Non-redundant VQ Channel Coding Using Modified Tabu Search Approach with Simulated Annealing

J. S. Pan♱, Z. M. Lu†, S. C. Chu*♱ and S. H. Sun†
♱National Kaohsiung Institute of Technology, Taiwan, R.O.C.
†Harbin Institute of Technology, Harbin, P. R.C.
*University of South Australia, Australia

Keywords: Codeword Index Assignment, Modified Tabu Search, Simulated Annealing, VQ

Abstract

Codeword Index Assignment (*CIA*) is a key issue to Vector Quantization (*VQ*). A new algorithm called Modified Tabu Search Algorithm (*MTSA*) is applied to codeword index assignment for noisy channels for the purpose of minimizing the distortion due to bit errors. Simulated Annealing (*SA*) technique and a new parameter are introduced in the Tabu Search Approach (*TSA*) to improve the performance of the tabu search approach. Experimental tests show the modified tabu search algorithm is superior to the tabu search algorithm by evaluating the performance of channel distortion after the same number of iterations.

1. Introduction

Vector Quantization (*VQ*)[1] has been a popular and efficient technique for image compression and speech coding. A vector consisting of k samples of information source in the k-dimensional Euclidean space is sent to the vector quantizer. The k-dimensional, N-level vector quantizer is defined as a mapping from a k-dimensional Euclidean space R^k into a certain finite set $C=\{c_1,c_2,\ldots,c_N\}$. The output of the vector quantizer is the index i of the codeword c_i which satisfies

$$i=\arg\min_p \sum_{i=1}^{k}(x^l-c_p^l)^2 \qquad (1)$$

Only the index i is sent over the channel to the receiver. However, vector quantization is highly sensitive to channel noise [2,3]. The channel errors can cause errors in the received indices, that is, the index i is changed to index j. Thus, distortions are introduced in the decoding phase. Assume the channel model is a binary symmetric channel with bit error probability ε , i.e.,

$$P(b(c_j)/b(c_i)) = (1-\varepsilon)^{m-H(b(c_i),b(c_j))}\varepsilon^{H(b(c_i),b(c_j))} \qquad (2)$$

Where $b(c_i), i=1,2,\ldots,N$, is the index with m bit string of codeword c_i; $P(b(c_j)/b(c_i))$ denote the probability that index $b(c_j)$ is received given the index $b(c_i)$ is sent; $H(b(c_j),b(c_i))$ denote the Hamming distance between $b(c_i)$ and $b(c_j)$, i.e., the number of bits in which $b(c_j)$ and $b(c_i)$ differ. Assume the distortion between c_j and c_i is given by a non-negative distortion measure $d(c_j,c_i)$. Thus, for a given assignment of codeword indices $b=(b(c_1),b(c_2),\ldots,b(c_N))$, the distortion caused by the channel noise can be expressed as

0-7803-5578-4/99/$10.00©1999 IEEE

$$D=\sum_{i=1}^{N}\{P(c_i)\cdot \left[\sum_{l=1}^{m}\varepsilon^{l}\cdot(1-\varepsilon)^{m-l}\cdot \sum_{j:H(b(c_i),b(c_j))=l} d(c_i,c_j)\right]\} \quad (3)$$

Where $P(c_i)$ denotes the probability density function of c_i. Assigning suitable indices to codewords can reduce distortion due to an imperfect channel. However, to test $N!$ assignments is a NP-hard problem. The binary switching algorithm (BSA) [4] is proposed to improve the codevector index assignment by Zeger and Gersho. The Simulated annealing technique is applied to design the codeword indices by Farvardin [2]. The parallel genetic algorithm was applied to improve the codeword index assignment by Pan et. al. [3]. The tabu search approaches were developed for codeword index assignment by Pan and Chu [5] and results show that the channel distortion can be improved by comparing with the binary switching algorithm and the parallel genetic algorithm. In this paper, simulated annealing technique is introduced in the tabu search. Experimental results show that the modified tabu search algorithm can improve the channel distortion by comparing with the simple tabu search algorithm.

2. Tabu Search Algorithm

The tabu search approach was proposed by Glover [6]. The basic idea of the tabu search is to explore the search space of all feasible solutions by a sequence of moves and to forbid some search directions at a present iteration in order to avoid cycling and jump off local optima. Let S_t, s_c and s_b be the test solutions, the best solution of current iteration and the best solution of all iterations, respectively. Let V_t, v_c and v_b denote the objective function values for test solutions, the objective function value for the best solution of current iteration and the objective function value for the best solution of all iterations, respectively.

$S_t=\{s_t^1,s_t^2,...,s_t^{N_s}\}$, $s_t^i=\{s_t^i(1),s_t^i(2),...,s_t^i(N)\}$, $1\le i\le N_S$, $s_c=\{s_c(1),s_c(2),...s_c(N)\}$, $s_b=\{s_b(1),s_b(2),...,s_b(N)\}$ and $V_t=\{v_t^1,v_t^2,...,v_t^{N_s}\}$, where N_S is the number of test solutions and N is the number of codevector indices. The initail test solutions are generated randomly. After the first iteration, the test solutions are generated from the best solution of current iteration by swapping two indicies ramdomly. The tabu list memory stores the swapped indices only. It is a tabu condition if the swapped indicies to generate the test solution from the best solution of current iteration are the same as any records in the tabu list memory. The algorithm is as follows.

