

Perceptually Guided Inexact DSP Design for Power, Area Efficient Hearing Aid

Sai Praveen Kadiyala*, Aritra Sen*, Shubham Mahajan, Qingyun Wang*, Avinash Lingamneni†, James Sneed German‡, Xu Hong‡, Ansuman Banerjee, Krishna V. Palem†, and Arindam Basu*

*School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore

†Department of Computer Science, Rice University, Houston, USA

‡School of Humanities and Social Sciences, Nanyang Technological University, Singapore

Email: arindam.basu@ntu.edu.sg

Abstract—*Inexact design* has been recognized as very viable approach to achieve significant gains in the energy, area and speed efficiencies of digital circuits. By deliberately trading error in return for such these gains, inexact circuits and architectures have been shown to be especially useful in contexts where our senses such as sight and hearing, can compensate for the loss in accuracy. It is therefore important to understand, characterize the manner in which our sensorial systems interact and compensate for the loss in accuracy. Further use this knowledge to optimize and guide the manner in which inexactness is introduced. For the first time, we achieve both of these goals in this paper in the context of human audition—specifically, using the architecture of a hearing-aid and the DSP primitive of an FIR filter as our candidate. Our algorithms for designing an inexact hearing-aid thus use intelligibility as the metric. The resulting inexact FIR filter in the hearing aid is 1.5X or 1.8X more efficient in terms of power-area product while producing 5% or 10% less intelligible speech respectively when compared with the corresponding exact version.¹

I. INTRODUCTION AND MOTIVATION

Recently, there is a huge interest in developing low power wearable sensors for medical applications and power aware digital designs are playing a crucial role in this process. Most of the available digital processors however do not take the advantage of *information processing* done by the natural auditory and visual path ways, hence leaving a scope to dig in this direction. The ability of human compensatory neurocognitive processing to tolerate relatively less accurate inputs may provide an excellent opportunity to lower the power and area requirement of processors in assistive devices by deliberately introducing errors in computation. This novel approach termed as *inexact design* has proved to yield significant gains in the context of hardware for DSP primitives [1], atmospheric modelling [3], MPEG coding [4] and recognition and classification tasks [5].

However, past work on inexact circuits has quantified the glitches or errors introduced by arithmetic magnitude—as opposed to using metrics that capture their impact on our senses in a natural way. Naturally, incorporating this effect adds to the complexity of design significantly since the human designer has to bridge the gap between the arithmetic error on the one hand, and its impact on our senses on the other. To remedy this situation, in this paper, we present the first result of

its kind wherein a neurocognitive model of human hearing is used to quantify glitches or loss in accuracy, through a novel metric of *intelligibility* of spoken sound. The model and the intelligibility metric are used to guide the design of an inexact processor for an assistive device—a hearing aid. Using the techniques described earlier, we can get reduction in the power area product (PAP) of the filter bank by 1.8X for an intelligibility loss of only 10% over conventional exact designs.

II. PERCEPTUALLY GUIDED PRUNING FOR EFFICIENT INEXACT CIRCUITS

Probabilistic Pruning [1], [2] is an inexact design technique that exploits the knowledge of the significance of a circuit component to derive a systematic approach to prune the “least useful” components in a circuit. In this paper, we will use this technique as the basis for introducing “inexactness” and enabling the energy-accuracy tradeoffs. The novel contributions of this paper in the area of pruning is the introduction of an efficient optimization strategy for pruning that is scalable to large digital systems.

A. Optimization Framework

Earlier work on pruning [2] has considered the removal of gates in arithmetic circuits like adders and multipliers by using explicit cost functions. However, such approaches are intractable for large systems because of the non-availability of such explicit cost functions and the severe overheads involved in determining those cost functions computationally at the granularity of individual gates. Hence, we propose to have a library \mathbf{E} of N_E elemental circuits, E_i , each of which can admit only one of several pre-characterized pruned topologies. These topologies can be indexed by an integer l that denotes the level or degree of pruning; larger values of l indicate more savings at the cost of increased error magnitude [2]. Let L_i denote an integer that indicates the maximum level of pruning of $E_i \in \mathbf{E}$ while $l = 1$ corresponds to the unpruned structure. Now, we can define a set \mathbf{C} of all components c_j allowed in our design by denoting $c_j = \{l, E_i\}$ where $E_i \in \mathbf{E}$ and $l(\leq L_i) \in N$. The circuit we want to optimize can be represented as a directed acyclic graph \mathbf{G} whose nodes are N_G components selected from \mathbf{C} , inputs, or outputs and whose edges are wires. We can now formulate an optimization problem where the performance of \mathbf{G} can be modified by

¹Financial support from MOE, Singapore through grant ARC8/13 is acknowledged.

