Available online at www.sciencedirect.com
ScienceDirect
ELSEVIER
April 2014, 21(2): 69–74
www.sciencedirect.com/science/journal/10058885
The Journal of China Universities of Posts and Telecommunications
http://jcupt.xsw.bupt.cn

# On the efficient search of punctured convolutional codes with simulated annealing algorithm

ZOU Wei-xia[1], WANG Zhen-yu[1] (✉), WANG Gui-ye[1], DU Guang-long[1], GAO Ying[2]

1. Key Laboratory of Wireless Universal Communications, Beijing University of Posts and Telecommunications, Beijing 100876, China
2. School of Electronic Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China

**Abstract**

Punctured convolution codes (PCCs) have a lot of applications in modern communication system. The efficient way to search for best PCCs with longer constraint lengths is desired since the complexity of exhaustive search becomes unacceptable. An efficient search method to find PCCs is proposed and simulated. At first, PCCs' searching problem is turned into an optimization problem through analysis of PCCs' judging criteria, and the inefficiency to use pattern search (PS) for many local optimums is pointed out. The simulated annealing (SA) is adapted to the non-convex optimization problem to find best PCCs with low complexity. Simulation indicates that SA performs very well both in complexity and success ratio, and PCCs with memories varying from 9 to 12 and rates varying from 2/3 to 4/5 searched by SA are presented.

**Keywords** PCCs, optimization problem, pattern search, simulated annealing

## 1 Introduction

According to Shannon second theory, when transmitting rate ($R_{\text{tran}}$) is less than channel capacity ($C_{\text{chan}}$), there exists the method to transmit information with errors converging to 0 when three properties such as random sequence, infinite length and probability decoding method are satisfied [1]. Aiming at such a purpose, various kinds of codes like linear block code, low-density parity-check code (LDPC), convolution codes are taken, and judging criteria that help to judge codes' error probability $P_{\text{e}}$ have been invented.

Convolutional codes (CCs) have good applications in modern communication systems with its high performance gaining and feasible practicability, due to the trellis structure which enables the efficient implementation of maximum likelihood decoding algorithm [2]. Thus, the consequently used Viterbi decoder gives rise to the significance of performance judging criteria like free distance and code spectrum. Unfortunately, Viterbi decoding's complexity will exponentially increase with the increment of constraint length and PCCs are invented with advantages of smaller complexity and more flexible rates [3]. Best PCCs with small constraint lengths have been exhaustively searched out and made into a codebook to support different complexities and rates requirements [4–5]. However, the rapidly developing communication system nowadays which requires more flexible rates and better performances needs PCCs with higher constraint lengths, and obtaining such PCCs by exhaustive search is not practical for the exponentially increasing searching time and complexity. Thus, an efficient way to search for best PCCs is desired.

Path pruning and puncturing are combined to achieve flexible PCCs with variable rates and decoding complexities that simplify the hardware implementation of the decoder, but PCCs' searching complexity still exponentially increases with longer constraint lengths [6]. A bidirectional efficient algorithm for searching trees (BEAST) is proposed to fast calculate PCCs' code spectrum to reduce the searching complexity [7–10], but even using BEAST in International Business Machines

Received date: 22-08-2013
Corresponding author: WANG Zhen-yu, E-mail: wzy221@gmail.com
DOI: 10.1016/S1005-8885(14)60288-0

Corporation's (IBM) complex hardware distribution cell broadband architecture could only let CCs exhaustively searched out instead of more complex PCCs with memory of 26 while CCs with memories of 27~29 are searched out using randomly methods [11]. PCCs deriving from 1/2 CCs with rate $R=(n-1)/n$, $n=5,6,\ldots,16$ are exhaustively searched out with memories of 7~10 [12], and PCCs deriving from 1/3 CCs are constructed from the above searched PCCs by using partial search, but the searching method is computationally complex and can not suit PCCs with higher memories and more flexible rates.

