readership that not only includes researchers, industrialists and scientist but also advanced students and practitioners. The aim of SPIJ is to publish research which is not only technically proficient, but contains innovation or information for our international readers. In order to position SPIJ as one of the top International journal in signal processing, a group of highly valuable and senior International scholars are serving its Editorial Board who ensures that each issue must publish qualitative research articles from International research communities relevant to signal processing fields.

SPIJ editors understand that how much it is important for authors and researchers to have their work published with a minimum delay after submission of their papers. They also strongly believe that the direct communication between the editors and authors are important for the welfare, quality and wellbeing of the Journal and its readers. Therefore, all activities from paper submission to paper publication are controlled through electronic systems that include electronic submission, editorial panel and review system that ensures rapid decision with least delays in the publication processes.

To build its international reputation, we are disseminating the publication information through Google Books, Google Scholar, Directory of Open Access Journals (DOAJ), Open J Gate, ScientificCommons, Docstoc and many more. Our International Editors are working on establishing ISI listing and a good impact factor for SPIJ. We would like to remind you that the success of our journal depends directly on the number of quality articles submitted for review. Accordingly, we would like to request your participation by submitting quality manuscripts for review and encouraging your colleagues to submit quality manuscripts for review. One of the great benefits we can provide to our prospective authors is the mentoring nature of our review process. SPIJ provides authors with high quality, helpful reviews that are shaped to assist authors in improving their manuscripts.

Editorial Board Members
Signal Processing: An International Journal (SPIJ)

TABLE OF CONTENTS

Volume 6, Issue 5, December 2012

Pages

Survey On Speech Synthesis

A. Indumathi *induravi.cbe@gmail.com*
Asst.Professor,Department of Computer Applications(MCA)
Dr.SNS Rajalakshmi College Of Arts & Science
Coimbatore,Tamil Nadu,India

Dr. E. Chandra *crcspeech@gmail.com*
Director,Department of Computer Applications(MCA)
Dr.SNS Rajalakshmi College OF Arts & Science
Coimbatore, Tamil Nadu, India

Abstract

The primary goal of this paper is to provide an overview of existing Text-To-Speech (TTS) Techniques by highlighting its usage and advantage. First Generation Techniques includes Formant Synthesis and Articulatory Synthesis. Formant Synthesis works by using individually controllable formant filters, which can be set to produce accurate estimations of the vocal-track transfer function. Articulatory Synthesis produces speech by direct modeling of Human articulator behavior. Second Generation Techniques incorporates Concatenative synthesis and Sinusoidal synthesis. Concatenative synthesis generates speech output by concatenating the segments of recorded speech. Generally, Concatenative synthesis generates the natural sounding synthesized speech. Sinusoidal Synthesis use a harmonic model and decompose each frame into a set of harmonics of an estimated fundamental frequency. The model parameters are the amplitudes and periods of the harmonics. With these, the value of the fundamental can be changed while keeping the same basic spectral..In adding, Third Generation includes Hidden Markov Model (HMM) and Unit Selection Synthesis.HMM trains the parameter module and produce high quality Speech. Finally, Unit Selection operates by selecting the best sequence of units from a large speech database which matches the specification.

Keywords: TTS, HMM, Synthesis

1. INTRODUCTION

With advancement in global communication, Speech processing has been a main stream in the area of research. The key goal of speech research is to build a system that mimic human. It can be achieved by speech recognition and speech synthesis. The first one converts the speech into text information, whereas, the second converts the text message information into speech which is also called Text-To-Speech (TTS).

Speech synthesis is rapidly developing technology which consists of two major phases [1].

(i) Text Analysis where the input as text is transcribed into a phonetic or linguistic representation using pronunciation rules and
(ii) Generation of speech waveforms or speech synthesis, where the acoustic speech output is produced from phonetic and prosodic information

Speech Synthesis can be performed by using any one of the approach in Speech Generation Techniques [3]. They are

- First Generation Techniques
- Second Generation Techniques
- Third Generation Techniques

First Generation Techniques require a quite detailed, low-level description of what is to be spoken. Second Generation Techniques uses Data driven approach to increase the quality of speech by reducing modeling pitch and timing. On the other hand, Third Generation techniques use Statistical, Machine-learning techniques to infer the specification - to - parameter mapping from data. Main reason for the arise of third generation is due to less memory occupation for storing factors of the model than to memorize the data. In addition to that, it allows to modify the model in various ways, like converting the original voice into a different voice[2].

2. FORMANT SYNTHESIS

Formant Synthesis was the first synthesis technique to be developed and was the dominant technique until the early 1980's.Formant Synthesis is often called synthesis by rule. The basic assumption of Formant synthesis is to model vocal tract transfer function by simulating formant frequencies and formant amplitudes. The vocal track has certain major resonant frequencies[4] .The frequencies change as the configuration of the vocal tract changes like resonant peaks in the vocal track transfer function (frequency response) are known as "formants".

The synthesis is a sort of source-filter-method that is based on mathematical models of the human speech organ. The formant synthesizer makes use of the acoustic-tube model, where the sound is generated from a source, which is periodic for voiced sounds and white noise for obstruent sounds. This basic source signal is then fed into the vocal-tract model. This signal passes into oral cavity and nasal cavity and finally it passes through a radiation component, which simulates the load propagation characteristics to produce speech pressure waveform.

Formant synthesis technology generates artificial and robotic-sounding speech. Formant-synthesized speech is reliably intelligible, even at very high speeds. Formant synthesis is not a very computationally intensive process especially for today's computing systems [5]. The strength of formant synthesis is its relative simplicity and the small memory footprint needed for the engine and its voice data. This acts as main advantage for embedded and mobile computing applications. DecTalk, Apollo, Orpheus and Eloquence are well known TTS engines that use formant synthesis.

