AUTOMATIC PARAMETER OPTIMIZATION FOR A PERCEPTUAL AUDIO CODEC

Martin Holters, Udo Zölzer

Helmut Schmidt University — University of the Federal Armed Forces
Department of Signal Processing and Communications
Hamburg, Germany

ABSTRACT

In audio codec design, often various parameters have to be fixed which may have a dramatic impact on codec performance. In this paper, we report on successful optimization of a codec based on perceptual criteria. Specifically, the PEAQ measure is used to determine the audio quality over a set of test items and search algorithms are used for optimization. First, simulated annealing is used for global search, then Rosenbrock's method is used to further refine the result. As shown by an example, the improvements gained by optimization compared to an educated guess are substantial.

Index Terms— Audio coding, optimization methods, simulated annealing

1. INTRODUCTION

In recent work, the authors have proposed delay-free audio coding schemes based on adaptive differential pulse code modulation (ADPCM) and various forms of adaptive spectral shaping of the coding noise [1, 2, 3]. In order to get the best possible audio quality from each codec and to allow a fair comparison of the approaches, it is necessary to choose the various parameters optimally. Unfortunately, this optimum cannot be derived analytically and an educated guess of the designer may be far from optimal.

Therefore, an automatic optimization based on two search methods has been employed which we believe could be fruitfully applied in the development of audio codecs in general. First, simulated annealing is used for global search, then Rosenbrock's method is used to further refine the result. Both methods can adjust to the local problem topology without requiring explicit computation or estimation of the gradient.

2. THE CODEC AND ITS PARAMETERS

While a detailed description of the coding schemes to be optimized is left to [1, 2, 3], we shall give a brief overview of the ADPCM codec at their core and introduce the parameters to optimize. The noise-shaping techniques are optimized similarly, so we will limit this paper on the ADPCM part for brevity's sake.

The ADPCM encoder starts by subtracting from its input signal a prediction based on past values. The resulting prediction error is then normalized to unit power and this normalized signal is finally requantized to the desired bit-rate, typically 3 bit or 4 bit per sample in our case.

The prediction is computed by an adaptive linear filter. Its filter order and the step size of the gradient based adaptation have to be specified. If the filter order is too low, the achievable prediction gain is insufficient, if the filter order is too high, the adaptation is slowed down and the predictor's tracking capabilities are reduced. Typically 2-digit values present a good compromise between steady-state performance and adaptation speed. The same trade-off is necessary for adaptation step-size; additionally, stability concerns pose a theoretical upper bound of 2 for the prediction scheme we use. Typical values lie well below 0.1, however.

For power normalization of the prediction error, the power is estimated by filtering squared past samples with a recursive first-order low-pass. The low-pass switches between two coefficients, depending on whether the power is increasing or decreasing, both of which lie between 0 and 1. Usually, the coefficient used for decreasing power is the lower one, resulting in a slower adaptation. A further parameter to choose is a lower bound for the estimated power to avoid excessive amplification which would result in quantizer overload at signal onsets. As this lower bound also limits the codec's dynamic range, a value of about 2×10^{-10} corresponding to -96 dB seems reasonable.

Finally, the quantizer levels used when requantizing to the desired bit-rate need to be determined. While the Lloyd-Max-algorithm in principle allows designing a minimum distortion quantizer [5, 6], it requires the input signal's probability distribution. Not only is the probability distribution unknown, but due to the backwards adaption of the codec, it depends on the codec parameters including the quantizer levels themselves. Nevertheless, the minimum distortion quantizer for a Gaussian distribution may serve as a suitable starting point. The only restriction enforced upon the quantizer is symmetry, primarily to reduce the number of parameters to optimize.

