Project	**IEEE 802.16 Broadband Wireless Access Working Group <http://ieee802.org/16>**	
Title	**Larger CTC block sizes for OFDMA**	
Date Submitted	**2005-03-12**	
Source(s)	John Benko, Marie-Helene Hamon France Telecom Research & Development 801 Gateway Blvd. Suite 500 South San Francisco, CA 94080	Voice: +1-650-875-1593 Fax: +1-650-875-1505 John.Benko@francetelecom.com, mhelene.hamon@francetelecom.com
Re:	IEEE P802.16-2004/Cor1-D1 (2005-02-11)	
Abstract	Optimized interleaver parameters for CTCs (Convolutional Turbo Codes) are given to support a better range of block sizes for OFDMA subchannelization	
Purpose	To modify the given tables, with accompanying text, in this contribution into IEEE P802.16-2004/Cor1-D1 (2005-02-11)	
Notice	This document has been prepared to assist IEEE 802.16. It is offered as a basis for discussion and is not binding on the contributing individual(s) or organization(s). The material in this document is subject to change in form and content after further study. The contributor(s) reserve(s) the right to add, amend or withdraw material contained herein.	
Release	The contributor grants a free, irrevocable license to the IEEE to incorporate material contained in this contribution, and any modifications thereof, in the creation of an IEEE Standards publication; to copyright in the IEEE's name any IEEE Standards publication even though it may include portions of this contribution; and at the IEEE's sole discretion to permit others to reproduce in whole or in part the resulting IEEE Standards publication. The contributor also acknowledges and accepts that this contribution may be made public by IEEE 802.16.	
Patent Policy and Procedures	The contributor is familiar with the IEEE 802.16 Patent Policy and Procedures <http://ieee802.org/16/ipr/patents/policy.html>, including the statement "IEEE standards may include the known use of patent(s), including patent applications, provided the IEEE receives assurance from the patent holder or applicant with respect to patents essential for compliance with both mandatory and optional portions of the standard." Early disclosure to the Working Group of patent information that might be relevant to the standard is essential to reduce the possibility for delays in the development process and increase the likelihood that the draft publication will be approved for publication. Please notify the Chair <mailto:chair@wirelessman.org> as early as possible, in written or electronic form, if patented technology (or technology under patent application) might be incorporated into a draft standard being developed within the IEEE 802.16 Working Group. The Chair will disclose this notification via the IEEE 802.16 web site <http://ieee802.org/16/ipr/patents/notices>.	

Larger CTC block sizes for OFDMA

John Benko, Marie-Helene Hamon
France Telecom Research & Development

Motivation

The concatenation/fragmentation scheme for CTC blocks is currently limited to a maximum of 60 byte data blocks. Fragmenting large packets into many small blocks yield sub-optimal performance. The performance is affected adversely in two ways. First, performance gain from FEC codes decreases with smaller block sizes. Second, fewer (larger sized) blocks will have a better chance of successful transmission than more (smaller sized) blocks.

A modified concatenation rule for CTC blocks is proposed here. This rule serves to correct the inefficiencies in the current rule.

With the current standard, the only way to see the performance gains with higher block sized CTCs is to use the *optional* H-ARQ mode. Optimized code performance (by increasing block size) should not be limited to H-ARQ, especially when it is straightforward to implement. Assuming a decoder is available to handle 600 byte data packets (max block size of H-ARQ), the added complexity to the original CTC implementation is negligible, when increasing the number of possible block-sizes. To implement this *only* requires the additional storage of the new P0, P1, P2, P3 parameters. This will ensure maximum performance gain with the larger block sizes. In addition to the new large block sizes, a modification to the concatenation rules is also proposed.

Performance

In order for packets to be received correctly, all block fragments must be received without error. A direct result of increasing the block sizes is a decrease in the number of fragmented blocks needed to be sent. Due to the concatenation scheme, the error rate performance of a burst is dependent on the number of FEC blocks and the size of each block. The concatenation ensures the minimum length of the last two FEC blocks to be at least half of the maximum FEC block length. This results in the concatenation of a large packet into a number of max FEC block sizes plus 2 blocks of FEC size at least _ of the max FEC block length.

Below two examples are given:

QPSK, Rate = _, 576 byte and 1500 byte MTU packets, yielding 96 and 250 sub-channels respectively.

Concatenation Scheme	**J** (QPSK, R=1/2)	**# Blocks** 576 bytes	**Coded Blocks(bytes)** 576 byte data packet	**# Blocks** 1500 byte	**Coded Blocks (bytes)** 1500 byte data packet
Current	10	**10**	8*120, 2*96	**26**	24*120, 2*60
Proposed	40	**3**	1*480, 1*348, 1*324	**7**	5*480, 2*300

In the case of a 1500 byte MTU packet, the proposed scheme will reduce the number of transmitted block from 26 to 7. This will result in a significant decrease in PER. The first figure depicts the performance obtained with the CTC blocksizes involved in the 1500-byte MTU packet example (current and new concatenation scheme) and show the gains when larger blocksizes are employed. The second figure shows the potential performance improvement with the new concatenation scheme.