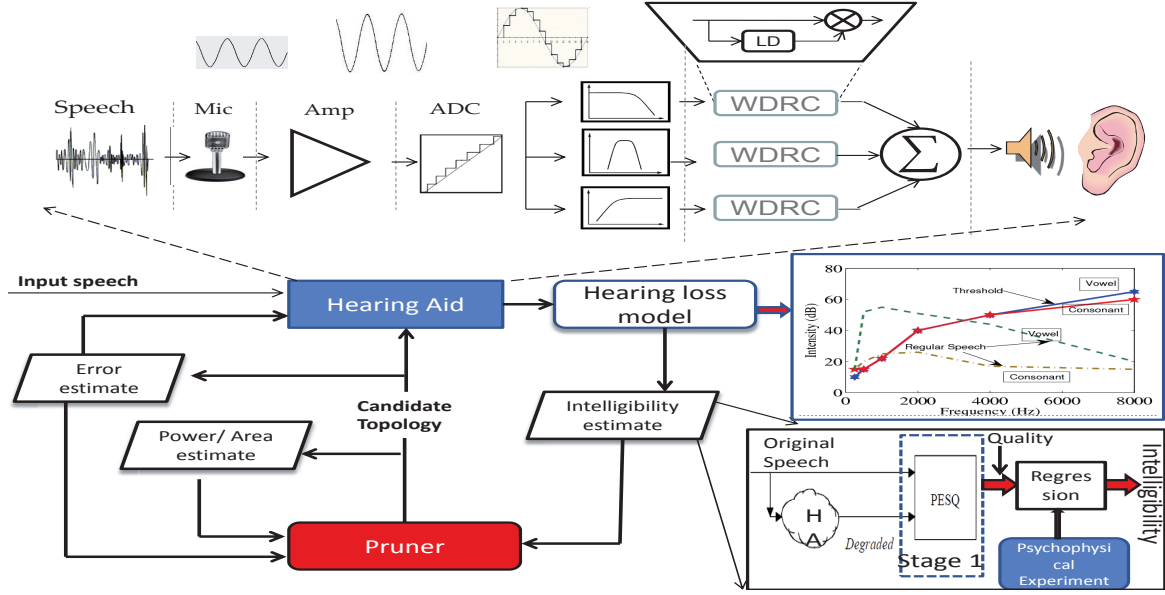


Fig. 1: Framework for evaluation of cost in the context of a hearing-aid and using that to introduce inexactness in the design. The optimization loop uses a library of pre-characterized inexact VLSI components to quickly achieve a near optimal solution which is then evaluated rigorously through detailed synthesis procedures.

varying $\vec{L} = [l_1 l_2 \dots l_{N_G}]$ in exchange of cost savings. To make this dependence explicit, we shall henceforth refer to the graph as $\mathbf{G}(\vec{L})$.

The quality of outputs $\{\mathbf{O}\}$ of $\mathbf{G}(\vec{L})$ for the same set of inputs $\{\mathbf{I}\}$ depends on the value of \vec{L} with the best quality of outputs being obtained for $\vec{L} = [1 \dots 1] = \vec{L}_u$ corresponding to the unpruned circuit. We formally define the performance as a function of \vec{L} by:

$$Per = \sum_{k=1}^{\nu} p_k Q_p(O_k(\vec{L}), O_k(\vec{L}_u)), Per \in \mathbb{R}^+ \quad (1)$$

where $O_k(\vec{L})$ represents the output of $\mathbf{G}(\vec{L})$ for input I_k that occurs with a probability p_k for $1 \leq k \leq \nu$. Q_p denotes a function that measures the *perceptual quality* of the output of the pruned circuit with respect to the output of the unpruned one based on models of human sensory processing with larger values denoting better quality. The problem can now be defined as finding the optimal value of \vec{L} , \vec{L}_{opt} such that:

$$\vec{L}_{opt} = \arg \min_{\vec{L}} M \text{ subject to } Per \geq Per_{TH} \quad (2)$$

where M is some cost metric of the circuit (like area or power) that needs to be minimized and Per_{TH} denotes a threshold for minimum acceptable perceptual quality.