3. ARTICULATORY SPEECH SYNTHESIS

Articulatory speech synthesis uses mechanical and acoustic models of speech production to synthesize speech. Articulatory speech synthesis transforms a vector of anatomic or physiologic parameters into a speech signal with predefined acoustic properties [1]. It produces a complete synthetic output, based on mathematical models of the structure (Lips, Teeth,Tongue,Glottis & Velum) processes(transit of airflow along the supraglottal cavities) of speech. This technique is computation-intensive so a memory necessity is almost nothing.

Acoustic models contain number of smaller uniform tubes which generate natural speech. These tubes are controlled by themselves. Natural movements in tubes can give rise to the complex patterns of speech, thus bypassing the problems of modeling complex formant trajectories explicitly. Articulatory synthesis models have an interim stage, in which the motion of the tubes is controlled by some simple process (mechanical damping or filtering), intended to model the fact that the articulators move with a certain inherent speed. This motor-control space is then used to provide the parameters for the specification-to-parameter component.

Two difficulties that arise in articulatory synthesis is how to generate the control parameters form the specification and how to find the right balance between highly accurate model that closely follows human physiology and a more pragmatic representation that is easy to design and control[6].

4. CONCATENATIVE SPEECH SYNTHESIS

Concatenative synthesis depends on speech signal processing of natural speech databases. The segmental database is built to reflect the major phonological features of a language [12]. Concatenation techniques take small units of speech, either waveform data or acoustically parameterized data, and concatenate sequences of these small units together to produce either time varying acoustic parameters or, alternatively, waveforms. The time-varying acoustic parameters then need to be converted into a waveform by passing them through a speech synthesizer.

A concatenation system are concerned with the selection of appropriate units and the algorithms that join those units together and performs some signal processing to smooth unit transitions and to match predefined prosodic schemes. They are three vital subtypes of Concatenative synthesis [9].

(i) Diphone based synthesis
(ii) Domain based synthesis and
(iii) Unit selection based synthesis

(i) Diphone based synthesis

Diphone synthesis is most popular method used for creating a synthetic voice from recordings or samples of a particular person. It uses a nominal speech database[11] .The quantity of diphones in database depends on the phonotactics of the language. In diphone synthesis, the strength of speech depends on sentence or expression and the model used for prosody. The speech may sound a bit monotonic because of improper modeling. A diphone synthesis doesn't work well in languages where there is a lot of inconsequence in the pronunciation rules and in special cases where letters is pronounced differently than in general. The diphone works better for languages that have large consistencies in the pronunciation

The major problem with diphone synthesis is, discontinuities happening at the interface between two bisects of a vowel. In some cases, they create a bi-vocalic sound quality and an perceptible discontinuity at the diphone boundary

(ii) Domain based synthesis

Domain based synthesis concatenates prerecorded vocabulary and axioms to produce entire utterances [1]. It is used in places where output speech is narrowed to a specific domain, like announcement of transit schedule, weather condition reports etc .As these systems are restricted by vocabulary and axioms in their databases, they can only create the combinations of vocabulary and axioms with preprogrammed content.

5. SINUSOIDAL SYNTHESIS

Sinusoidal synthesis uses a harmonic model and decomposes each frame into a set of harmonics of an estimated fundamental frequency [3]. The fundamental parameters like amplitudes, frequencies and phases are changed by keeping the same spectral envelope. The basic idea is to model every significant spectral component as a sinusoid

This model composes harmonic component and a noise component. The first stage in this technique is to classify the frames into voiced and unvoiced portion and dictate the parameters of the harmonic and noise components and the relative strength of the contribution of each component to the frame [7]. Estimated pitch and fundamental frequency is fitted to each frame and from this error between the speech generated and the real waveform is found. Harmonic frames with low error are considered as voice part and frames with high error are considered as noise part.

6. HIDDEN MARKOV MODEL SYNTHESIS

Hidden Markov models (HMMs) is a statistical machine learning speech synthesis to simulate real life stochastic processes [8]. In HMM based synthesis, the speech parameters like frequency spectrum, fundamental frequency and duration are statistically modeled and speech is generated by using HMM based on maximum likelihood criterion. A hidden Markov model is a collection of states connected by transitions. Each transition carries two sets of probabilities: a transition probability, which provides the probability for taking the transition, and an output probability density function, which defines the conditional probability of emitting each output symbol from a finite alphabet, given that the transition is taken.

HMM synthesis provides a means by which to train the specification to parameter module automatically, thus bypassing the problems associated with hand –written rules[12]. The trained models can produce high quality synthesis, and have the advantages of being compact and amenable to modification for voice transformation and other purposes.

This technique consumes large CPU resources but very little memory. This approach seems to give a better prosody, without glitches, and still producing very natural sounding, human-like speech [15].

7. UNIT SELECTION SYNTHESIS

Unit selection synthesis is the dominant synthesis technique in text to speech .This technique is the extension of second generation concatenative technique and deals with the issues of how to manage large numbers of units, how to extend prosody and timing control, and how to alleviate the distortions caused by signal processing [11].

Unit selection technique uses a rich variety of speech, with the aim of capturing more natural variation and relying less on signal processing [5]. The specification and the units are completely described by a feature structure, which can be any mixture of linguistic and acoustic features. During synthesis, an algorithm selects one unit from the possible choices, in an attempt to find the best overall sequence of units which matches the specification and also it enables us to use the carrier speech and lessen the problems arising from designing and recording a database that creates a unit for every feature value.

