Generalized Punctured Convolutional Codes

Bartolomeu F. Uchôa-Filho, Richard Demo Souza, Cecilio Pimentel, and Mao-Chao Lin

***Abstract*— This letter introduces the class of generalized punctured convolutional codes (GPCCs), which is broader than and encompasses the class of the standard punctured convolutional codes (PCCs). A code in this class can be represented by a trellis module, the GPCC trellis module, whose topology resembles that of the minimal trellis module. The GPCC trellis module for a PCC is isomorphic to the minimal trellis module. A list containing GPCCs with better distance spectrum than the best known PCCs with same code rate and trellis complexity is presented.**

I. Introduction

A CONVOLUTIONAL code can be represented by a semi-infinite trellis consisting, after a short transient, of concatenated copies of a topological structure called *trellis module*. A trellis module M for a rate $R = k/n$ (i.e., a (n, k)) convolutional code C consists of $n' \leq n$ trellis sections (from depth 0 to depth n'), 2^{ν_t} states at depth t, 2^{b_t} edges emanating from each state at depth t, and l_t bits labeling each edge from depth t to depth $t+1$ (for $0 \leq t \leq n'-1$). McEliece and Lin [1] stated that the computational effort required by the Viterbi algorithm to decode a convolutional code is proportional to the total number of edge symbols per information bit in the trellis module representing the code. This is said to be the *trellis complexity of the module* M, denoted by $TC(M)$.

There can be many trellis modules describing the same code. The trellis complexity of the *conventional* trellis module for the (n, k) convolutional code C with memory order ν, denoted by M_{conv}, is given by $TC(M_{conv}) = (n/k)\,2^{\nu+k}$ symbols per bit. Punctured convolutional codes (PCCs) [1] form a special class of convolutional codes that can be described by an alternative, low-complexity trellis module, the PCC trellis module (M_{PCC}), with trellis complexity $TC(M_{PCC}) = (n/k)\,2^{\nu+1}$ symbols per bit. Some of the best known PCCs were tabulated in [2].

A theory of minimal trellis for convolutional codes has been developed in [1], [3]. The minimal trellis module $\widetilde{M}$ for the (n, k) convolutional code C consists of $n' = n$ trellis sections, k of which have branch complexity $\widetilde{b}_t = 1$ and the remaining $(n-k)$ trellis sections are informationless, i.e., a single edge leaves each state ($\widetilde{b}_t = 0$). There are $2^{\widetilde{\nu}_t}$ states at depth t, and $l_t = 1$ for all t. Since a low-complexity Viterbi decoder is desirable, we adopt henceforth the trellis complexity of $\widetilde{M}$, $TC(\widetilde{M})$, as the *trellis complexity of the convolutional code* C.

In this letter, we search for good (in a distance spectrum sense) (n, k) convolutional codes with fixed $TC(\widetilde{M})$. It appears that a convolutional code search taking this measure of complexity has only been considered in the literature by Tang and Lin [4]. The convolutional codes they found, all of which of rate $(n, n-1)$, had better weight spectrum or decoding complexity than the PCCs in [2]. Herein, we aim at finding convolutional codes better than PCCs for other code rates as well. To achieve this goal, we introduce a sufficiently broad class of convolutional codes, the *generalized punctured convolutional codes* (GPCCs), which encompasses the class of PCCs. A code in this class can be represented by a trellis module — the GPCC trellis module (M_{GPCC}) — that shares all of the topological characteristics of the minimal trellis, except possibly the minimality property. It will be shown that many of the $(n, n-1)$ codes found in [4] are GPCCs. Moreover, it is possible to define a template for the scalar generator matrix G_{scalar} of the GPCC which yields naturally to the minimal-span form [1], allowing the predetermination of the value of $TC(\widetilde{M})$ for an ensemble of GPCCs.