The parameters of the codec, together with typical values obtained in manual trials during codec development, are summarized in table 1. The total number of parameters is 9

parameter	range	typical values
prediction filter order	positive integer	30 to 100
predictor step-size	0 to 2	10^{-5} to 10^{-1}
power estimation filter coefficients	0 to 1	0.01 to 0.1
power lower bound	0 to 1	2×10^{-10}
positive quantizer levels	positive reals	distributed between 0 and 3 with higher density for lower values

Table 1. The parameters to optimize and typical values obtained from manual trials.

for quantization to 3 bit per sample and 13 for 4 bit per sample due to the higher number of quantizer levels to determine. This dimensionality of the parameter space is moderate so that optimization should be feasible despite the complicated cost function we will use.

3. OBJECTIVE PERCEPTUAL EVALUATION

Intuitively, the aim of the parameter optimization can be simply put: Achieve the best possible audio quality over a range of representative test items. As test items, the European Broadcasting Union (EBU) Sound Quality Assessment Material (SQAM) is appropriate as it was created just for the purpose of quality assessment and is often used when evaluating audio codecs. The choice of a suitable quality measure needs more attention.

Instead of performing listening tests for every iteration of an optimization procedure, an algorithmic evaluation is required that is able to approximate the results of a real listening test. Such an algorithm is defined in [4], commonly referred to as PEAQ, perceptual evaluation of audio quality. Interestingly, the aim of the standard is not only to allow objective comparison of codecs based on perceptual criteria, but also to support codec development — however, no prior attempts at automatic optimization based on this measure are reported.

The PEAQ algorithm requires the original reference signal and the test signal after coding and decoding and yields the objective difference grade (ODG), a value between -4 (very annoying differences) and 0 (no differences perceptible). Applying the PEAQ algorithm to all 70 SQAM items after coding and decoding using some parameter vector $\boldsymbol{\chi}$, 70 individual ODGs $K_n(\boldsymbol{\chi})$ are obtained. To be usable as a cost function for optimization, these must be reduced to a single value.

Simple averaging is insufficient, as the result of an optimization may then be a codec with excellent performance for most items, but very poor results for a few items, severely limiting its general applicability. Using the worst individual result as cost function is also inappropriate: With no changes to the worst item, possible improvements to others would be ignored. As a compromise, the fourth powers may be averaged, yielding the cost function

$$K(\boldsymbol{\chi}) = \sum_{n=1}^{70} K_n^4(\boldsymbol{\chi}) \tag{1}$$

which puts strong emphasis on worse signals without completely neglecting better ones.

The down-sides of using the ODG as a quality measure are high computational demands and a non-continuous cost function. Depending on the predictor order used in the codec, the process of encoding, decoding and calculating the ODG for all 70 items takes more then 20 minutes on current PC hardware. Therefore, an optimization technique is required which uses as few cost function evaluations as possible.

Unfortunately, due to the non-continuity of the cost function, no gradient is available for optimization. This prohibits the application of most commonly used non-linear optimization algorithms. The best alternative are search type algorithms that use the information gained from cost function evaluations in past iterations to decide on the best parameter set to be used for the next evaluation.

4. GLOBAL SEARCH WITH SIMULATED ANNEALING

The first search algorithm we apply is simulated annealing, originally proposed for discrete optimization [7, 8], which models an annealing melt, going from a hot chaotic state to a energetically near optimal crystalline structure. Starting from a parameter vector $\boldsymbol{\chi}_i$ in the i-th iteration, a new candidate vector $\tilde{\boldsymbol{\chi}}_i$ is obtained by applying a small random modification to $\boldsymbol{\chi}_i$. The candidate vector becomes the next $\boldsymbol{\chi}_{i+1}$ if $K(\tilde{\boldsymbol{\chi}}_i) \le K(\boldsymbol{\chi}_i)$ or if a random experiment succeeds, the probability of which decreases with increasing $K(\tilde{\boldsymbol{\chi}}_i) - K(\boldsymbol{\chi}_i)$; otherwise, $\boldsymbol{\chi}_{i+1} = \boldsymbol{\chi}_i$ is retained.