The greatest difference between a Unit selection and a diphone voice is the length of the used speech segments. There are entire words and phrases stored in the unit database. This implies that the database for the Unit selection voices is many times bigger than for diphone voices [13]. Thus, the memory consumption is huge while the CPU consumption is low

8. CONCLUSION

Speech synthesis is a developing technology which has been incorporated in many real time applications. Existing speech synthesis technique produces quite intelligible and acceptable level of speech output [11] .There is still a long way to reach goal .A number of advances in the area of NLP or DSP have recently boosted up the quality and naturalness of available voices. However researcher have to concentrate on certain areas like prosodic, text preprocessing and pronunciation in order to produce natural and pleasant speech, improve voice quality and linguistic analysis.

In Formant synthesis, rules are needed to specify timing of the source and the dynamic values of all filter parameters which is difficult for simple words. Each synthesis technique has its own limitations and it can be selected depending on the applications. Articulatory synthesis produces intelligible speech, but its output is far from natural sounding .Collecting articulatory data is a costly and fairly invasive process. Articulatory synthesis is appealing for scientific purposes and may one day provide completely “tunable” high quality speech. In Unit selection synthesis, determination of high–level linguistic features are easy, but lack natural distances and can lead to an explosion in feature combinations.HMM synthesis produce speech that is smooth but of poor voice quality however, it has several advantages over unit selection synthesis like flexibility, small footprint, can combine with new techniques to generate new voices with very small training sets.

9. FUTURE DIRECTIONS

Speech synthesis may be used in all kind of human machine interactions. For example, in warning and alarm systems synthesized speech may be used instead of warning lights or buzzers. Speech synthesis takes the major role in mobile phones. The future of text-to-speech can be evolved by improving the following properties.

Text analysis *:* TTS is entirely data-driven and the recent advances in statistical NLP can be used for search engine and document translation .I think many of these techniques are directly applicable to TTS and can be adopted.

Synthesis Algorithms: Its harder to predict concern speech synthesis algorithms. A few years ago, formant synthesis ,concatenative synthesis seemed to be the dominant technique, but recently HMM synthesis and Unit synthesis become dominant .I believe that each speech synthesis techniques has its own weight age according to their applications**.**

Quality Improvements*:* In terms of Overall quality and performance, the problems of text analysis can be fully solved with today's technology. The researchers have to concentrate on good quality databases.

Relationship with linguistics: Overall performance of speech synthesis can be increased by improving the relationship with linguistics.

10. REFERENCES

[1]. L.R.Rainer,"Applications of Voice processing to Telecommunications", *Proc.IEE,Vol.82,PP.199-228,1994.*

[2]. J.Allen,S.Hunncutt and D.H. Klatt,"From Text – to – Speech:The MItalk Systems", *Cambridge university Press, Cambridge, 1987.*

[3]. Hon,H.Acero,A.Huang,X.,Liu,J.,and plumpe,M."Automatic Generation of Synthesis units for Trainable Text-to-speech Systems" . *In proceedings of the IEEE international conference on Acousitics, Speech and Singnal Processing 1998.*

[4]. T.Styger ,E Keller, "Format Synthesis", *Fundamental of Speech Synthesis and Speech Recognition; Basi concept,State of the Art and future challenges (PP.109-128).*

[5]. D.H.Klatt "Review of Text – to- speech Conversion for English", *Journal of the Accoustical Society of America, Vol. 82(3),1987.*

[6]. B.Kroger, "Minimal Rules for Articulatory Speech Synthesis", *Proceedings of EUSIPCO92,pp,331-334,1992.*

[7]. Y.Stylianao, "Modeling Speech Based on the Harmoni Plus Noise Models,"*Springes 2005.*

[8]. K.Tucodo et al., "Hidden Semi-Marrov model based speech synthesis", *Inter Speech PP.1185-1180,2004.*

[9]. Moulines,E., and Charpertier, F."Pitch Synchronous waveform processing techniques for Text- to –Speech Synthesis using diphones." *Speech Communications 9,pp 453-67.1990.*

[10]. Garcia,G.J. Pampin,1999," Data Compression of Sinusoidal modeling parameters based on psychoaccousitc marking", *Proc.ICMC. International computer Music Conference,Beijing,China.*

[11]. A.Hunt and A.Black, " Unit selection in Concatenative Speech Synthesis system using large speech database," *Proc IEEE Int, Conf-Accoust., Speech,Signal Processing,pp 373-376,1996.*

[12]. A.Black and Po Taylor,"The Festival Speech Synthesis System:system documentation,Technical Report" *HCHC/TR – 8s, 1997.*

[13]. Wouter. J., and Macon, M. W. "Unit fusion for concatenative speech synthesis". *In proceedings of the International Conference on Spoken Language Proceedings 2000(2000).*

[14]. Yamagishi, J., Onishi,K., Masuko, T., and Kobayashi, T. "Modeling of Various Speaking styles and emotions for HMM-Based speech synthesis". *In Proceedings of Eurospeech 2003(2003).*

[15]. Zen.H. Tokuda, K.Masuko, T., Kobayashi, T., and Kitamura, T.Hidden "Semi-Makov model based synthesis." *In the Proceedings of 8th International Conference on Spoken Language Processing, Interspeech 2004(2004).*

[16]. *Othman.O.Khalifa,et al "SMA Talk: Standard malay Text to Speech Talk System"- Signal Processing :An International Journal (SPIJ),Vol:2,Issue:5,pp:1-26,2008.*

Finite Wordlength Linear-Phase FIR Filter Design Using Babai's Algorithm

Andraž Božiček *andraz.bozicek@fri.uni-lj.si*
Faculty of Computer and Information Science
University of Ljubljana
Ljubljana, 1000, Slovenia