II. GPCCs

A (n, k) GPCC is a time-varying convolutional code of period n defined by the binary generator scalars g_i^t, where $t = 0, 1, \ldots, n-1$ is a phase index and, for each fixed t, $i = 0, 1, \ldots, \widehat{\nu}_t$, where $\widehat{\nu}_t$ is the memory order at phase t, with the following restrictions:

- $g_0^t = 0$ for all $t \in J$, and $g_0^t = 1$ for all $t \in I \backslash J$, where J is some subset of size $n-k$ of the set $I = \{0, 1, \ldots, n-1\}$;
- $\widehat{\nu}_{t+1} \leq \widehat{\nu}_t + \widehat{b}_t$, for $t = 0, 1, 2, \ldots, n-2$, and $\widehat{\nu}_0 \leq \widehat{\nu}_{n-1} + \widehat{b}_{n-1}$, where $\widehat{b}_t = 0$ if $t \in J$, and $\widehat{b}_t = 1$ if $t \in I \backslash J$.

Arranged in the "matrix module" (a vertical slice in G_{scalar} corresponding to one trellis module [1, eq. (2.4)]), the generator scalars are seen as (only the non-zero rows are shown):

$$\begin{bmatrix} g^0_{\widehat{\nu}_0} & & & & & & \\ \vdots & & g^{p-1}_{\widehat{\nu}_{p-1}} & & & & \\ g^0_1 & \ddots & \vdots & g^p_{\widehat{\nu}_p} & g^{p+1}_{\widehat{\nu}_{p+1}} & & \\ g^0_0 & & g^{p-1}_1 & \vdots & \vdots & & g^{n-1}_{\widehat{\nu}_{n-1}} \\ & & g^{p-1}_0 & g^p_1 & g^{p+1}_1 & \ddots & \vdots \\ & & & g^p_0 = 0 & g^{p+1}_0 & & g^{n-1}_1 \\ & & & & & & g^{n-1}_0 \end{bmatrix} \tag{1}$$

Manuscript received June 1, 2005. The associate editor coordinating the review of this letter and approving it for publication was Prof. Alexander Haimovich. This work was supported by CNPq (Brazil) under grants 301253/2004-8 and 302286/2004-7, and by NSCROC (R.O.C.) under grant NSC91-2213-E002-100.

Bartolomeu F. Uchôa-Filho is with GPqCom/EEL/CTC/UFSC, Florianópolis-SC, 88040-900, Brazil (e-mail: uchoa@eel.ufsc.br).

Richard Demo Souza is with CPGEI/CEFET-PR, Curitiba-PR, 80230-901, Brazil (e-mail: richard@cpgei.cefetpr.br).

Cecilio Pimentel is with CODEC/DES/UFPE, Recife-PE, Brazil (e-mail: cecilio@ufpe.br).

Mao Chao-Lin is with the Dept. of Electrical Engineering, National Taiwan University, Taipei, Taiwan, R.O.C. (e-mail: mclin@cc.ee.ntu.edu.tw).

Digital Object Identifier 10.1109/LCOMM.2005.12013.

where we have considered that $p \in J$. The lack of information ($\widehat{b}_p = 0$) at phase p causes a "leap" in (1), and g_0^{p+1} is placed to the right of g_0^p (and not following the diagonal, as usual).

The GPCC trellis module (M_{GPCC}) for a rate (n, k) GPCC has state complexity profile (obtained by examining the columns of (1)) $\widehat{\boldsymbol{\nu}} = \{\widehat{\nu}_0, \widehat{\nu}_1, \ldots, \widehat{\nu}_{n-1}\}$, branch complexity profile $\widehat{\boldsymbol{b}} = \{\widehat{b}_0, \widehat{b}_1, \ldots, \widehat{b}_{n-1}\}$, and a single bit (i.e., $l_t = 1$) labeling each edge for all $t \in I$. The bit value at depth t is obtained from (1) in the usual way. The restrictions defined previously were imposed to avoid unreachable states in the GPCC trellis module. The trellis complexity of M_{GPCC} is given by:

$$TC(M_{GPCC}) = \frac{1}{k} \sum_{t=0}^{n-1} 2^{\widehat{\nu}_t + \widehat{b}_t} \tag{2}$$

symbols per bit.

