Optimization and Assessment of Wavelet Packet Decompositions with Evolutionary Computation

Thomas Schell

Department of Scientific Computing, University of Salzburg, Jakob Haringer Street 2, A-5020 Salzburg, Austria
Email: tschell@cosy.sbg.ac.at

Andreas Uhl

Department of Scientific Computing, University of Salzburg, Jakob Haringer Street 2, A-5020 Salzburg, Austria
Email: uhl@cosy.sbg.ac.at

In image compression, the wavelet transformation is a state-of-the-art component. Recently, wavelet packet decomposition has received quite an interest. A popular approach for wavelet packet decomposition is the near-best-basis algorithm using nonadditive cost functions. In contrast to additive cost functions, the wavelet packet decomposition of the near-best-basis algorithm is only suboptimal. We apply methods from the field of evolutionary computation (EC) to test the quality of the near-best-basis results. We observe a phenomenon: the results of the near-best-basis algorithm are inferior in terms of cost-function optimization but are superior in terms of rate/distortion performance compared to EC methods.

Keywords and phrases: image compression, wavelet packets, best basis algorithm, genetic algorithms, random search.

1. INTRODUCTION

The DCT-based schemes for still-image compression (e.g., the JPEG standard [1]) have been superceded in favor of wavelet-based schemes in the last years. Consequently, the new JPEG2000 standard [2] is based on the wavelet transformation. Apart from the pyramidal decomposition, JPEG2000 part II also allows wavelet packet (WP) decomposition which is of particular interest to our studies.

WP-based image compression methods which have been developed [3, 4, 5, 6] outperform the most advanced wavelet coders (e.g., SPIHT [7]) significantly for textured images in terms of rate/distortion performance (r/d).

In the context of image compression, a more advanced but also more costly technique is to use a framework that includes both rate and distortion, where the best-basis (BB) subtree which minimizes the global distortion for a given coding budget is searched [8, 9]. Other methods use fixed bases of subbands for similar signals (e.g., fingerprints [10]) or search for good representations with general purpose optimization methods [11, 12].

Usually in wavelet-based image compression, only the coarse scale approximation subband is successively decomposed. With the WP decomposition also, the detail subbands lend themselves to further decomposition. From a practical

point of view, each decomposed subband results in four new subbands: approximation, horizontal detail, vertical detail, and diagonal detail. Each of these four subbands can be recursively decomposed at will. Consequently, the decomposition can be represented by a quadtree.

Concerning WPs, a key issue is the choice of the decomposition quadtree. Obviously, not every subband must be decomposed further; therefore, a criterion which determines whether a decomposition step should take place or not is needed.

Coifman and Wickerhauser [13] introduced additive cost functions and the BB algorithm which provides an optimal decomposition according to a specific cost metric. Taswell [14] introduced nonadditive cost functions which are thought to anticipate the properties of "good" decomposition quadtrees more accurately. With nonadditive cost functions, the BB algorithm mutates to a near-best-basis (NBB) algorithm because the decomposition trees are only suboptimal. The divide-and-conquer principle of the BB relies on the locality (additivity) of the underlying cost function. In the case of nonadditive cost functions, this locality does not exist.

In this work, we are interested in the assessment of the WP decompositions provided by the NBB algorithm. We focus on the quality of the NBB results in terms of

cost-function optimization as well as image quality (PSNR). Both, the cost-function value and the corresponding image quality of a WP decomposition is suboptimal due to the construction of the NBB algorithm.

We have interfaced the optimization process of WP decompositions by means of cost functions with the concepts of evolutionary computation (EC). Hereby, we obtain an alternative method to optimize WP decompositions by means of cost functions. Both approaches, NBB and EC, are subject to our experiments. The results provide valuable new insights concerning the intrinsic processes of the NBB algorithm. Our EC approach perfectly suits the needs for the assessment of the NBB algorithm but, from a practical point of view, the EC approach is not competitive in terms of computational complexity.

In Section 2, we review the definition of the cost functions which we analyze in our experiments. The NBB algorithm is described in Section 3. For the EC methods, we need a "flat" representation of quadtrees (Section 4). In Sections 5 and 6, we review genetic algorithms and random search specifically adapted to WP optimization. For our experiments, we apply an SPIHT inspired software package for image compression by means of WP decomposition. Our central tool of analysis are scatter plots of WP decompositions (Section 7). In Section 8, we compare the NBB algorithm and EC for optimizing WP decompositions.

2. COST FUNCTIONS

As a preliminary, we review the definitions of a cost function and the additivity. A cost function is a function $C : \mathbb{R}^M \times \mathbb{R}^N \to \mathbb{R}$. If $\mathbf{y} \in \mathbb{R}^M \times \mathbb{R}^N$ is a matrix of wavelet coefficients and C is a cost function, then $C(0) = 0$ and $C(\mathbf{y}) = \sum_{i,j} C(y_{ij})$. A cost function C is additive if and only if

$$C_a(\mathbf{z}_1 \oplus \mathbf{z}_2) = C_a(\mathbf{z}_1) + C_a(\mathbf{z}_2), \tag{1}$$

where $\mathbf{z}_1, \mathbf{z}_2 \in \mathbb{R}^M \times \mathbb{R}^N$ are matrices of wavelet coefficients. The goal of any optimization algorithm is to identify a WP decomposition with a minimal cost-function value.