(i) Set the tabu list size T_S, the number of test solutions N_S and the maximum number of iterations I_m. Set the iteration counter $i=1$ and the insertion point of the tabu list $t_l=1$. Generate N_S initail solutions $S_t=\{s_t^1,s_t^2,...,s_t^{N_s}\}$ randomly, calculate the corresponding objective values $V_t=\{v_t^1,v_t^2,...,v_t^{N_s}\}$ using Eq. 3 and select the current best solution $s_c=s_t^j$, $j=\arg\min_l v_t^l$, $1\le l\le N_S$. Set $s_b=s_c$ and $v_b=v_c$.

(ii) Copy the current best assignment s_c to each test solution s_t^i, $1\le i\le N_S$. For each test solution s_t^i, $1\le i\le N_S$, generate two random integers r_1^i and r_2^i, $1\le r_1^i\le N$, $1\le r_2^i\le N$, $r_1^j\ne r_2^i$, N is the number of codeword indicies. Generate the new test solutions by swapping $s_t^i(r_1^i)$ and $s_t^i(r_2^i)$. Calculate the correspanding objective values $v_t^1,v_t^2,...v_t^{N_s}$ for the new test solutions.

(iii) Sort $v_t^1,v_t^2,...v_t^{N_s}$ in increasing order. From the best new test solution to the worst new test solution, if the new test solution is a non-tabu solution or it is a tabu solution but its objective value is better than the best value of all iteration v_b (aspiration level), then choose this new solution as the current best solution s_c and choose its objective value as the current best objective value v_c, go to step (iv); otherwise, try the next new test solution. If all new test solutions are tabu solutions, then go to step (ii).

(iv) If $v_b>v_c$, set $s_b=s_c$ and $v_b=v_c$. Insert the swapped indices of the current best solution s_c into the tabu list. Set the inserting point of the tabu list $t_l = t_l + 1$.

(v) If $t_l > T_S$, set $t_l = 1$. If $i < I_m$, set $i = i + 1$ and go to step (ii); otherwise, record the best codevector indicies assignment and terminate the algorithm.

3. Modified Tabu Search

We can see in the step (iii) of the tabu search algorithm, if the test solution is a non-tabu solution, we consider it as a current best solution unconditionally. However, this solution is often worse than the current best solution. Simulated annealing technique [2] can be introduced in this step to improve the performance. On the other hand, we often encounter the phenomena that the best solution in the tabu search is unchanged for a lot of iterations. We can make a slight modification in step (iv) and introduce in a parameter N_u called *unchanged times limit* to deal with this phenomena. At first, in step (i), set the initial temperature T_n, n=0, then we can modify the Step (iii) and Step (iv) as follows:

(iii) Sort $v_t^1, v_t^2, \ldots v_t^{N_S}$ in increasing order. From the best new test solution to the worst new test solution, if the new test solution $s_t^i, 1 \le i \le N_S$, is a non-tabu solution and its objective value v_t^i is less than v_c or it is a tabu solution but its objective value is less than the best value of all iterations v_b (aspiration level), then choose this new solution as the current best solution s_c and choose its objective value as the current best objective value v_c, go to step (iv); else if the new test solution $s_t^i, 1 \le i \le N_S$, is a non-tabu solution and its objective value v_t^i is larger than v_c, then accept this new solution as the current best solution s_c and choose its objective value as the current best objective value v_c with probability $\exp(\frac{v_c - v_t^i}{T_n})$, decreasing the annealing temperature, i.e. $T_{n+1} = T_n - \Delta T_n$, $n = n + 1$, if it is accepted then go to step (iv); otherwise, try the next new test solution. If all new test solutions are tabu solutions or all new test solutions are rejected, then go to step (ii).

(vi) If $v_b > v_c$, set $s_b = s_c$ and $v_b = v_c$. If v_b is unchanged for N_u iterations, then set $s_c = s_b$. Insert the swapped indices of the current best solution s_c into the tabu list. Set the inserting point of the tabu list $t_l = t_l + 1$.

4. Experiments And Conclusions

The test material for these experiments was a LENA image. It consists of 512×512 pixels with 8-*b* resolution. The codebook consisting 256 codewords is generated by the well know LBG algorithm [7]. The vector dimension is 16.

Experiments were carried out to test the performance of the proposed modified tabu search approach and the tabu search algorithm as well as the BSA algorithm. In this paper, the distribution of the codeword probability was set to a uniform distribution and the distortion between two codewords was measured in terms of mean squared error. For a given codeword index assignment, the distortion was computed using Eq. 3. We set $N_S = 20$, $T_S = 20$, $I_m = 100$, $\varepsilon = 0.01$.