We now show that the PCCs form a subclass of the GPCCs. Consider for simplicity a (n, k) PCC with memory ν where $k > n/2$. It is shown in [6] that this PCC can be obtained by puncturing a rate $1/n$ time invariant convolutional code of memory ν or, equivalently, a rate 1/2 periodically time-varying convolutional code (PTVCC) of period k and memory ν. Assuming the latter case, then, for the punctured phases, the generator sub-matrices of the PCC are of the form $G_i^t = [G_i^t(1) \;\; x]$, where x represents a punctured output, and, for the non-punctured phases, $G_i^t = [G_i^t(1) \;\; G_i^t(2)]$. The generator scalars of the GPCC are obtained as follows. Let u_t denote the number of non-punctured phases in the PTVCC occurring prior to phase t. By convention, $u_0 = 0$. Then, for $0 \leq t \leq k-1$, set $g_i^{t+u_t} = G_i^t(1)$ and, if the t-th phase is non-punctured, then also set $g_i^{t+u_t+1} = G_i^t(2)$. The state and branch complexity profiles of M_{GPCC} for this PCC are given as follows. For $0 \leq t \leq k-1$, set $\widehat{\nu}_{t+u_t} = \nu$, $\widehat{b}_{t+u_t} = 1$ and, if the t-th phase is non-punctured, then also set $\widehat{\nu}_{t+u_t+1} = \nu + 1$, $\widehat{b}_{t+u_t+1} = 0$. From the construction of the minimal trellis module [1], we can see that the state complexities of M_{GPCC} and $\widetilde{M}$ coincide, i.e., $\widehat{\nu}_t = \widetilde{\nu}_t$ for all t. According to a property of the minimal trellis for block codes (see, for instance, [5, Theorem 4.26]), which can be adapted to convolutional codes, the equality above for all t implies that the two trellis modules are isomorphic. Moreover, for $\widehat{\boldsymbol{\nu}}$ and $\widehat{\boldsymbol{b}}$ as defined above, the summation in (2) becomes $n \cdot 2^{\nu+1}$. Therefore, for any PCC, M_{GPCC} is isomorphic to $\widetilde{M}$, and $TC(\widetilde{M}) = TC(M_{GPCC}) = TC(M_{PCC})$.

III. Code Search Results

In order to find good GPCCs, we first calculated the value of $TC(\widetilde{M})$ for the existing PCCs [2]. We then proposed templates for G_{scalar} of GPCCs. By placing in this matrix 0's and 1's in specific positions, while the others were set free to assume any binary value, we could define ensembles of GPCCs with the same $TC(\widetilde{M})$ as for the existing PCCs.

As an example, consider the best (5,3) PCC with $\nu = 4$, found by an exhaustive search in [2]. This code has $d_f = 6$ and distance spectrum 18,0,139,0,1210,... The state complexity profile of the GPCC trellis module for this PCC is $\widehat{\boldsymbol{\nu}} = (4, 4, 5, 4, 5)$, and the branch complexity profile $\widehat{\boldsymbol{b}} = (1, 1, 0, 1, 0)$. Since this code is a PCC, by the property presented at the end of Section II we have that $\widetilde{\boldsymbol{\nu}} = \widehat{\boldsymbol{\nu}}$ and $\widetilde{\boldsymbol{b}} = \widehat{\boldsymbol{b}}$. So $TC(\widetilde{M}) = 53.33$ symbols per bit. To exemplify our code search, consider the following template for the "matrix module" of a (5,3) GPCC with state and branch complexity profiles of the GPCC trellis module $\widehat{\boldsymbol{\nu}} = (4, 4, 5, 5, 6)$ and $\widehat{\boldsymbol{b}} = (1, 1, 0, 1, 0)$:

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ \overline{1} & 0 & 0 & 0 & 0 \\ * & 1 & \underline{1} & 1 & \overline{1} \\ * & * & \overline{1} & 0 & 0 \\ * & * & * & * & * \\ \underline{1} & * & * & * & * \\ 0 & \underline{1} & * & * & * \\ 0 & 0 & 0 & \underline{1} & * \end{bmatrix} \tag{3}$$

where the nontrivial binary entries are marked by asterisk, and the positions of the leading (underlined) and trailing (overlined) 1's of each row impose state and branch complexity profiles of the minimal trellis module $\widetilde{\boldsymbol{\nu}} = (4, 4, 5, 4, 5)$ and $\widetilde{\boldsymbol{b}} = (1, 1, 0, 1, 0)$, and thus guarantee the same trellis complexity ($TC(\widetilde{M}) = 53.33$ symbols per bit) for any GPCC in this ensemble. After defining a non-exhaustive series of such templates for a fixed rate and $TC(\widetilde{M})$, we searched exhaustively within the corresponding ensembles and looked for the best GPCCs (note that it is not necessary to build the minimal trellis during the search procedure, since $TC(\widetilde{M})$ is fixed *a priori*). We found from template (3) a (5,3) GPCC with $d_f = 6$ and distance spectrum 15,0,136,0,1208,... Although the GPCC trellis module for this code is more complex than the GPCC trellis module for the PCC, the minimal trellis modules for the two codes have exactly the same state and branch complexity profiles ($\widetilde{\boldsymbol{\nu}} = (4, 4, 5, 4, 5)$ and $\widetilde{\boldsymbol{b}} = (1, 1, 0, 1, 0)$). Therefore, the two codes have the same trellis complexity, but the GPCC we found has a better distance spectrum.

As a second example, consider the best (4,3) PCC with $\nu = 4$, listed in [2]. This code has $d_f = 4$ and distance spectrum 5,42,134,662,3643, ... Tang and Lin [4] have found a (4,3) convolutional code with the same d_f and $TC(\widetilde{M})$, but with distance spectrum 3,44,160,638,3558,... To see that this code is in fact a GPCC, first write its "matrix module":

$$G_{mod} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{4}$$

We can perform row operations on (4). Let $[\kappa]$ denote the κ-th row of a matrix, where $\kappa = 1, 2, \ldots$. Performing $[3\kappa] \leftarrow [3\kappa] + [3\kappa + 2]$, we turn G_{mod} in (4) into a GPCC form. We thus have a GPCC whose GPCC trellis module has state complexity profile (4,5,6,5) and branch complexity profile (1,1,0,1), satisfying the topological restrictions. Note that this code is not a PCC, and has distance spectrum better than that of the best PCC of the same rate and trellis complexity.