Alternatively to the NBB algorithm (Section 3), we apply methods from evolutionary computation (Sections 5 and 6) to optimize WP decompositions. The fitness of a particular WP decomposition is estimated with nonadditive cost functions. We employ the three nonadditive cost functions listed below.

(i) Coifman Wickerhauser entropy. Coifman and Wickerhauser [15] defined the entropy for wavelet coefficients as follows:

$$C_n^1(\mathbf{y}) = \sum_{i,j: p_{ij} \neq 0} p_{ij} \ln p_{ij}, \quad p_{ij} = \frac{|y_{ij}|^2}{\|\mathbf{y}\|^2}. \tag{2}$$

(ii) Weak l^p Norm. For the weak l^p norm [16], we need to reorder and transform the coefficients y_{ij}. All coefficients y_{ij} are rearranged in a decreasing absolute-value sorted vector $\mathbf{z}$,

that is, $z_1 = |y_{i_1 j_1}| \geq \cdots \geq z_{MN} = |y_{i_M j_N}|$. Hence, the size of vector $\mathbf{z}$ is MN. The cost-function value is calculated as follows:

$$C_n^{4,p}(\mathbf{y}) = \max_k k^{1/p} z_k. \tag{3}$$

From the definition of the weak l^p norm, we deduce that unfavorable slowly decreasing sequences or, in the worst case, uniform sequences of vectors $\mathbf{z}$ cause high numerical values of the norm, whereas fast decreasing $\mathbf{z}$'s result in low ones.

(iii) Shannon entropy. Below, we will consider the matrix $\mathbf{y}$ simply as a collection of real-valued coefficients x_i, $1 \leq i \leq MN$. The matrix $\mathbf{y}$ is rearranged such that the first row is concatenated with the second row at the right side and then the new row is concatenated with the third row and so on. With a simple histogram binning method, we will estimate the probability mass function. The sample data interval is given by $a = \min_i x_i$ and $b = \max_i x_i$. Given the number of bins J, the bin width w is $w = (b - a)/J$. The frequency f_j for the jth bin is defined by $f_j = \#\{x_i \mid x_i \leq a + jw\} - \sum_{k=1}^{j-1} f_k$. The probabilities p_j are calculated from the frequencies f_j simply by $p_j = f_j/MN$. From the obtained class probabilities, we can calculate the Shannon entropy [14]

$$C_n^{2,J}(\mathbf{y}) = - \sum_{j=1}^{J} p_j \log_2 p_j. \tag{4}$$

Cost functions are an indirect strategy to optimize the image quality. PSNR can be seen as a nonadditive cost function. With a slightly modified NBB, PSNR as a cost function provides WP decomposition with an excellent r/d performance, but at the expense of high computational costs [12].

3. NBB ALGORITHM

With additive cost functions, a dynamic programming approach, that is, the BB algorithm [13], provides the optimal WP decomposition with respect to the applied cost function. Basically, the BB algorithm traverses the quadtree in a depth-first-search manner and starts at the level right above the leaves of the decomposition quadtree. The sum of the cost of the children node is compared to the cost of the parent node. If the sum is less than the cost of the parent node, the situation remains unchanged. But, if the cost of the parent node is less than the cost of the children, then the child nodes are pruned off the tree. From bottom upwards, the tree is reduced whenever the cost of a certain branch can be reduced. An illustrating example is presented in [15]. It is an essential property of the BB algorithm that the decomposition tree is optimal in terms of the cost criteria, but not in terms of the obtained r/d performance.

When switching from additive to nonadditive cost functions, the locality of the cost function evaluation is lost. The BB algorithm can still be applied because the correlation among the subbands is assumed to be minor but obviously the result is only suboptimal. Hence, instead of BB, this new variant is called NBB [14].

4. ENCODING OF WP QUADTREES

To interface the WP software and the EC methods, we use a flat representation of a WP-decomposition quadtree. In other words, we want an encoding scheme for quadtrees in the form of a (binary) string. Therefore, we have adopted the idea of coding a heap in the heap-sort algorithm. We use strings **b** of finite length L over a binary alphabet $\{0, 1\}$. If the bit at index k, $1 \leq k \leq L$, is set, then the according subband has to be decomposed. Otherwise, the decomposition stops in this branch of the tree

$$b_k = \begin{cases} 1 & \text{decompose,} \\ 0 & \text{stop.} \end{cases} \quad (5)$$

If the bit at index k is set ($b_k = 1$), the indices of the resulting four subbands are derived by

$$k'_m = 4 \cdot k + m, \quad 1 \leq m \leq 4. \quad (6)$$

In heaps, the levels of the tree are implicit. We denote the maximal level of the quadtree by $l_{\max} \in \mathbb{N}$. At this level, all nodes are leaves of the quadtree. The level l of any node k in the quadtree can be determined by

$$l = \begin{cases} 0, & k = 0 \text{ (root)}, \\ l : \sum_{r=0}^{l-1} 4^r \leq k < \sum_{r=0}^{l} 4^r, & k > 0. \end{cases} \quad (7)$$

The range of level l is $0 \leq l \leq l_{\max}$.