B. Greedy Optimization Algorithm

The optimization problem posed in (2) can be solved by a modified greedy version of a gradient descent approach with the following features:

- We evaluate only $s (< N_G)$ randomly selected components of the gradient and this random candidate set is modified every iteration.

- To give preference to those components which can reduce M without compromising Per too much, we modify the cost function to be:

$$C = \frac{M}{Per} + \lambda u(Per - Per_{TH}), Per \neq 0 \quad (3)$$

where λ is a large positive number to prevent choices which reduce the performance below threshold and u is the heaviside function.

- To prevent large jumps in error, we only allow changes in pruning levels of the $q (< s)$ components which have the q largest gradient values by using a ‘rank’ function, $rank_q$ that assigns values $1, 1/2, \dots, 1/q$ to the selected ones while assigning a value of 0 to others. Hence, the final update equation is given by:

$$\vec{L}(n+1) = \vec{L}(n) + rank_q(I_s(n) \nabla C) \quad (4)$$

where I_s is the identity matrix with all rows set to zero other than ‘s’ randomly chosen ones.

III. SIMULATION FRAMEWORK: HEARING AID ARCHITECTURE, HEARING LOSS MODEL, COGNITIVE MODEL AND GAIN/ERROR ESTIMATOR

To demonstrate the operation of this algorithm, we choose a digital hearing aid, which interacts with our auditory sense, as the platform. Figure 1 depicts our entire simulation framework implemented in MATLAB that includes the hearing aid architecture, hearing loss model, cognitive model and gain/error estimator. We shall describe each of them in details in the following sub-sections.

A. Hearing Aid Architecture

The basic architecture of a hearing aid, shown in Fig. 1 has an analog front-end amplifier followed by a wide dynamic range analog-to-digital converter (ADC) with sigma-delta ADC being the most popular choice [7]. This is followed by the digital processor, which typically has two main parts [8]: a filter bank to decompose the speech signal into different sub-bands and a wide dynamic range compressor (WDRC) that compresses the input speech to fit the reduced dynamic range of an impaired ear. In this work, we have focussed on the VLSI implementation of the filter bank which is an important part of this architecture and has been the focus of significant research [9], [10]. The WDRC is implemented as a software module in MATLAB.

Due to the good match with the frequency characteristics of the human ear, ANSI S1.11 1/3-octave filter bank specifications [11] are used to guide the design of the FIR filter bank in this work. ANSI S1.11 standard defines 43 1/3 octave bands covering frequencies 0 – 20 kHz. For our application, we have chosen class-2 filters and the bands 22 – 39 in the ANSI standard that covers the frequency range of 250 Hz to 8 kHz generally assumed to be necessary for the comprehension of speech.

The input speech signal is decomposed into 18 frequency bands by the filter block. Straightforward implementation of the 18 bands of the ANSI specification would involve design of very high order filters since the bandwidths of the bands 22–27 are very low. Hence, we use the multi-rate architecture similar to [9], [10]. The bands 37, 38 and 39 constitute one octave; hence the 18 bands cover 6 octaves.

B. Hearing Loss Model

The parameters of the WDRC in the hearing aid needs to be tuned according to the chosen hearing loss model which is patient specific. For this work, ‘Presbycusis’ is chosen as the hearing loss model since this is one of the most common sensorineural hearing loss problem. The audiogram of a person suffering from ‘Presbycusis’ is shown in Fig. 1. It can be seen that the hearing threshold, defined as the softest or lowest intensity of sounds that one can hear, are more than the intensity of normal speech for certain frequencies of sound. People with such disorders fail to hear sounds in this frequency range.

C. Cognitive Model

Due to human cognitive abilities, the signal to noise ratio (SNR) of a speech sample is not necessarily proportional to its ‘intelligibility’ as perceived by a human listener. To estimate intelligibility, we have developed a two-stage model where the first stage uses a standard method to estimate speech quality while a second custom module is developed on the basis of behavioral experiments to transform the quality metric to intelligibility. Perceptual Evaluation of Speech Quality (PESQ) is a family of standards (ITU-T recommendation P.862) [6] comprising a test methodology for the automated assessment of the speech quality as experienced by a user of a telephony

system. In our case, we use the output of the inexact hearing aid as the degraded speech input to PESQ. The details of the algorithm can be obtained from [6] and references therein. The output of PESQ is a score indicating ‘quality’ of speech which is not necessarily a direct measure of intelligibility—however, the pre-processing performed by PESQ is still relevant for us. Hence, we use this quality metric as a feature that can be mapped to intelligibility as described next.