Abstract

Optimal finite linear-phase impulse response (FIR) filters are most often designed using the Remez algorithm, which computes so-called infinite precision filter coefficients. In many practical applications, it is necessary to represent these coefficients by a finite number of bits. The problem of finite wordlength linear-phase filters is not as trivial as it would seem. The simple rounding of coefficients computed by the Remez algorithm gives us a suboptimal filter. Optimal finite wordlength linear-phase FIR filters are usually designed using integer linear programming, which takes a lot of time to compute the coefficients. In this paper, we introduce a new approach to the design of finite wordlength FIR filters using very fast Babai's algorithm. Babai's algorithm solves the closest vector problem, and it uses the basis reduced by the LLL algorithm as an input. We have used algorithms which solve the problem in the L_2 norm and then added heuristics that improve the results relative to the L_∞ norm. The design method with Babai's algorithm and heuristics has been tested on filters with different sets of frequency-domain specifications.

Keywords: FIR filter design, finite wordlength coefficients, Babai's algorithm, LLL algorithm, closest vector problem.

1. INTRODUCTION

The design of optimal finite wordlength finite impulse response (FIR) filters can be formulated as the Chebyshev approximation problem [1]. It is viewed as a criterion that the weighted approximation error between the desired and the actual frequency response is spread evenly accross the passband and stopband of the filter. This criterion minimizes the maximum absolute error. There exist very good approximation algorithms (including the well-known Remez algorithm), which give the optimal polynomial coefficients in the L_∞ norm. Standard approximation algorithms yield unbounded or so-called "infinite precision" coefficients. In many practical situations, we want to use cheaper and faster fixed-point digital signal processors (DSPs).

There are many approaches which yield the finite wordlength linear-phase coefficients, but not all are optimal. The most simple approach is to round coefficients calculated by the Remez algorithm to a desired length. However, this results in poor frequency response of the filter and suboptimal coefficients [2]. Kodek [2] proposes the use of the mixed integer linear programming (MILP) technique in the design of finite wordlength linear-phase FIR filters to give optimal coefficients. The slowness is the only disadvantage of this technique. An approach which significantly speeds up the calculation of coefficients is represented in [3]. By knowing the lower bound of approximation error, the number of subproblems can be reduced. This also reduces the amount of calculation. Derivation of an improved lower bound that uses the LLL algorithm is given in [4].

As was mentioned earlier, we can formulate the problem of optimal finite wordlength linear-phase FIR filter design as a polynomial approximation. The polynomial approximation has been solved with approaches based on the lattice theory algorithms, i.e. the LLL algorithm and Babai's nearest plane algorithm [5]. The LLL algorithm is a polynomial-time lattice reduction algorithm, named after its three authors. The formal description of the algorithm is in [6], and the implementation of the algorithm, including the pseudo-code, can be found in [7]. The aim of the LLL algorithm is to reduce the lattice basis, so the new lattice is equally described with shorter and almost orthogonal vectors. The LLL algorithm has been successfully used in many areas, including practical applications such as cryptography, GPS navigation, and wireless communications. Babai's nearest plane algorithm solves the closest vector problem [8] and it uses the LLL reduced basis as the input.

2. OPTIMAL FINITE WORDLENGTH LINEAR-PHASE FIR FILTERS

The frequency response 1 of an optimal infinite precision linear-phase FIR digital filter of length N is equal to

$$H_r(\omega) = Q(\omega)P(\omega), \tag{1}$$
$$P(\omega) = \sum_{k=0}^{L} \alpha(k)\cos\omega k, \tag{2}$$

where $Q(\omega)$ is a real function and $\alpha(k)$ are the coefficients of the filter depending on filter length (odd or even) and filter symmetry (positive or negative). There are exactly four types of linear-phase FIR filters. The upper limit L in the sum is $L = \frac{M-1}{2}$ for type 1 filters, $L = \frac{M-3}{2}$ for type 3 filters, and $L = \frac{M}{2} - 1$ for type 2 and type 4 filters. M is defined as the filter length. We also define the desired frequency response $H_{dr}(\omega)$ (which is defined to be unity in the passband and stopband of the filter) and a weighting function $W(\omega)$ (which allows us to choose the relative size of the approximation error in the passband and stopband of the filter). For mathematical convenience, a modified weighting function $\widehat{W}(\omega)$ and a modified desired frequency response are defined as

$$\widehat{W}(\omega) = W(\omega)Q(\omega) \tag{3}$$
$$\widehat{H}_{dr}(\omega) = \frac{H_{dr}(\omega)}{Q(\omega)} \tag{4}$$

The weighted approximation error is defined as

$$E(\omega) = W(\omega)[H_{dr}(\omega) - H_r(\omega)] = \widehat{W}(\omega)\left[\widehat{H}_{dr}(\omega) - \sum_{k=0}^{L}\alpha(k)\cos\omega k\right] \tag{5}$$

To determine the filter cofficients $\alpha(k)$, the following minimax approximation problem has to be solved

$$\min_{P(\omega)}\left[\max_{\omega\epsilon S}\left|\widehat{W}(\omega)\left[\widehat{H}_{dr}(\omega) - \sum_{k=0}^{L}\alpha(k)\cos\omega k\right]\right|\right] \tag{6}$$

The set S consists of the passbands and stopbands of the desired filter, e.g. $S = \Omega_p \cup \Omega_s$.
To make the finite wordlength constraint, the filter coefficients $\alpha(k)$ are b-bit integers from the set I_b, where

$$I_b = \{-2^{b-1}, \ldots, -1, 0, 1, \ldots, 2^{b-1}\}. \tag{7}$$

The most significant bit of the coefficient represents the sign and the other $b-1$ bits represent the magnitude.