5. GENETIC ALGORITHM

Genetic algorithms (GAs) are evolution-based search algorithms especially designed for parameter optimization problems with vast search spaces. GAs were first proposed in the seventies by Holland [17]. Generally, parameter optimization problems consist of an objective function to evaluate and estimate the quality of an admissible parameter set, that is, a solution of the problem (not necessarily the optimal, just any one). For the GA, the parameter set needs to be encoded into a string over a finite alphabet (usually a binary alphabet). The encoded parameter set is called a *genotype*. Usually, the objective function is slightly modified to meet the requirements of the GA and hence will be called *fitness function*. The fitness function determines the quality (fitness) for each genotype (encoded solution). The combination of a genotype and the corresponding fitness forms an *individual*. At the start of an evolution process, an initial *population*, which consists of a fixed number of individuals, is generated randomly. In a selection process, individuals of high fitness are selected for *recombination*. The selection scheme mimics nature's principle of the *survival of the fittest*. During recombination, two individuals at the time exchange genetic material, that is, parts of the genotype string, are exchanged at random. After a new intermediate population has been created, a mutation operator is applied. The mutation operator randomly changes some of the *alleles* (values at certain positions/loci of

the genotype) with a small probability in order to ensure that alleles which might have vanished from the population have a chance to reenter. After applying mutation, the intermediate population has turned into a new one (next generation) replacing the former.

For our experiments, we apply a GA which starts with an initial population of 100 individuals. The initial population is generated randomly. The chromosomes are decoded into WP decompositions as described in Section 4. The fitness of the individuals is determined with a cost function (Section 2). Then, the standard cycle of selection, crossover, and mutation is repeated 100 times, that is, we evolve 100 generations of the initial population. The maximum number of generations was selected empirically such that selection schemes with a low selection pressure sufficiently converge. As selection methods, we use binary tournament selection (TS) with partial replacement [18] and linear ranking selection (LKR) with $\eta = 0.9$ [19]. We have experimented with two variants of crossover. Firstly, we applied standard two-point crossover but obviously this type of crossover does not take into account the tree structure of the chromosomes. Additionally, we have conducted experiments with a tree-crossover operator (Section 5.1) which is specifically adapted to operations on quadtrees. For both, two-point crossover and tree crossover, the crossover rate is set to 0.6 and the mutation rate is set to 0.01 for all experiments.

As a by-product, we obtained the results presented in Figure 1 for the image Barbara (Figure 5). Instead of a cost function, we apply the image quality (PSNR) to determine the fitness of an individual (i.e., WP decomposition). We present the development of the PSNR during the course of a GA. We show the GA results in the following parameter combinations: LRK and TS, each with either two-point crossover or with tree crossover. After every 100th sample (population size of the GA) of the random search (RS, Section 6), we indicate the best-so-far WP decomposition. Obviously, for each evaluation of a WP decomposition, a full compression and decompression step which causes a tremendous execution time is required. The result of a NBB optimization using weak l^1 norm is displayed as a horizontal line because the runtime of the NBB algorithm is far below the time which is required to evolve one generation of the GA. The PSNR of the NBB algorithm is out of reach for RS and GA. The tree-crossover operator does not improve the performance of the standard GA. The execution of a GA or RS run lasts from 6 to 10 days on an AMD Duron processor with 600 MHz. The GA using TS with and without tree crossover was not able to complete the 100 generations within this time limit. Further examples of WP optimization by means of EC are discussed in [20].

5.1. Tree crossover

Standard crossover operators (e.g., one-point or two-point crossover) have a considerably disruptive effect on the tree structure of subbands which is encoded into a binary string. With the encoding discussed above, a one- or two-point crossover results in two new individuals with tree structures which are almost unrelated to the tree structures of their
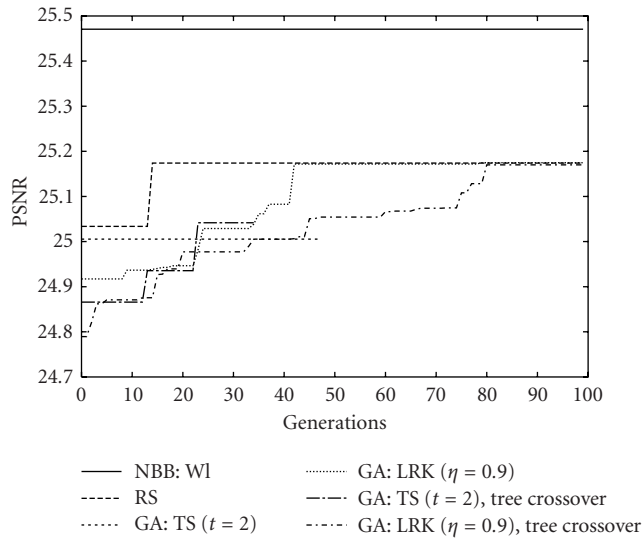
FIGURE 1: Comparison of NBB, GA, and RS.