To obtain the intelligibility of degraded speech, a database of speech samples was created using a corpus of four hundred two-syllable words drawn from the Celex lexical database for English [12]. These words were then corrupted with either white noise or cocktail party noise of different intensities so that the SNR of each sample varied between -10 and 20 dB. This study was approved by the Internal Review Board (IRB) at Nanyang Technological University, Singapore. Five subjects with normal hearing were now chosen for the psychophysics experiment in which they were instructed to listen to a randomly selected set of 100 words with varying levels of SNR and type the word they thought they heard. These results were processed manually to correct for homophones and spelling errors. As can be expected, the correctness of results indicating intelligibility does not change much as long as SNR is high enough (5 dB). However, below a certain threshold (SNR ≈ -2.5 dB), there is a sharp drop in intelligibility. For our final model, we process the same speech samples for different SNR levels used in the test through PESQ to get a perceptual quality metric. Then a polynomial regression method is used to convert this quality score into the percentage correctness or intelligibility obtained in the behavioral experiment.

D. Gain/Error Estimator

For every step of the iterative optimization process, we need to estimate the gains achieved in power and area due to pruning and the corresponding error introduced (shown in Fig. 1). This is done by creating a library of different pruned multipliers and adders which are used in the design. For each of these, a detailed characterization is done to obtain the area and power benefits for each design over the unpruned structure. Table I demonstrates the result for such a characterization of pruned multipliers in $65nm$ CMOS process. The characterization methodology is described in [2].

To get good estimates of the error, we first generate a probability distribution of error at the output of an individual pruned block by comparing the results of a pruned and unpruned circuit in simulations. This was done by observing the error for 10,000 trials with uniformly distributed random inputs for each pruned topology and generating a histogram with 1000 bins. We now generate a random number in MATLAB according to this distribution (by mapping from an uniform random distribution to the desired one through the cumulative distribution function) and add it to the output of each block according to its own error profile.

TABLE I: Characteristics of Pruned Multipliers in 65nm CMOS

Pruning Level	Power (Normalized)	Area (Normalized)	Mean Error	StDev of Error
1	1	1	0	0
2	0.804	0.875	0.122	0.72
3	0.692	0.753	0.373	1.06
4	0.519	0.733	-0.486	1.53
5	0.472	0.711	-0.038	2.62
6	0.458	0.57	-0.540	3.68
7	0.363	0.476	-2.540	7.54
8	0.302	0.472	-3.010	11.50
9	0.245	0.385	-6.490	14.89
10	0.22	0.417	-6.200	22.50
11	0.184	0.323	-14.33	29.81

IV. RESULTS

The algorithm was described in a generic way in Section II; we shall first mention the specific values of the parameters used in our design. Since the area and energy consumed by an array multiplier is ≈ 10 times more than that of a ripple carry adder, we have only $N_E = 1$ library component since multipliers were the only element chosen for pruning. It can be seen from Table I that the maximum level of pruning for a multiplier is $L_1 = 11$. In all, 510 ripple carry adders and array multipliers each make up the arithmetic architecture of the FIR filter implying $N_G = 510$. The performance of the circuit is measured in terms of the Intelligibility (I) which is denoted as Q_p in section II. The final performance metric Per is obtained by averaging the intelligibility over $\nu = 3$ sample words. The circuit metric we want to optimize is calculated in terms of the power (P) and area (A) consumed. Correspondingly, the cost function used is:

$$C_2 = (e^P \times e^A) \setminus I + \lambda u(I - I_{thresh}); \quad (5)$$

where the exponential functions are used to amplify the small changes in P and A compared to I .

The final result of the optimization algorithm is plotted as a tradeoff curve between Intelligibility and PAP in figure 2. In the same figure, the signal to noise ratio (SNR) of the degraded speech samples is also plotted. It can be seen that the SNR of the speech degrades rapidly after a few iterations but the Intelligibility degrades marginally up to 1.8X reduction in PAP. This again points to the fact that the perceptual quality of a speech signal cannot be objectively judged by its SNR. From this curve, we can see that it is possible to operate the FIR filter bank at a savings of 1.5X or 1.8X in PAP while sacrificing 5% or 10% in intelligibility respectively.