3. POLYNOMIAL APPROXIMATION AND FINITE WORDLENGTH LINEAR FIR FILTER DESIGN USING BABAI'S ALGORITHM AND HEURISTICS

To represent an arbitrary non-trivial mathematical function on a computer, one usually uses its approximation. The approximation problem can be defined as a search for a function $g(x)$, which belongs to a given class of functions and is as close as possible to a function $f(x)$. Because there

exist efficient schemes of polynomial evaluation, the approximation function $g(x)$ usually belongs to a class of polynomials.
The typical approximation problem is to search the polynomial $g(x)$ degree $\leq n$ which is sufficiently close to $f(x)$. The distance between functions is defined using the norm. The quality of the approximation is measured with the norm of the remainder $\|f - g\|$. Different norms have different approximation functions. The most usual are the L_2 and L_∞ norms. If a continuous function $f(x)$ on an interval $[a, b]$ is assumed, then the approximation problem can be defined as

the L_2 norm searching the $g(x)$, which minimizes

$$\|f(x) - g(x)\|_2 = \sqrt{\int_a^b |f(x) - g(x)|^2 dx} \tag{8}$$

the L_∞ norm searching the $g(x)$, which minimizes

$$\|f(x) - g(x)\|_\infty = \max_{a \leq x \leq b} |f(x) - g(x)| \tag{9}$$

The approximation polynomial can be a trigonometric polynomial. Finite wordlength FIR filter design can be formulated as the problem of approximation by sums of cosines.

According to (5) and (6) the aim of the polynomial approximation is that the desired frequency response $\tilde{H}_{dr}(\omega)$ and polynomial $P(\omega)$ are as close as possible. If $P(\omega)$ is represented as the vector $\vec{v}$ and $\tilde{H}_{dr}(\omega)$ as the vector $\vec{y}$

$$\underbrace{\begin{pmatrix} \frac{\alpha_0}{2^{b-1}} + \frac{\alpha_1 \cos\omega_1}{2^{b-1}} + \cdots + \frac{\alpha_{L-1}\cos(L-1)\omega_1}{2^{b-1}} + \frac{\alpha_L \cos L\omega_1}{2^{b-1}} \\ \frac{\alpha_0}{2^{b-1}} + \frac{\alpha_1 \cos\omega_2}{2^{b-1}} + \cdots + \frac{\alpha_{L-1}\cos(L-1)\omega_2}{2^{b-1}} + \frac{\alpha_L \cos L\omega_2}{2^{b-1}} \\ \vdots \\ \frac{\alpha_0}{2^{b-1}} + \frac{\alpha_1 \cos\omega_l}{2^{b-1}} + \cdots + \frac{\alpha_{L-1}\cos(L-1)\omega_l}{2^{b-1}} + \frac{\alpha_L \cos L\omega_l}{2^{b-1}} \end{pmatrix}}_{\vec{v}} \text{ and } \underbrace{\begin{pmatrix} \widetilde{H_{dr}}(\omega_1) \\ \widetilde{H_{dr}}(\omega_2) \\ \vdots \\ \widetilde{H_{dr}}(\omega_l) \end{pmatrix}}_{\vec{y}}, \tag{10}$$

then we wish that vectors $\vec{v}$ and $\vec{y}$ are as close as possible according to the L_∞ norm. Frequencies $\omega_1, \omega_2, \ldots, \omega_l$ represents equally discretized frequencies in the interval $[0, \pi]$. The transition band is, unlike to the Remez design method, observed in the interval. If $P(\omega)$ is rewritten as vectors

$$\alpha_0 \underbrace{\begin{pmatrix} \frac{1}{2^{b-1}} \\ \frac{1}{2^{b-1}} \\ \vdots \\ \frac{1}{2^{b-1}} \end{pmatrix}}_{\vec{v}_0} + \alpha_1 \underbrace{\begin{pmatrix} \frac{\cos\omega_1}{2^{b-1}} \\ \frac{\cos\omega_2}{2^{b-1}} \\ \vdots \\ \frac{\cos\omega_l}{2^{b-1}} \end{pmatrix}}_{\vec{v}_1} + \cdots + \alpha_{L-1} \underbrace{\begin{pmatrix} \frac{\cos(L-1)\omega_1}{2^{b-1}} \\ \frac{\cos(L-1)\omega_2}{2^{b-1}} \\ \vdots \\ \frac{\cos(L-1)\omega_l}{2^{b-1}} \end{pmatrix}}_{\vec{v}_{L-1}} + \alpha_L \underbrace{\begin{pmatrix} \frac{\cos L\omega_1}{2^{b-1}} \\ \frac{\cos L\omega_2}{2^{b-1}} \\ \vdots \\ \frac{\cos L\omega_l}{2^{b-1}} \end{pmatrix}}_{\vec{v}_L}, \tag{11}$$

then integer coefficients $\alpha_0, \ldots, \alpha_L$ that minimize

$$\|\vec{v} - \vec{y}\|_\infty = \|\alpha_0 \vec{v}_0 + \alpha_1 \vec{v}_1 + \cdots + \alpha_L \vec{v}_L - \vec{y}\|_\infty \tag{12}$$

have to be found.

Problem (11) can be formulated as the closest vector problem in the L_∞ norm. Kannan's algorithm solves this problem in L_∞, but its complexity is super-exponential [5]. In practice it is better to use Babai's algorithm, which solves the problem in the L_2 norm, and then use some heuristics to improve the results relative to the L_∞ norm.