TABLE 1: Chromosomes of two individuals.

	1	2	3	4	5	6	7	8	9	10	11	12	13	$\cdots$
A	1	0	1	0	0	0	1	0	0	0	0	0	1	1	$\cdots$
B	1	1	0	1	1	0	0	1	1	1	1	0	0	$\cdots$

FIGURE 2: Parent individuals before crossover.

FIGURE 3: Individuals after conventional one-point crossover.

parents. This obviously contradicts the basic idea of a GA, that is, the GA is expected to evolve better individuals from good parents.

To demonstrate the effect of standard one-point crossover, we present a simple example. The chromosomes of the parent individuals A and B are listed in Table 1 and the according binary trees are shown in Figure 2. As a cut point for the crossover, we choose the gap between gene 6 and 7. The chromosome parts from locus 7 to the right end of the chromosome are exchanged between individuals A and B. This results in two new trees (i.e., individual A′ and B′) which are displayed in Figure 3. Evidently, the new generation of trees differ considerably from their parents.
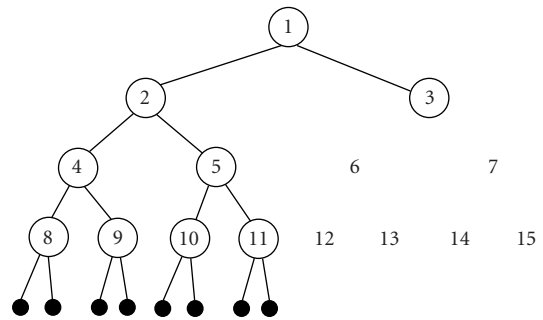
The notion is to introduce a problem-inspired crossover such that the overall tree structure is preserved while only local parts of the subband trees are altered [11]. Specifically, one node in each individual (i.e., subband tree) is chosen at random, then the according subtrees are exchanged between the individuals. In our example, the candidate nodes for the crossover are node 2 in individual A and node 10 in individual B. The tree crossover produces a new pair of descendants A′′ and B′′ which are displayed in Figure 4. Compared to the standard crossover operator, tree crossover moderately alters the structure of the parent individuals and generates new ones.

6. RANDOM SEARCH

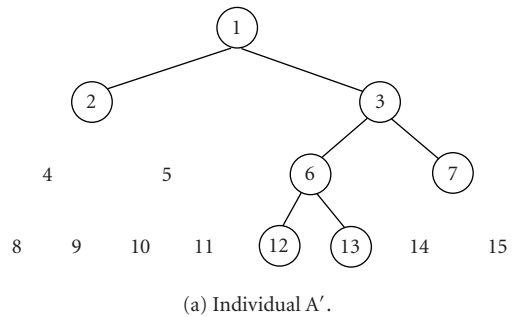The random generation of WP decompositions is not straightforward due to the quadtree structure. If we consider a 0/1 string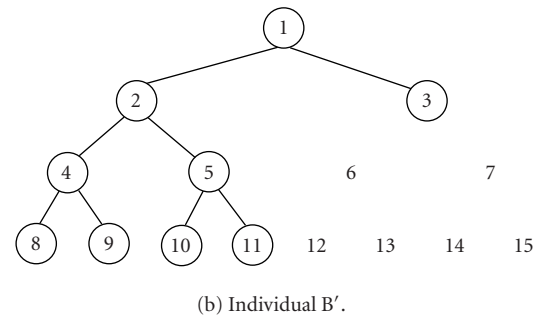 as an encoded quadtree (Section 4), we could obtain random WP decomposition just by creating random 0/1 strings of a given length. An obvious drawback is that this method acts in favor of small quadtrees. We assume that
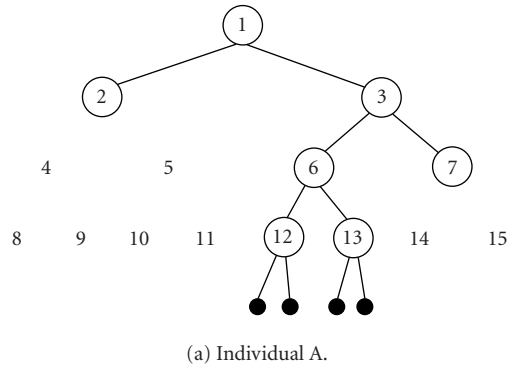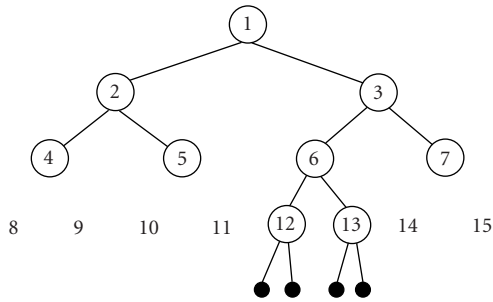
(a) Individual A″.
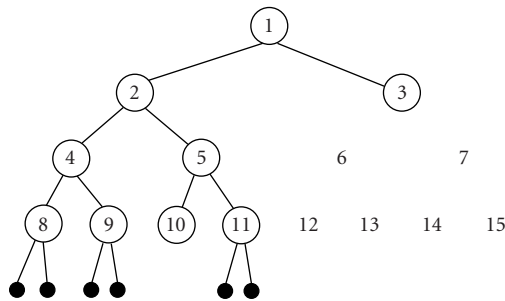
(b) Individual B″.