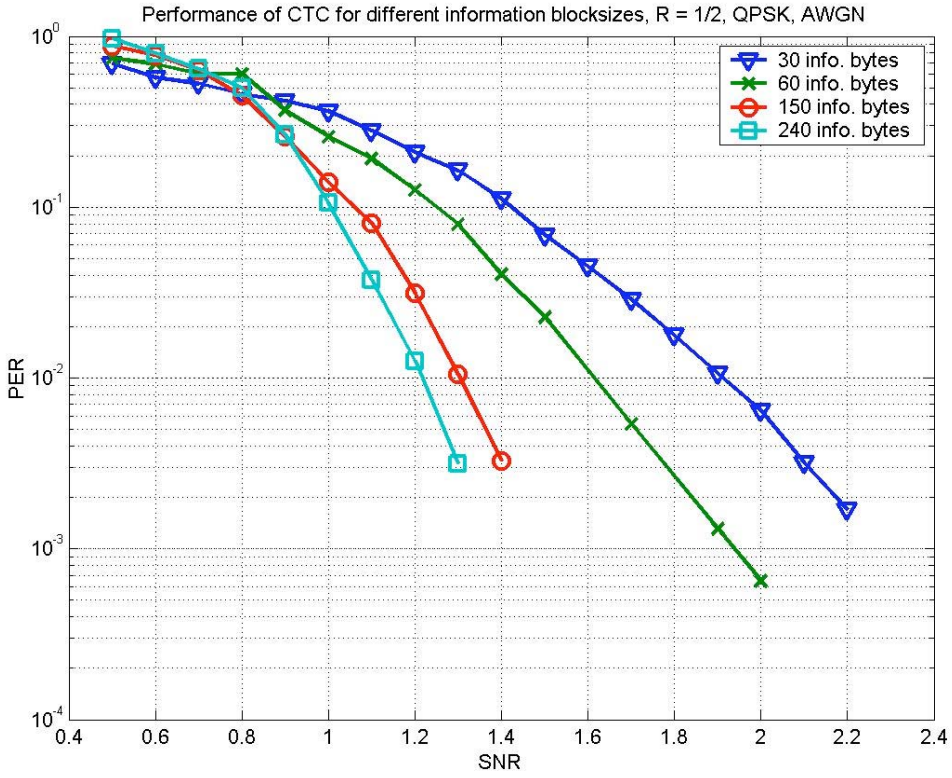

Figure 1 – Performance of CTC for different blocksizes (1500-byte MTU packet example)

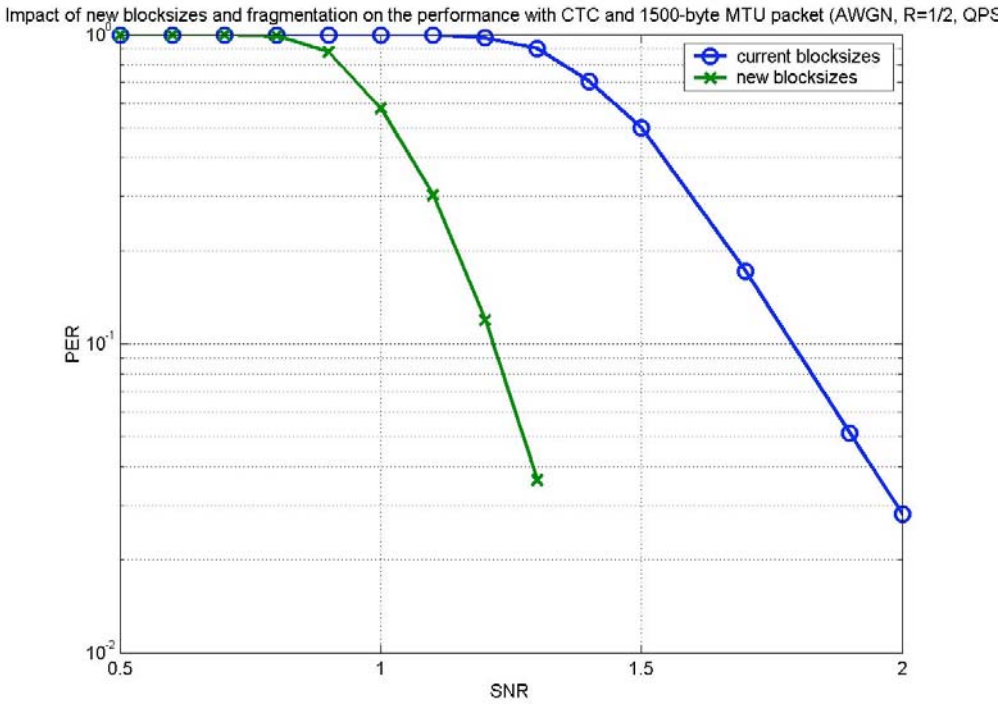

Figure 2 – Performance comparison of the current concatenation scheme and the proposed solution

Proposed Solution

In this proposal the maximum block size, which was previously only 60 information bytes in non H-ARQ mode, will be increased to a maximum of 240 information bytes. This corresponds to an increase in subchannel slots used in QPSK from 10 to up to 40, and in 64QAM from 3 up to 12. Outer interleaver parameters for the new block sizes are also defined in Table 327a.