Babai's algorithm returns the vector $\vec{v}$ as the result. In [5], heuristics similar to that used in this paper are described. The heuristics assume that the result in the L_∞ norm is close to the result in the L_2 norm. The heuristics search the neighborhood of the vector $\vec{v}$. Vectors which represent polynomial $P(\omega)$ (and are LLL-reduced) are added and subtracted from the vector $\vec{v}$. In this manner, we get candidates for a result which is, according to the L_∞ norm, better than the previous result. The L_∞ norm is calculated between the candidates in the desired frequency response, and if the L_∞ norm is lower than it was for the previous result than we keep the new result. This procedure is repeated until the result is improved. The heuristics are described with the pseudocode presented in Table 1.

Description: Explore the neighborhood of the vector $\vec{v}$ to get close to the desired frequency response (vector $\vec{y}$) according to the L_∞ norm ***Input:*** vector $\vec{v}$, vector $\vec{y}$, LLL-reduced basis B' The vector $\vec{v}$ represents the frequency response of the filter and is according to the L_2 norm as close as possible to the desired frequency response of the filter. The vector $\vec{y}$ represents the desired frequency response. The LLL-reduced basis B' includes vectors $\vec{v}_0', \vec{v}_1', \ldots, \vec{v}_L'$. ***Output:*** vector $\vec{v}$. Set Z is filled with $2L+2$ vectors: $\vec{z}_0 = \vec{v} + \vec{v}_0', \vec{z}_1 = \vec{v} - \vec{v}_0', \vec{z}_2 = \vec{v} + \vec{v}_1', \vec{z}_3 = \vec{v} - \vec{v}_1', \ldots, \vec{z}_{2L} = \vec{v} + \vec{v}_L', \vec{z}_{2L+1} = \vec{v} - \vec{v}_L'$. Reference norm is calculated $ref = \Vert\vec{v} - \vec{y}\Vert_\infty$. For $i = 0$ to $2L+1$ calculate $\delta_i = \Vert\vec{z}_i - \vec{y}\Vert_\infty$. If exists $\min_{i=1\ldots 2L+1} \delta_i$ such that $\delta_i < ref$, then new $\vec{v} = \vec{z}_i$ and goto 2. Else return $\vec{v}$.

TABLE 1: Heuristics pseudocode

The disadvantage of the finite wordlength linear-phase FIR filter design method described is that the weighting function $\widehat{W}(\omega)$ cannot be observed in the design process.

4. RESULTS

Our design method has been tested on 25 filters with five different sets of frequency-domain specifications. The frequency domain specifications are given in Table 2.

Set	Band 1	Band 2	Band 3
A	$\Omega_p = [0;\ 0{,}40\pi]$ pass $W(\omega_p) = 1$	$\Omega_s = [0{,}50\pi;\ \pi]$ stop $W(\omega_s) = 1$	
B	$\Omega_p = [0;\ 0{,}40\pi]$ pass $W(\omega_p) = 1$	$\Omega_s = [0{,}50\pi;\ \pi]$ stop $W(\omega_s) = 10$	
C	$\Omega_{p1} = [0;\ 0{,}24\pi]$ pass $W(\omega_{p1}) = 1$	$\Omega_s = [0{,}40\pi;\ 0{,}68\pi]$ stop $W(\omega_s) = 1$	$\Omega_{p2} = [0{,}84\pi;\ \pi]$ pass $W(\omega_{p2}) = 1$
D	$\Omega_{p1} = [0;\ 0{,}24\pi]$ pass $W(\omega_{p1}) = 1$	$\Omega_s = [0{,}40\pi;\ 0{,}68\pi]$ stop $W(\omega_s) = 10$	$\Omega_{p2} = [0{,}84\pi;\ \pi]$ pass $W(\omega_{p2}) = 1$
E	$\Omega_p = [0{,}02\pi;\ 0{,}42\pi]$ pass $W(\omega_p) = 1$	$\Omega_s = [0{,}52\pi;\ 0{,}98\pi]$ stop $W(\omega_s) = 1$	

TABLE 2: Characteristics of filter test sets

Initially, filter A with 45 eight-bit coefficients was synthesized. As can be seen in Table 3, the design method with Babai's algorithm and heuristics did not give us the optimal filter, but the result was better than using rounding of coefficients calculated by the Remez algorithm. The calculation time of Babai's algorithm and heuristics was very short compared to the MILP method. The heuristics have also slightly improved the result of Babai's algorithm. Fig. 1 shows the magnitude response for the above filter designed using the MILP and Babai's algorithm with heuristics.

	Rounding	**Babai's algorithm**	**Babai's algorithm with heuristics**	**MILP**
Max. deviation	0.037059	0.032884	0.032668	0.028847
Passband	0.03125 (0.267279 dB)	0.029542 (0.252882 dB)	0.032668 (0.279217 dB)	0.028847 (0.247016 dB)
Stopband	0.037059 (28.622046 dB)	0.032884 (29.660432 dB)	0.031262 (30.099566 dB)	0.027691 (31.153227 dB)
Calculation time	0.064 s	0.192 s	0.210 s	44.431 s