FIGURE 4: Individuals after tree crossover.

FIGURE 5: Barbara.

the root node always exists and that it is on level $l = 0$. This is a useful assumption because we need at least one wavelet decomposition. The probability to obtain a node at level l is $(1/2)^l$. Due to the rapidly decreasing probabilities, the quadtrees will be rather sparse.

Another admittedly theoretical approach would be to assign a uniform probability to all possible quadtrees. Then, this set is sampled for WP decompositions. Some simple considerations will show that in this case small quadtrees are excluded from evaluation. In the following, we will calculate the number $A(k)$ of trees with nodes on equal or less than k levels. If $k = 0$, then we have $A(0) := 1$ because there is only the root node on level $l = 0$. For $A(k)$, we obtain the recursion $A(k) = [1 + A(k - 1)]^4$ because we can construct quadtrees of height equal to or less than k by adding a new root node to trees of height $k - 1$. The number of quadtrees $B(k)$ of height k is given by $B(0) := 1$ and $B(k) = A(k) - A(k - 1), k \geq 1$. From the latter argument, we see that the number of quadtrees of height $B(k)$ increases exponentially. Consequently, the number of trees of low height is diminishing and hence, when uniformly sampling the set of quadtrees, they are almost excluded from the evaluation.

With image compression in mind, we are interested in trees of low height because trees with a low number of nodes and a simple structure require less resources when encoded into a bitstream. Therefore, we have adopted the RS approach of the first paragraph with a minor modification. We require that the approximation subband is at least decomposed down to level 4 because it contains usually a considerable amount of the overall signal energy.

Similar to the GA, we can apply the RS using PSNR instead of cost functions to evaluate WP decompositions. Using a RS as discussed above with a decomposition depth of at least 4 for the approximation subband, we generate 4000 almost unique samples of WP decompositions and evaluate the corresponding PSNR. The WP decomposition with the highest PSNR value is recorded. We have repeated the single RS runs at least 90 times. The best three results in decreasing order and the least result of a single RS run for the image Barbara are presented as follows: 24.648, 24.6418, 24.6368, ..., 24.4094.

If we compare the results of the RS to those obtained by NBB with cost function weak l^1 norm (PSNR 25.47), we realize that the RS is about 1 dB below the NBB algorithm. To increase the probability of a high quality result of the RS, a drastic increase of the sample size is required, which again would result in a tremendous increase of the RS runtime.

7. CORRELATION OF COST FUNCTIONS AND IMAGE QUALITY

Our experiments are based on a test library of images with a broad spectrum of visual features. In this work, we present the results for the well-known image Barbara. The considerable amount of texture in the test picture demonstrates the superior performance of the WP approach in principle.

The output of the NBB, GA, and RS is a WP decomposition. WPs are a generalization of the pyramidal decomposition. Therefore, we apply an algorithm similar to SPIHT which exploits the hierarchical structure of the wavelet coefficients [21] (SMAWZ). SMAWZ uses the foundations of SPIHT, most importantly the zero-tree paradigm, and adapts them to WPs.

Cost functions are the central design element in the NBB algorithm. The working hypothesis of (additive and nonadditive) cost functions is that a WP decomposition with an optimal cost-function value provides also a (sub-) optimal r/d performance. The optimization of WP decompositions via cost functions is an indirect strategy. Therefore, we compare the results of the EC methods to that of the NBB algorithm by generating scatter plots. In these plots, we simultaneously
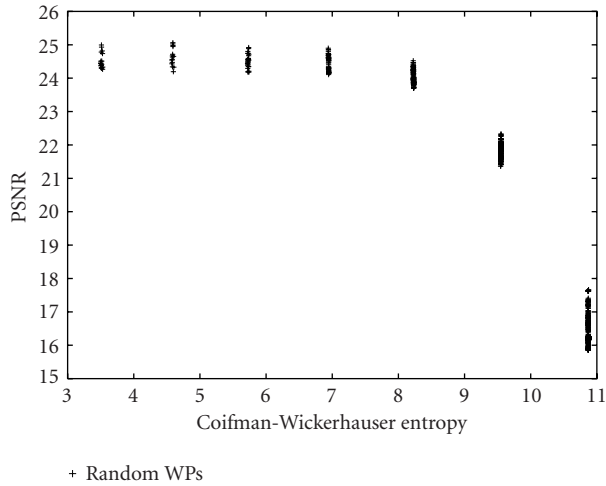
Figure 6: Correlation between Coifman-Wickerhauser entropy and PSNR.