Specifically, if $K(\tilde{\boldsymbol{\chi}}_i) > K(\boldsymbol{\chi}_i)$, $K(\tilde{\boldsymbol{\chi}}_i)$ is accepted with probability

$$P_i = \exp\left(-\frac{K(\tilde{\boldsymbol{\chi}}_i) - K(\boldsymbol{\chi}_i)}{T_i}\right), \tag{2}$$

where the temperature T_i controls the overall acceptance probability and is reduced during the optimization. In the beginning, almost all variations are accepted resulting in a global search without getting trapped in a local minimum, while later, variations leading to an improvement are preferred, guiding the search to the minimum.

In practice, the random experiment to determine acceptance is drawing a random number r_i uniformly distributed over $[0,1]$ and accepting $\tilde{\boldsymbol{\chi}}_i$ if $r_i \le P_i$. This allows to first determine r_i and then solve equation (2) for the maximum

$$K_{\max,i} = K(\boldsymbol{\chi}_i) - T_i \log r_i \tag{3}$$

allowed for acceptance of $\tilde{\boldsymbol{\chi}}_i$. This may be used to reduce the computational demands when evaluating the cost function, as evaluating the sum in equation (1) may be terminated as soon as $K_{\max,i}$ is exceeded, so that for a rejected $\tilde{\boldsymbol{\chi}}_i$, typically only a subset of the $K_n(\tilde{\boldsymbol{\chi}}_i)$ have to be computed. By evaluating those $K_n(\tilde{\boldsymbol{\chi}}_i)$ first for which the highest values may be expected from the results of previous iterations, the benefit is maximized.

For discrete problems, the variation of $\boldsymbol{\chi}_i$ to obtain $\tilde{\boldsymbol{\chi}}_i$ usually is simply taking a discrete step of one or more randomly chosen variables. For continuous problems, more elaborate schemes are possible [9] to take into account the significant differences in typical value ranges of the various codec parameters and the complicated mutual dependencies. This is done by using an additive variation

$$\tilde{\boldsymbol{\chi}}_i = \boldsymbol{\chi}_i + \boldsymbol{\Delta}_i, \tag{4}$$

where the random vector $\boldsymbol{\Delta}_i$ is chosen to have zero mean and obey a covariance $\boldsymbol{S}_i$. The trick is to use the covariance of past $\boldsymbol{\chi}_i$, as these automatically carry information about the local topology of the cost function. We use an exponentially weighted covariance estimation with simple recursive computation via

$$\boldsymbol{\mu}_i = (1-\lambda)\boldsymbol{\chi}_i + \lambda\boldsymbol{\mu}_{i-1} \tag{5}$$

$$\boldsymbol{S}_i = \frac{1-\lambda^2}{2\lambda^2}(\boldsymbol{\chi}_i - \boldsymbol{\mu}_i)(\boldsymbol{\chi}_i - \boldsymbol{\mu}_i)^T + \lambda\boldsymbol{S}_{i-1}, \tag{6}$$

where good results where achieved with $\lambda = 0.95$.

The last design choice is the cooling scheme by which T_i is decreased. Cooling too quickly will result in premature convergence to a local minimum, while cooling too slowly will dramatically increase run-time. As a compromise, we choose

$$T_{i+1} = \begin{cases} 0.95 \cdot T_i & \text{if } \tilde{\boldsymbol{\chi}}_i \text{ is accepted} \\ T_i & \text{else.} \end{cases} \tag{7}$$

Although this scheme is far from guaranteeing convergence to the global optimum, in practice, the solutions we could obtain in acceptable time are very good.

5. LOCAL SEARCH WITH ROSENBROCK'S METHOD

While simulated annealing is well suited for globally exploring the parameter space with closer examination of regions with better performance, its final convergence to a local minimum is relatively slow. We therefore stop the global search when no significant changes have occurred for some time and turn to a local search algorithm.