TABLE 3: Results for filter A, 45 eight-bit coefficients

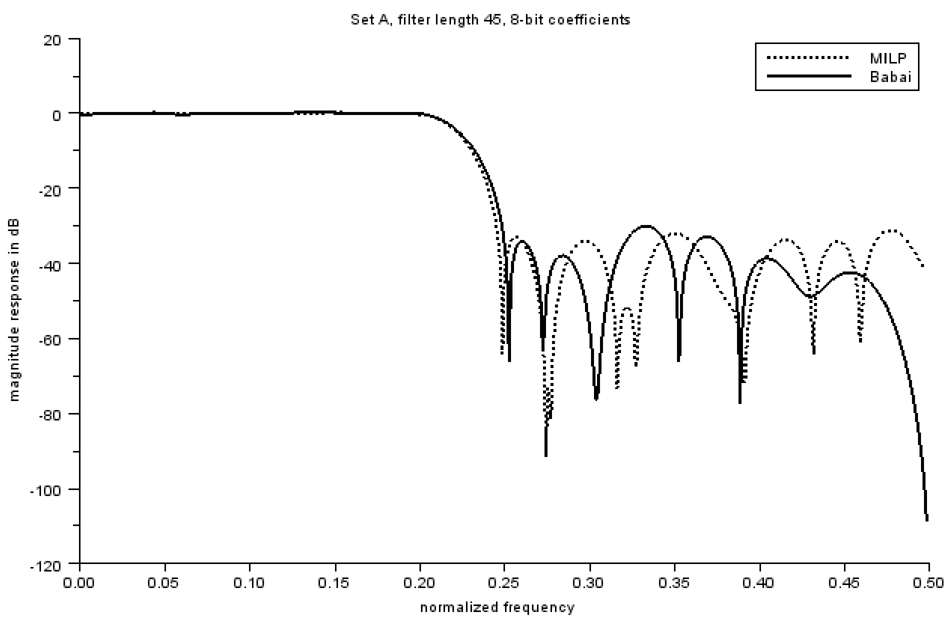

FIGURE 1: Magnitude response for filter from set A (length 45, 8-bit coefficients)

The non-unit weighting function could not be used in the design process. When designing filters from sets B and D, the magnitude response from the Remez algorithm was approximated. In these cases, the results were not very good. In all other cases, results were better than using the simple rounding technique, and in some cases optimal filter coefficients were calculated. The results can be seen in Table 4.

Set			Rounding	Babai's algorithm with heuristics	MILP
A	M=25, b=7	max. deviation	0.078125	0.065237	0.065237
	M=35, b=8	max. deviation	0.032668	0.032668	0.029966
	M=45, b=8	max. deviation	0.037059	0.032668	0.028847
	M=45, b=10	max. deviation	0.013872	0.010182	0.010182
	M=55, b=10	max. deviation	0.00979	0.009144	0.008296
B	M=25, b=8	passband deviation	0.140625	0.143501	0.154246
		stopband deviation	0.032914	0.039063	0.015234
	M=35, b=9	passband deviation	0.074219	0.066406	0.073276
		stopband deviation	0.015902	0.022931	0.007313
	M=45, b=9	passband deviation	0.03801	0.034366	0.052641
		stopband deviation	0.011719	0.015908	0.005681
	M=45, b=11	passband deviation	0.024179	0.025522	0.026655
		stopband deviation	0.006177	0.006282	0.002673
	M=55, b=11	passband deviation	0.010579	0.011861	0.01666
		stopband deviation	0.006234	0.004939	0.001643
C	M=25, b=7	max. deviation	0.041507	0.036676	0.036676
	M=35, b=8	max. deviation	0.046875	0.03125	0.016767
	M=45, b=8	max. deviation	0.030466	0.029409	0.016085
	M=45, b=10	max. deviation	0.00993	0.006843	0.004761
	M=55, b=10	max. deviation	0.010839	0.007636	0.004365
D	M=25, b=8	passband 1 deviation	0.0625	0.0625	0.078125
		stopband deviation	0.014408	0.014408	0.007966
		passband 2 deviation	0.0625	0.0625	0.078125
	M=35, b=9	passband 1 deviation	0.015987	0.023438	0.03125
		stopband deviation	0.012202	0.011957	0.003302
		passband 2 deviation	0.027344	0.023438	0.03125
	M=45, b=9	passband 1 deviation	0.01768	0.014144	0.022836
		stopband deviation	0.010906	0.011689	0.002552
		passband 2 deviation	0.018901	0.011097	0.025071
	M=45, b=11	passband 1 deviation	0.004244	0.003761	0.006396
		stopband deviation	0.003222	0.002954	0.000745
		passband 2 deviation	0.003906	0.004551	0.005306
	M=55, b=11	passband 1 deviation	0.002167	0.002167	0.00518
		stopband deviation	0.00371	0.00371	0.000618
		passband 2 deviation	0.004379	0.004379	0.005859
E	M=25, b=7	max. deviation	0.077633	0.062141	0.06071
	M=35, b=8	max. deviation	0.046934	0.033004	0.032888
	M=45, b=8	max. deviation	0.035784	0.035784	0.028987
	M=45, b=10	max. deviation	0.015263	0.011185	0.010104
	M=55, b=10	max. deviation	0.011802	0.009119	0.008199

TABLE 4: Results

5. CONCLUSION

A new method using Babai's algorithm and heuristics for the design of finite wordlength linear-phase FIR filters was presented in this paper. Heuristics were used to improve the Babai's algorithm result and to bring the result closer to the L_∞ norm. Testing showed that the heuristics improve the result of Babai's algorithm. The major advantage of this design method is the speed of the algorithm. Major disadvantages are that we do not get always the optimal filter coefficients and we cannot design filters with non-unit weighting functions. These will be the main focus of our future research, and our aim is to minimize those disadvantages.

Our future research will also attempt to include some other concepts. The problem of searching the closest point in a lattice is in communications community referred as the sphere decoding. Using results of [9], [10] some of the improvements over the Babai's algorithm could be reached. In [11], [12] the peak constrained least squares method is introduced. This method balances the minimax and the squared error criteria. This approach could be useful framework to pose the filter design problem.