Figure 7: Correlation between Coifman-Wickerhauser entropy and PSNR for WP decompositions obtained by NBB, RS, and GA.

provide for each WP decomposition the information about the cost-function value and the image quality (PSNR).

Figure 6 displays the correlation of the nonadditive cost-function Coifman-Wickerhauser entropy and the PSNR. For the plot, we generated 1000 random WP decompositions and calculated the value of the cost function and the PSNR after a compression with 0.1 bpp. Note that WP decompositions with the same decomposition level of the approximation subband are grouped into clouds.

8. QUALITY OF THE NBB ALGORITHM WITH RESPECT TO COST-FUNCTION OPTIMIZATION

The basic idea of our assessment of the NBB algorithm is to use the GA to evolve WP decompositions by means of cost-function optimization. Therefore, we choose some nonadditive cost functions and compute WP decompositions with the NBB algorithm, a GA, and a RS. For each cost function, we obtain a collection of suboptimal WP decompositions. We calculate the PSNR for each of the WP decompositions and generate scatter plots (PSNR versus cost-function value). The comparison of the NBB, GA, and RS results provide surprising insight into the intrinsic processes of the NBB algorithm.

We apply the GA and RS as discussed in Sections 5 and 6, using the nonadditive cost-functions Coifman-Wickerhauser entropy, weak l^1 norm, and Shannon entropy to optimize WP decompositions. The GA as well as the RS generate and evaluate 10^4 WP decompositions. The image Barbara is decomposed according to the output of NBB, GA, and RS and compressed to 0.1 bpp. Afterwards, we determine the PSNR of the original and the decompressed image.

In Figure 7, we present the plot for the correlation between the Coifman-Wickerhauser entropy and PSNR for NBB, GA, and RS. The WP decomposition obtained by the NBB algorithm is displayed as a single dot. The other dots
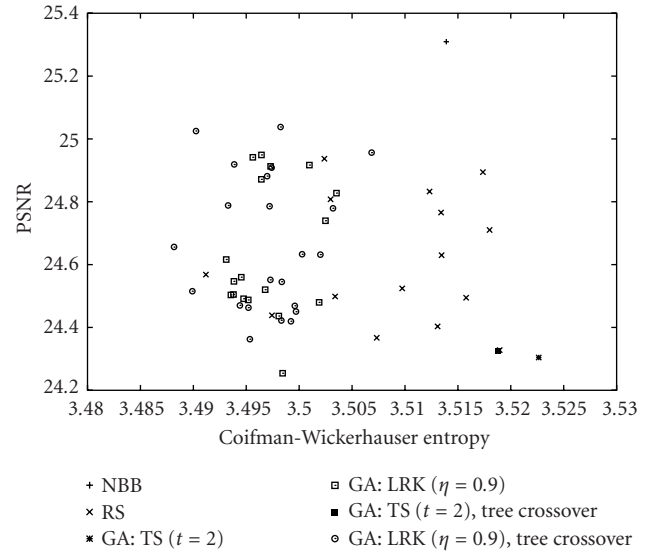
represent the best individual found either by a RS or a GA run. With the Coifman-Wickerhauser entropy, we notice a defect in the construction of the cost function. Even though the GA and RS provide WP decompositions with a cost-function value less than that of the NBB, the WP decomposition of the NBB is superior in terms of image quality. As a matter of fact, the NBB provides suboptimal WP decompositions with respect to the Coifman-Wickerhauser entropy.

The correlation between weak l^1 norm and PSNR is displayed in Figure 8. Similar to the scatter-plot of the Coifman-Wickerhauser entropy, the WP decomposition of the NBB is an isolated dot. But this time, the GA and the RS are not able to provide a WP decomposition with a cost-function value less than the cost-function value of the NBB-WP decomposition.

Even more interesting is the cost-function Shannon entropy (Figure 9). Similar to the Coifman-Wickerhauser entropy, the Shannon entropy provides WP decompositions with a cost-function value lower than the NBB. In the upper right of the figure, there is a singular result of the GA using TS. This WP decomposition has an even higher cost-function value than the one of the NBB but is superior in terms of PSNR.

In general, the GA employing LRK provides better results than the GA using TS concerning the cost-function values. Within the GA-LRK results, there seems to be a slight advantage for the tree crossover. In all three figures, the GA-LRK with and without tree crossover is clearly ahead of the RS. This is evidence for a more efficient optimization process of the GA compared to RS.