V. CONCLUSION AND DISCUSSION

We presented a methodical approach for reducing the cost associated with a digital circuit (e.g., power, area) by factoring in the neurocognitive processing done in our brains on incoming sensory signals. Our model allows one to quickly estimate the effect of inexact design on the user’s experience without having to perform costly field studies. To demonstrate our methodology, we chose a digital hearing aid as the platform with circuit pruning to introduce inexactness and used ‘intelligibility’ of speech as the metric. We introduced a novel greedy heuristic-based pruning strategy that allows us to

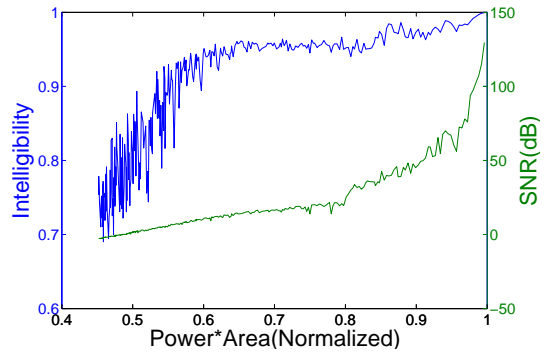


Fig. 2: Plot showing the trade-off between Intelligibility and SNR with the power area product (PAP). The drop in Intelligibility is significantly smaller compared to drop in SNR with decrease in PAP.

prune very large circuits where the optimal solution might be extremely time consuming to find. Using our methods to prune the filter bank in the hearing-aid, we demonstrate 1.5 – 1.8X improvement in performance in terms of power-area product while producing 5 – 10% less intelligible speech.

REFERENCES

- [1] A. Lingamneni, A. Basu, C. Enz, K. Palem, and C. Piguet, “Improving Energy Gains of Inexact DSP Hardware Through Reciprocal Error Compensation,” in *Proceedings of the 50th IEEE/ACM Design Automation Conference*, Austin, Texas, USA, 2013.
- [2] Avinash Lingamneni, Christian Enz, Jean-Luc Nagel, Krishna Palem, and Christian Piguet, “Energy Parsimonious Circuit Design through Probabilistic Pruning,” in *Proceedings of the 14th Design, Automation and Test in Europe (DATE 2011)*, Grenoble, France, Mar. 2011, pp. 764–9.
- [3] Peter D. Duben, et al., “On the Use of Inexact, Pruned Hardware in Atmospheric Modelling,” *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 372, no. 2018, pp. 20130276, 2014.
- [4] Ranjan Ashish, et. al, “ASLAN: Synthesis of Approximate Sequential Circuits,” *Proceedings of the 2014 Design, Automation & Test in Europe Conference*, 2014, pp. 364.
- [5] Venkataramani Swagath, et. al., “AxNN: Energy-efficient Neuromorphic Systems Using Approximate Computing,” *Proceedings of the 2014 International Symposium on Low power Electronics and Design*, 2014, pp. 27–32.
- [6] J. G. Beerends, A. P. Hekstra, A. W. Rix, and M. P. Hollier, “Perceptual Evaluation of Speech Quality (PESQ): The New ITU-T Standard for end-to-end Speech Quality Assessment Part II—Psychoacoustic model,” *J. Audio Eng. Soc.*, vol. 50, no. 10, pp. 765–778, 2002.
- [7] D. George Gata, W. Sjrursen, and J. R. Hochschild et al., “A 1.1-V 270- μ A Mixed-Signal Hearing Aid Chip,” *IEEE Journal Of Solid-State Circuits*, vol. 37, no. 12, pp. 1670–78, 2002.
- [8] J.M. Kates, *Digital Hearing Aids*, Plural Publishing, Incorporated, 2008.
- [9] Yong Lian and Ying Wei, “A computationally efficient nonuniform FIR digital filter bank for hearing aids,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 52, no. 12, pp. 2754–2762, 2005.
- [10] Yu-Ting Kuo, Tay-Jyi Lin, Yueh-Tai Li, and Chih-Wei Liu, “Design and Implementation of Low-Power ANSI S1.11 Filter Bank for Digital Hearing Aids,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 57, no. 7, pp. 1684–1696, 2010.
- [11] ANSI S1.11-2004, *Specification for Octave Band and Fractional Octave band Analog and Digital Filters*, Standards Secretariat Acoustical Society of America, 2004.
- [12] R. H. Baayen, R. Piepenbrock, and L. Gulikers, *The CELEX Lexical Database (CD-ROM)*, Linguistic Data Consortium, Philadelphia, 1995.