The local search method we use has been proposed by Rosenbrock [10]. The algorithm is organized in rounds, where in each round, optimization is performed by searching along orthonormal basis vectors of the parameter space. After each round, these orthonormal vectors are adapted based on the complete step taken in the round.

parameter	initial value	final value
prediction filter order	64	59
predictor step-size	5×10^{-3}	$6.791\,10 \times 10^{-3}$
power estimation filter coefficients	0.1 and 0.03	0.093 422 3 and 0.049 881 2
power lower bound	2×10^{-10}	$9.383\,34 \times 10^{-11}$
positive quantizer levels	0.2451, 0.7560, 1.3440, 2.1520	0.195498, 0.763898, 1.48229, 2.98078

Table 2. Initial and final values of the optimization.

Specifically, let $\boldsymbol{u}_m$ denote the orthonormal basis vectors and k_m associated step sizes. Then, iterating over the $\boldsymbol{u}_m$, new candidate parameter vectors $\tilde{\boldsymbol{\chi}}_i = \boldsymbol{\chi}_i + k_m\boldsymbol{u}_m$ are constructed and accepted as $\boldsymbol{\chi}_{i+1} = \tilde{\boldsymbol{\chi}}_i$ if $K(\tilde{\boldsymbol{\chi}}_i) \le K(\boldsymbol{\chi}_i)$, otherwise $\boldsymbol{\chi}_{i+1} = \boldsymbol{\chi}_i$. If $\tilde{\boldsymbol{\chi}}_i$ is accepted, the respective step size k_m is increased by some factor $\alpha > 1$, otherwise, it is decreased and reversed with some other factor $-1 < \beta < 0$ (we use $\alpha = 5$ and $\beta = -0.5$). The round continues until for each $\boldsymbol{u}_m$, at least one $\tilde{\boldsymbol{\chi}}_i$ has been accepted and at least one has been rejected.

After each round, a new set of basis vectors is determined. First, from the total step $\boldsymbol{\Delta}$ taken in the whole round, a set of helper vectors

$$\boldsymbol{q}_m = \boldsymbol{\Delta} - \sum_{n=1}^{m-1} \boldsymbol{\Delta}^T \boldsymbol{u}_n \boldsymbol{u}_n \tag{8}$$

is computed such that $\boldsymbol{q}_1 = \boldsymbol{\Delta}$ is equal to the whole step, $\boldsymbol{q}_2$ does not contain the part in direction $\boldsymbol{u}_1$, $\boldsymbol{q}_3$ does not contain the parts in directions $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$, and so on. From these $\boldsymbol{q}_m$, the new $\boldsymbol{u}_m$ used in the next round are determined by Gram-Schmidt-orthogonalization and normalization.

6. RESULTS

To demonstrate the performance of the optimization, we show the behavior for optimizing the ADPCM system with quantization to 3 bit per sample. We start with the initial values given in table 2, which are a reasonable educated guess. The covariance matrix of the simulated annealing search is initialized to a diagonal matrix containing one hundredth of the square root of the respective parameter's initial values. Hence, in the beginning the different parameters are modified independently with an average step size proportional to their value. The temperature is initialized to $T_0 = 1680$.

The course of the cost function over the iterations of the simulated annealing search is depicted in figure 1. It is clearly visible how at first, almost arbitrary deteriorations may occur, while later on, only small increases of the cost functions are

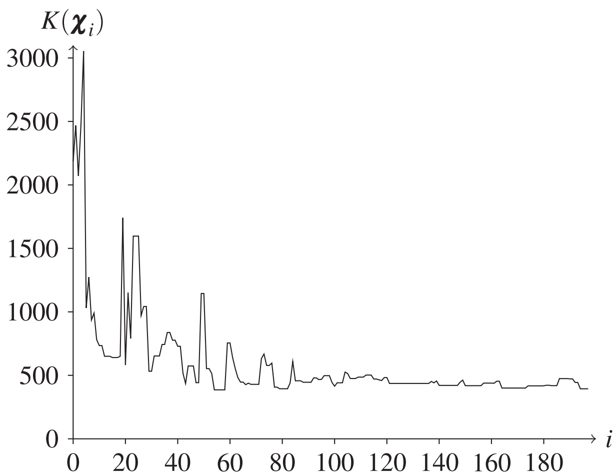

Fig. 1. Cost function versus iteration for simulated annealing.