In two cases (Figures 7 and 9), we observe the best cost-function values for the GA- and the RS-WP decomposition. Nevertheless, the NBB-WP decomposition provides higher image quality with an inferior cost-function value. The singular result for the GA of Figure 9 is yet another example
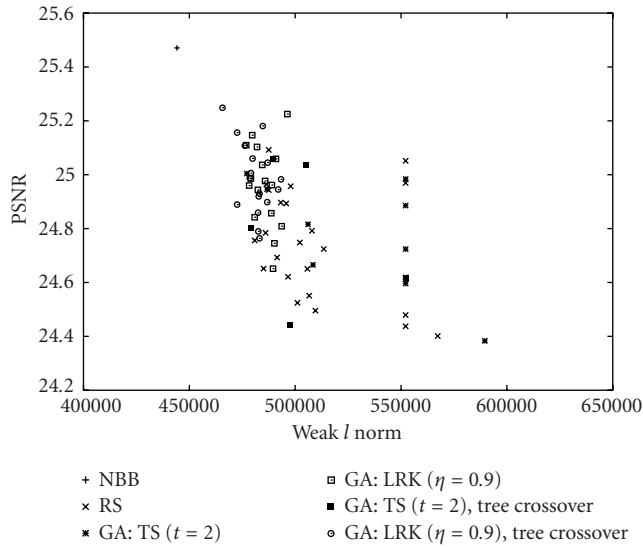
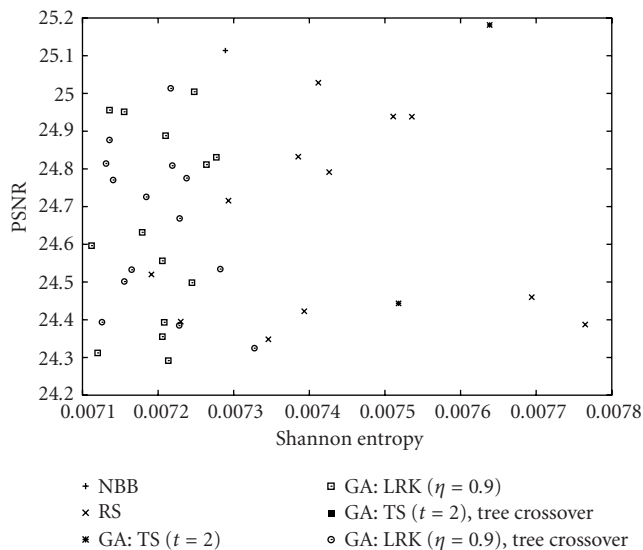FIGURE 8: Correlation between weak l^1 norm and PSNR for WP decompositions obtained by NBB, RS, and GA.

FIGURE 9: Correlation between Shannon entropy and PSNR for WP decompositions obtained by NBB, RS, and GA. The results of GA: TS ($t = 2$), tree crossover are not displayed due to zooming.

for this phenomenon. As a result, the correlation of the cost-function value and the PSNR, as indicated in all three scatter plots, is imperfect. (In the case of perfect correlation, we would observe a line starting in the right and descending to the left.)

The NBB algorithm generates WP decompositions according to split and combine decisions based on cost-function evaluations. In contrast, RS and GA generate a complete WP decomposition and the cost-function value is computed afterwards. The overall cost-function values of NBB,

RS, and GA fail to consistently predict the image quality, that is, a lower cost-function value does not assert a higher image quality.

9. SUMMARY

The NBB algorithm for WP decomposition provides, due to the construction only, suboptimal cost-function values as well as suboptimal image quality. We are interested in an assessment of the quality of the NBB results.

We have adapted a GA and a RS to the problem of WP-decomposition optimization by means of additive and nonadditive cost functions. For the GA, a problem-inspired crossover operator was implemented to reduce the disruptive effect on decomposition trees when recombining the chromosomes of WP decompositions. Obviously, the computational complexity of RS and GA are exorbitantly higher than that of the NBB algorithm. But the RS and GA are in this case helper applications for the assessment of the NBB algorithm.

We compute WP decompositions with the NBB algorithm, the RS, and GA. The central tool of analysis is the correlation between cost-function value and the corresponding PSNR of WP decompositions which we visualize with scatter plots. The scatter plots reveal the imperfect correlation between cost-function value and image quality for WP decompositions for all of the presented nonadditive cost functions. This also holds true for many other additive and nonadditive cost functions. We observed that the NBB-WP decomposition provided excellent image quality even though the corresponding cost-function value was sometimes considerably inferior compared to the results of the RS and GA. Consequently, our results revealed defects in the prediction of image quality by means of cost functions.

With the RS and GA at hand, we applied minor modifications to these algorithms. Instead of employing cost functions for optimizing WP decompositions, we used the PSNR as a fitness function which resulted in a further increase of computational complexity because each evaluation of a WP decomposition requires a full compression and decompression step. Hereby, we directly optimize the image quality. This direct approach of optimizing WP decomposition with GA and RS, employing PSNR as a fitness function, requires further improvement to exceed the performance of the NBB.

REFERENCES

[1] W. B. Pennebaker and J. L. Mitchell, *JPEG: Still Image Data Compression Standard*, Van Nostrand Reinhold, New York, NY, USA, 1993.
[2] D. Taubman and M. W. Marcellin, *JPEG2000: Image Compression Fundamentals, Standards and Practice*, Kluwer Academic Publishers, Boston, Mass, USA, 2002.
[3] J. R. Goldschneider and E. A. Riskin, "Optimal bit allocation and best-basis selection for wavelet packets and TSVQ," *IEEE Trans. Image Processing*, vol. 8, no. 9, pp. 1305–1309, 1999.
[4] F. G. Meyer, A. Z. Averbuch, and J.-O. Strömberg, "Fast adaptive wavelet packet image compression," *IEEE Trans. Image Processing*, vol. 9, no. 5, pp. 792–800, 2000.