6. REFERENCES

[1] J. G. Proakis and D. G. Manolakis, *Introduction to Digital Signal Processing*., pp. 614-726.

[2] D. M. Kodek, "Design of optimal finite wordlength FIR digital filters using integer programming techniques," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 38, no. 3, pp. 304- 308, junij 1980.

[3] D. M. Kodek, "Performance Limit of Finite Wordlength FIR Digital Filters," *IEEE Transactions on Signal Processing*, vol. 53, no. 7, pp. 2462- 2469, julij 2005.

[4] D. M. Kodek, "LLL Algorithm and the Optimal Finite Wordlength FIR Design," *IEEE Transactions on Signal Processing*, vol. 60, no. 3, pp. 1493-1498 , March 2012.

[5] N. Brisebarre and S. Chevillard, "Efficient polynomial L-approximations," *Computer Arithmetic, 2007. ARITH '07. 18th IEEE Symposium on*, pp. 169-176, junij 2007.

[6] A. K. Lenstra, H. W. Lenstra, and L. Lovász, "Factoring polynomials with rational coefficients," *Mathematische Annalen*, vol. 264, no. 4, pp. 515-634, december 1982.

[7] N. P. Smart, *The algorithmic resolution of diophantine equations*.: Cambridge University Press, 1998, pp. 59-76.

[8] L. Babai, "On Lovász' Lattice Reduction and the Nearest Lattice Point Problem," in *Proceedings of the 2nd Symposium of Theoretical Aspects of Computer Science (STACS '85)*, London, 1985, pp. 13-20.

[9] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger, "Closest point search in lattices," *IEEE Transactions on Information Theory*, vol. 48, no. 8, pp. 2201-2214, August 2002.

[10] A. D. Murugan, H. El Gamal, M. O. Damen, and G. Caire, "A unified framework for tree search decoding: rediscovering the sequential decoder," *IEEE Transactions on Information Theory*, vol. 52, no. 3, pp. 933-953 , March 2006.

[11] J. W. Adams and J. L. Sullivan, "Peak-constrained least-squares optimization," *IEEE Transactions on Signal Processing*, vol. 46, no. 2, pp. 306-321, February 1998.

[12] I. W. Selesnick, M. Lang, and C. S. Burrus, "Constrained least square design of FIR filters without specified transition bands," *IEEE Transactions on Signal Processing*, vol. 44, no. 8, pp. 1879-1892, August 1996.

INSTRUCTIONS TO CONTRIBUTORS

The *International Journal of Signal Processing (SPIJ)* lays emphasis on all aspects of the theory and practice of signal processing (analogue and digital) in new and emerging technologies. It features original research work, review articles, and accounts of practical developments. It is intended for a rapid dissemination of knowledge and experience to engineers and scientists working in the research, development, practical application or design and analysis of signal processing, algorithms and architecture performance analysis (including measurement, modeling, and simulation) of signal processing systems.

As SPIJ is directed as much at the practicing engineer as at the academic researcher, we encourage practicing electronic, electrical, mechanical, systems, sensor, instrumentation, chemical engineers, researchers in advanced control systems and signal processing, applied mathematicians, computer scientists among others, to express their views and ideas on the current trends, challenges, implementation problems and state of the art technologies.

To build its International reputation, we are disseminating the publication information through Google Books, Google Scholar, Directory of Open Access Journals (DOAJ), Open J Gate, ScientificCommons, Docstoc and many more. Our International Editors are working on establishing ISI listing and a good impact factor for SPIJ.

The initial efforts helped to shape the editorial policy and to sharpen the focus of the journal. Started with Volume 6, 2012, SPIJ appears with more focused issues related to signal processing studies. Besides normal publications, SPIJ intend to organized special issues on more focused topics. Each special issue will have a designated editor (editors) – either member of the editorial board or another recognized specialist in the respective field.

We are open to contributions, proposals for any topic as well as for editors and reviewers. We understand that it is through the effort of volunteers that CSC Journals continues to grow and flourish.

SPIJ LIST OF TOPICS

The realm of Signal Processing: An International Journal (SPIJ) extends, but not limited, to the following:

- Biomedical Signal Processing
- Communication Signal Processing
- Detection and Estimation
- Earth Resources Signal Processing
- Industrial Applications
- Optical Signal Processing
- Radar Signal Processing
- Signal Filtering
- Signal Processing Technology
- Software Developments
- Spectral Analysis
- Stochastic Processes
- Acoustic and Vibration Signal Processing
- Data Processing
- Digital Signal Processing
- Geophysical and Astrophysical Signal Processing
- Multi-dimensional Signal Processing
- Pattern Recognition
- Remote Sensing
- Signal Processing Systems
- Signal Theory
- Sonar Signal Processing
- Speech Processing

CALL FOR PAPERS

Volume: 7 - **Issue: 1**

i. Paper Submission: December 31, 2012 **ii. Author Notification:** January 31, 2013

iii. Issue Publication: February 2013

CONTACT INFORMATION

Computer Science Journals Sdn BhD
M-3-19 Plaza Damas, Sri Hartamas
50480, Kuala Lumpur, MALAYSIA

Phone: 006 03 6207 1607
Fax: 006 03 6207 1697

Email: cscpress@cscjournals.org

COMPUTER SCIENCE JOURNALS SDN BHD
M-3-19, PLAZA DAMAS
SRI HARTAMAS
50480, KUALA LUMPUR
MALAYSIA

PHONE: 006 03 6207 1607
FAX: 006 03 6207 1697
EMAIL: cscpress@cscjournals.org