In order to deal with the exponentially increasing searching complexity, PCCs' searching problem is turned into an optimization problem with consideration of performance criteria and quantum genetic algorithm (QGA) is adapted to deal with the proposed PCCs' searching optimization problem instead of using exhaustive search method [13]. However, there are still rooms to reduce searching complexity for QGA needs frequent coding and decoding operations and needs parallel quantum computing mechanism which perplexes the algorithm to overcome the premature problem. Meanwhile, SA uses accepting probability mechanism to overcome the premature problem which has lower searching complexity compared with QGA, especially when the problem of QGA's sensitivity to initial parameters is not properly handled. Thus, an efficient method is proposed which adapts SA to deal with the proposed PCCs' searching optimization problem aiming to find a global optimum PCCs with lower complexity, and presents PCCs with memories of 9~12 searched by SA. The remainder of this article is organized as follows: the optimization model is described and the error probability and judging criteria are analyzed in Sect. 2. In Sect. 3, the simulation of SA is described and illustrated. Conclusions are shown in Sect. 4.

The followed symbols are used:

$\boldsymbol{P}$: punctured matrix of PCCs.

$\boldsymbol{G}$: generator matrix of PCCs.

$K$: constraint length of PCCs.

$m$: the memory of PCCs and equals to $K-1$.

$n$: the number of the output channel.

$k$: the number of the input channel.

$R$: ratio of number of input to number of output.

$d_{\mathrm{f}}$: free distance of PCCs.

## 2 Optimization model

### 2.1 Analysis of optimization problem and relevant criteria

According to Shannon second theory, the number of errors converges to 0 if $R_{\mathrm{tran}}$ is less than $C_{\mathrm{chan}}$ and three properties mentioned in Ref.[1] are satisfied. In order to achieve such a theoretical performance, various methods have been used to deal with CCs like changing formations to get randomicity in Turbo code, using Viterbi decoder to satisfy the underlying probability decoding and using practical backtracking length fluctuating from 5 times of constraint length to 10 times of constraint length to let the performance close to that of infinite length according to the converse theory of Shannon second theory $1 - C_{\mathrm{chan}}/R_{\mathrm{tran}} - 1/(nR_{\mathrm{tran}}) \leqslant P_{\mathrm{e}}$. Thus, how to construct a mapping ruler with certain generator matrix $\boldsymbol{G}$ and punctured matrix $\boldsymbol{P}$ to be best close to the Shannon boundary is the key point.

PCCs' performance is determined by the generator matrix $\boldsymbol{G}$ and the punctured matrix $\boldsymbol{P}$. A mathematical equation which directly indicates the nonlinear relationship between $\boldsymbol{G}$, $\boldsymbol{P}$ and the error probability performance can not be validated. How to choose suitable $\boldsymbol{G}$ and $\boldsymbol{P}$ to get the lowest error probability $P_{\mathrm{e}}$ is essentially the issue of how to find a mapping ruler with lowest intersections among PCCs' typical sets. Thus, Hamming distance or Euclidean distance which portrays the intersections among possible typical sets and reflects to what an extent would $P_{\mathrm{e}}$ be bigger than its low boundary $P_{\mathrm{e}} \approx 2^{-n(1-h(p)-R_{\mathrm{tran}})} = 2^{-n(C_{\mathrm{chan}}-R_{\mathrm{tran}})}$ ($h(p)$ is noise's entropy) is wisely considered. Based on such an idea, criteria like free distance and code spectrum are used to turn the searching problem into a mathematical optimization problem.

**Criterion 1** The PCCs' free distance $d_{\mathrm{f}}$ is:

$$d_{\mathrm{f}} = \min_{M_0 \neq 0} w(\boldsymbol{M}_\infty \boldsymbol{G}_\infty \boldsymbol{P}_\infty) \tag{1}$$

where $\boldsymbol{M}_\infty$, $\boldsymbol{G}_\infty$, $\boldsymbol{P}_\infty$ are the original code vector, the generator matrix and the punctured matrix with infinite size respectively. $M_0$ is the first element of $\boldsymbol{M}_\infty$. $w$ is the Hamming weight function. $d_{\mathrm{f}}$ is the minimum Hamming weight.

**Criterion 2** $A_{d_{\mathrm{f}}}$ is the number of codes which have the Hamming weight $d_{\mathrm{f}}$.

**Criterion 3** Bad convolutional code is the code whose generator matrix would cause error propagation and needs to be rejected. In mathematical calculation, it equals to