TABLE I
SOME GOOD GENERALIZED PUNCTURED CONVOLUTIONAL CODES

R	ν	$G(D)$	$TC(\widetilde{M})$	d_f	Spectrum
2/3	3^t	[1 2 3; 6 1 3]	24.00	4	2,11,34,109,366
2/3	4^t	[1 3 3; 10 12 7]	48.00	5	4,17,54,192,681
3/4	3^b	[3 1 1 0; 0 3 1 1; 2 2 3 3]	21.33	4	29,0,532,0,10059
3/4	4^t	[1 1 2 3; 2 1 1 1; 4 6 4 1]	42.67	4	3,44,160,638,3558
3/5	3^n	[3 1 0 1 0; 0 3 3 3 2; 2 2 2 1 1]	26.67	4	1,5,11,38,112
3/5	4^n	[3 3 2 2 3; 2 3 3 1 0; 6 0 2 3 1]	53.33	6	15,0,136,0,1208
4/5	3^t	[1 1 1 1 0; 2 2 1 1 1; 0 2 2 1 1; 2 2 0 2 3]	20.00	3	5,36,200,1065,5893
4/5	4^b	[3 1 1 1 0; 2 3 0 1 1; 2 2 3 1 0; 0 0 2 3 2]	40.00	4	30,126,815,4822,29046
4/7	3^n	[1 1 0 1 1 1 1; 2 1 1 1 0 1 0; 0 2 2 3 3 1 0; 2 2 2 0 0 1 1]	28.00	5	4,13,33,81,203
4/7	4^n	[3 1 0 1 1 0 1; 2 3 3 3 2 0 1; 2 2 2 1 1 1 0; 0 2 0 2 0 3 3]	56.00	6	8,27,46,143,380
5/6	3^t	[1 1 0 1 0 0; 0 1 1 0 1 0; 2 0 0 1 0 1; 2 2 2 0 1 1; 0 0 0 0 2 3]	19.20	3	15,96,601,3903,25325
5/6	4^b	[1 0 0 1 1 1; 2 1 1 1 0 0; 0 2 1 1 0 1; 2 2 2 1 1 1; 0 2 0 2 1 1]	38.40	4	111,0,5628,0,293724
5/7	3^n	[1 1 0 0 1 1 1; 0 1 1 1 1 0 0; 2 0 0 1 1 1 0; 2 2 2 0 1 0 0; 0 0 2 2 0 1 1]	22.40	4	15,55,174,798,3214
5/7	4^b	[1 1 0 0 1 1 1; 2 1 1 1 0 1 0; 2 2 2 1 0 0 1; 0 2 0 2 1 1 0; 2 0 0 0 2 1 1]	44.80	4	2,27,109,445,1955

^{t} Code found in [4], but with different $G(D)$; ^{b} Code listed in [2]; ^{n} New code found in this study by a random search.

Some good GPCCs are tabulated in Table I for different code rates R, and memory order ν of the corresponding PCC (which has the same $TC(\widetilde{M})$ of the listed GPCC). The polynomial generator matrices $G(D)$ are shown in octal form with the highest power in D in the most significant bit of the representation (*e.g.* $6 \equiv D + D^2$). Table I shows several $(n-1, n)$ GPCCs with the same distance spectrum of the best $(n-1, n)$ codes listed in [4]. For other code rates, the new GPCCs shown in Table I have better distance spectrum than the corresponding PCCs [2] with the same trellis complexity.

IV. CONCLUSIONS

The trellis complexities considered in this paper were restricted to those of the PCCs in [2]. The codes listed in Table I only improved the distance multiplicity, however a whole bunch of more complex convolutional codes with greater d_f exist. The study of GPCCs with other code rates and various trellis complexities is a research topic currently being investigated by the authors. For example, by placing in (3) the leading and trailing 1's in specific positions we found a (5,3) GPCC with $d_f = 5$ and distance spectrum 5,13,38,113,303,..., but with $TC(\widetilde{M}) = 37.33$ symbols per bit.

REFERENCES

[1] R. J. McEliece and W. Lin, "The trellis complexity of convolutional codes," *IEEE Trans. Inform. Theory*, vol. 42, pp. 1855-1864, Nov. 1996.

[2] I. E. Bocharova and B. D. Kudryashov, "Rational rate punctured convolutional codes for soft-decision Viterbi decoding," *IEEE Trans. Inform. Theory*, vol. 43, pp. 1305-1313, July 1997.

[3] V. Sidorenko and V. Zyablov, "Decoding of convolutional codes using a syndrome trellis," *IEEE Trans. Inform. Theory*, vol. 40, pp. 1663-1666, Sept. 1994.

[4] H.-H. Tang and M.-C. Lin, "On $(n, n-1)$ convolutional codes with low trellis complexity," *IEEE Trans. Commun.*, vol. 50, pp. 37-47, Jan. 2002.

[5] A. Vardy, "Trellis structure of codes," *Handbook of Coding Theory*, vol. II (V.S. Pless and W.C. Huffman, eds.) Amsterdam, The Netherlands: North-Holland, (1998).

[6] B. F. Uchôa Filho and R. Palazzo, Jr., "Unit-memory codes with simplified maximum likelihood decoding", in *Proc. ISITA'2000*, vol. I, pp. 13-16, Nov. 2000.