Mean squared errors for ten runs of modified tabu search approach and tabu search algorithm for 256 codevectors are shown in table 1. From these experiments, the superiority of the modified tabu search algorithm compared with the tabu search algorithm and BSA algorithm is demonstrated.

Two examples of the relationship between the best solution of the current iteration and the best solution of all iterations for 0.01 bit error probability are illustrated in Fig. 2 (for TS) and Fig. 3 (for Modified TS with SA). In Fig. 2 we can see in the last fifty iterations, the TS algorithm has no ability to seek a better solution than before, while in Fig. 3 the MTS algorithm can seek a better solution than before. Because in MTS algorithm, when the best solution of all iterations is unchanged for a certain number of iterations, we will set the best solution of current iteration as the best solution of all iterations to start seeking a better solution from the best solution of all iterations again. In Fig. 3, the efficiency of the proposed algorithm is demonstrated.

Acknowledgements

This work was supported by National Science Council, Taiwan, R.O.C., under Grant No. NSC-87-2213-E-151-007.

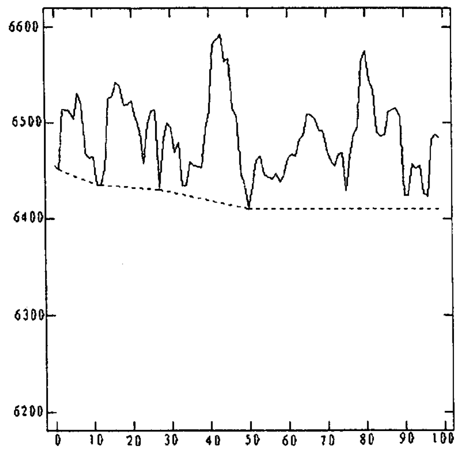

Fig. 2 *Mean squared error for the best solution of all iterations and the best solution of current iteration of the tabu search algorithm.*

— best value of current iteration

... best value of all iterations

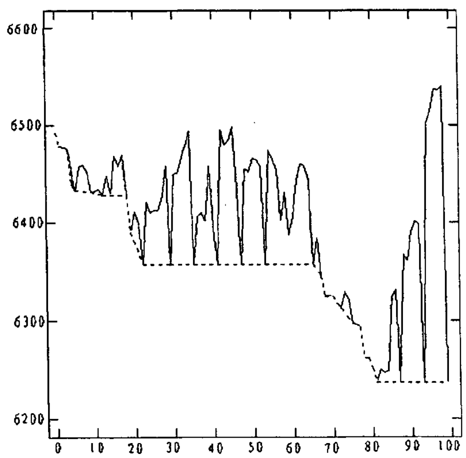

Fig. 3 *Mean squared error for the best solution of all iterations and the best solution of current iteration of the modified tabu search algorithm.*

— best value of current iteration

... best value of all iterations

References

[1] Gray, R. M., "Vector Quantization", IEEE ASSP Magazine, April 1984, pp. 4-29.

[2] Farvardin, N., "A Study of Vector Quantization for Noisy Channels", IEEE Trans., 1990, vol. IT-36, no. 4, pp. 799-809.

[3] Pan, J. S., Mcinnes, F. R. and Jack, M. A., " Application of Parallel Genetic Algorithm and Property of Multiple Global Optima to VQ Codevector Index Assignment for Noisy Channels", Electron. Lett, 1996, vol. 32, no. 4, pp. 296-297.

[4] Zeger, K. and Gersho, A., "Pseudo-Gray Coding", IEEE Trans. Communication, 1990, vol. 38, no. 12, pp. 2147-2158.

[5] Pan, J. S. and Chu, S. C., "Non-redundant VQ Channel Coding Using Tabu Search Strategy", Electron. Lett., 1996, vol. 32, no. 17, pp. 1545-1546.

[6] Glover, F., "Tabu Search. Part I", ORSA Journal on Computing, 1989, vol. 1, no. 3, pp. 190-206.

[7] Linde, Y., Buzo, A. and Gray, R. M., "An Algorithm for Vector Quantizer Design", IEEE Trans. Communications, 1980, vol. COM-28, no. 1, pp. 84-95.

Table 1: Mean Squred errors for 10 runs of tabu search approach, binary swiching algorithm and the modified tabu search approach with simulated annealing.

Seed	BSA	TSA	MTSA
1	6456.6471	6410.3457	6237.1710
2	6414.3576	6394.6789	6383.1157
3	6423.9860	6423.6908	6211.1256
4	6484.4183	6384.8098	6304.4508
5	6437.4753	6450.7843	6257.4589
6	6465.6138	6435.3270	6298.7870
7	6449.3648	6299.6289	6345.7407
8	6307.3897	6357.7430	6243.2809
9	6501.6786	6427.8007	6301.8097
10	6347.0982	6351.3477	6247.7477
Average	6428.8029	6393.6157	6283.0668