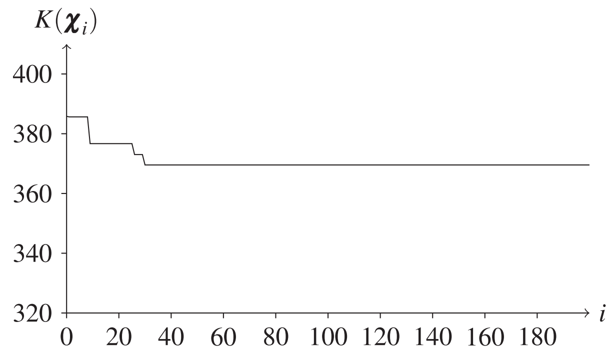

Fig. 2. Cost function versus iteration for Rosenbrock's method.

allowed, leading to an overall downwards trend. The simulated annealing search is terminated after 197 iterations. The cost function is reduced from $K(\boldsymbol{\chi}_0) = 2184$ to $K(\boldsymbol{\chi}_{197}) = 394$, a clear improvement.

For initializing the Rosenbrock search, the overall best solution during the simulated annealing which occurs already after the 54th iteration with $K(\boldsymbol{\chi}_{54}) = 386$ is used instead of the final one. The orthonormal basis vector are initialized to the unit vectors, the step sizes are chosen as one thousandth of the respective parameter values.

The resulting cost function course is depicted in figure 2. In only 30 iterations, the cost function is further reduced to 370. Here, a local minimum is reached, so that no further reductions occur.

The final parameter values found are also listed in table 2. The largest deviation from the initial values can be observed for the quantization levels, while the changes of the other parameters seem almost insignificant. Nevertheless, an impressive improvement in audio quality, as reflected by the cost function, is achieved.

7. CONCLUSION

We have presented an optimization approach for codec parameter selection based on perceptual criteria. As evaluating the cost function for perceptual evaluation is expensive, search methods that save on the number of required evaluations are employed, namely simulated annealing and Rosenbrock's method. While the former is well-suited for global search, the latter has better local convergence properties. The improvements gained compared to an educated guess of the codec parameters are substantial.

8. REFERENCES

[1] M. Holters, O.M. Pabst, and U. Zölzer, "ADPCM with adaptive pre- and post-filtering for delay-free audio coding," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Honolulu, Apr. 2007, vol. 1, pp. 269–272.

[2] M. Holters and U. Zölzer, "Delay-free lossy audio coding using shelving pre- and post-filters," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Las Vegas, Apr. 2008, pp. 209–212.

[3] M. Holters, C.R. Helmrich, and U. Zölzer, "Delay-free audio coding based on ADPCM and error feedback," in *Proc. 11th Int. Conf. Digital Audio Effects (DAFx)*, Espoo, Finland, Sept. 2008, pp. 221–225.

[4] ITU Recommendation ITU-R BS.1387-1, "Method for objective measurements of perceived audio quality," Nov. 2001.

[5] J. Max, "Quantizing for minimum distortion," *IRE Trans. Inf. Theory*, vol. 6, no. 1, pp. 7–12, Mar. 1960.

[6] S.P. Lloyd, "Least square quantization in PCM," *IEEE Trans. Inf. Theory*, vol. 28, no. 2, pp. 129–137, Mar. 1982.

[7] S. Kirkpatrick, C.D. Gelatt, Jr., and M.P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, May 1983.

[8] V. Černý, "Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm," *J. Opt. Theory Appl.*, vol. 45, no. 1, pp. 41–51, Jan. 1985.

[9] D. Vanderbilt and S.G. Louie, "A Monte Carlo simulated annealing approach to optimization over continuous variables," *J. Comp. Phys.*, vol. 36, no. 2, pp. 259–271, Nov. 1984.

[10] H.H. Rosenbrock, "An automatic method for finding the greatest or least value of a function," *Computer Journal*, vol. 3, no. 3, pp. 175–184, 1960.