[5] R. Öktem, L. Öktem, and K. Egiazarian, "Wavelet based image compression by adaptive scanning of transform coefficients," *Journal of Electronic Imaging*, vol. 2, no. 11, pp. 257–261, 2002.

[6] Z. Xiong, K. Ramchandran, and M. T. Orchard, "Wavelet packet image coding using space-frequency quantization," *IEEE Trans. Image Processing*, vol. 7, no. 6, pp. 892–898, 1998.

[7] A. Said and W. A. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 6, no. 3, pp. 243–250, 1996.

[8] K. Ramchandran and M. Vetterli, "Best wavelet packet bases in a rate-distortion sense," *IEEE Trans. Image Processing*, vol. 2, no. 2, pp. 160–175, 1993.

[9] N. M. Rajpoot, R. G. Wilson, F. G. Meyer, and R. R. Coifman, "A new basis selection paradigm for wavelet packet image coding," in *Proc. International Conference on Image Processing (ICIP '01)*, pp. 816–819, Thessaloniki, Greece, October 2001.

[10] T. Hopper, "Compression of gray-scale fingerprint images," in *Wavelet Applications*, H. H. Szu, Ed., vol. 2242 of *SPIE Proceedings*, pp. 180–187, Orlando, Fla, USA, 1994.

[11] T. Schell and A. Uhl, "Customized evolutionary optimization of subband structures for wavelet packet image compression," in *Advances in Fuzzy Systems and Evolutionary Computation*, N. Mastorakis, Ed., pp. 293–298, World Scientific Engineering Society, Puerto de la Cruz, Spain, February 2001.

[12] T. Schell and A. Uhl, "New models for generating optimal wavelet-packet-tree-structures," in *Proc. 3rd IEEE Benelux Signal Processing Symposium (SPS '02)*, pp. 225–228, IEEE Benelux Signal Processing Chapter, Leuven, Belgium, March 2002.

[13] R. R. Coifman and M. V. Wickerhauser, "Entropy based algorithms for best basis selection," *IEEE Transactions on Information Theory*, vol. 38, no. 2, pp. 713–718, 1992.

[14] C. Taswell, "Satisficing search algorithms for selecting near-best bases in adaptive tree-structured wavelet transforms," *IEEE Transactions on Signal Processing*, vol. 44, no. 10, pp. 2423–2438, 1996.

[15] M. V. Wickerhauser, *Adapted Wavelet Analysis from Theory to Software*, A. K. Peters, Wellesley, Mass, USA, 1994.

[16] C. Taswell, "Near-best basis selection algorithms with non-additive information cost functions," in *Proc. IEEE International Symposium on Time-Frequency and Time-Scale Analysis (TFTS '94)*, M. Amin, Ed., pp. 13–16, IEEE Press, Philadelphia, Pa, USA, October 1994.

[17] J. H. Holland, *Adaptation in Natural and Artificial Systems*, MIT Press, Ann Arbor, Mich, USA, 1975.

[18] T. Schell and S. Wegenkittl, "Looking beyond selection probabilities: adaption of the χ^2 measure for the performance analysis of selection methods in GA," *Evolutionary Computation*, vol. 9, no. 2, pp. 243–256, 2001.

[19] J. E. Baker, "Adaptive selection methods for genetic algorithms," in *Proc. 1st International Conference on Genetic Algorithms and Their Applications*, J. J. Grefenstette, Ed., pp. 101–111, Lawrence Erlbaum Associates, Hillsdale, NJ, USA, July 1985.

[20] T. Schell, *Evolutionary optimization: selection schemes, sampling and applications in image processing and pseudo random number generation*, Ph.D. thesis, University of Salzburg, Salzburg, Austria, 2001.

[21] R. Kutil, "A significance map based adaptive wavelet zerotree codec (SMAWZ)," in *Media Processors 2002*, S. Panchanathan, V. Bove, and S. I. Sudharsanan, Eds., vol. 4674 of *SPIE Proceedings*, pp. 61–71, San Jose, Calif, USA, January 2002.

Thomas Schell received his M.S. degree in computer science from Salzburg University, Austria and from the Bowling Green State University, USA and a Ph.D. from Salzburg University. Currently, he is with the Department of Scientific Computing as a Research and Teaching Assistant at Salzburg University. His research focuses on evolutionary computing and signal processing, especially image compression.

Andreas Uhl received the B.S. and M.S. degrees (both in mathematics) from Salzburg University and he completed his Ph.D. on applied mathematics at the same university. He is currently an Associate Professor with tenure in computer science affiliated with the Department of Scientific Computing, and with the Research Institute for Software Technology, Salzburg University. He is also a part-time lecturer at the Carinthia Tech Institute. His research interests include multimedia signal processing (with emphasis on compression and security issues), parallel and distributed processing, and number theoretical methods in numerics.