Suggestion: To replace Table 324 with 324a, 325 with 325a, and 326 with 326a, to Section 8.4.9.2.3.1, and Table 329 with 329a in Section 8.4.9.2.3.4.2.

1. Section 8.4.9.2.3.1

Table 324a – Subchannel concatenation rule for CTC

Number of Subchannels	Subchannels concatenated
n ≤= j AND n mod 7 ≠ 0	1 block of n slots
n<=j AND n mod 7 = 0	1 block of 4n/7 slots 1 block of 3n/7 slots

n > j	If(n mod j = 0)
	k blocks of j slots
	else
	(k-1) blocks of j slots
	1 block of L_{b1} slots
	1 block of L_{b2} slots
	Where:
	L_{b1} = ceil((m+j)/2)
	L_{b2} = floor((m+j)/2)
	If (L_{b1} mod 7= 0) or (L_{b2} mod 7= 0)
	$L_{b1} = L_{b1} + 1$; $L_{b2} = L_{b2} - 1$;

Table 325a – Encoding subchannel concatenation for different rates in CTC

Modulation and rate	j
QPSK 1/2	j = 40
QPSK 3/4	j = 26
QAM16 1/2	j = 20
QAM16 3/4	j = 13
QAM64 1/2	j = 13
QAM64 2/3	j = 10
QAM64 3/4	j = 8
QAM64 5/6	j = 8

Table 326a – CTC channel coding per Modulation

Data block size (byte)	Encoded data block size (bytes)								N	P0	P1	P2	P3
	QPSK		16QAM		64QAM								
	1/2	3/4	1/2	3/4	1/2	2/3	3/4	5/6					
6	12	-	-	-	-	-	-	-	24	5	0	0	0
9	-	12	-	-	-	-	-	-	36	11	18	0	18
12	24	-	24	-	-	-	-	-	48	13	24	0	24
18	36	24	-	24	36	-	-	-	72	11	6	0	6
24	48	-	48	-	-	36	-	-	96	7	48	24	72
27	-	36	-	-	-	-	36	-	108	11	54	56	2
30	60	-	-	-	-	-	-	36	120	13	60	0	60
36	72	48	72	48	72	-	-	-	144	17	74	72	2
45	-	60	-	-	-	-	-	-	180	11	90	0	90
48	96	-	96	-	-	72	-	-	192	11	96	48	144
54	108	72	-	72	108	-	72	-	216	13	108	0	108
60	120	-	120	-	-	-	-	72	240	13	120	60	180
66	132	-	-	-	-	-	-	-	264	23	2	160	30
72	144	96	144	96	144	108	-	-	288	23	50	188	50
78	156	-	-	-	-	-	-	-	312	23	102	64	38
81	-	108	-	-	-	-	108	-	324	11	172	164	16
90	180	120	-	120	180	-	-	108	360	29	56	0	68
96	192	-	192	-	-	144	-	-	384	29	68	140	56
99	-	132	-	-	-	-	-	-	396	29	36	128	76
102	204	-	-	-	-	-	-	-	408	29	124	204	40

108	216	144	216	144	216	-	144	-	432	13	0	4	8
114	228	-	-	-	-	-	-	-	456	31	100	224	104
117	-	156	-	-	-	-	-	-	468	31	98	220	98
120	240	-	240	-	-	180	-	144	480	31	52	240	52
132	264	-	264	-	-	-	-	-	528	31	24	36	104
135	-	180	-	-	-	-	180	-	540	31	42	248	34
138	276	-	-	-	-	-	-	-	552	35	14	136	6
144	288	192	288	192	288	216	-	-	576	31	42	232	18
150	300	-	-	-	-	-	-	180	600	37	20	152	0
153	-	204	-	-	-	-	-	-	612	37	6	164	14
156	312	-	312	-	-	-	-	-	624	37	312	156	468
162	324	216	-	216	324	-	216	-	648	37	62	160	34
171	-	228	-	-	-	-	-	-	684	37	108	136	8
174	348	-	-	-	-	-	-	-	696	37	0	128	12
180	360	240	360	240	360	-	-	216	720	37	92	100	68
186	372	-	-	-	-	-	-	-	744	37	54	196	50
192	384	-	384	-	-	288	-	-	768	19	384	216	600
198	396	264	-	264	396	-	-	-	792	41	0	228	24
204	408	-	408	-	-	-	-	-	816	37	408	204	612
207	-	276	-	-	-	-	-	-	828	41	136	288	192
216	432	288	432	288	432	324	288	-	864	19	2	16	6
222	444	-	-	-	-	-	-	-	888	43	10	220	18
225	-	300	-	-	-	-	-	-	900	43	8	56	20
228	456	-	456	-	-	-	-	-	912	43	96	8	124
234	468	312	-	312	468	-	-	-	936	43	120	140	124
240	480	-	480	-	-	360	-	288	960	43	52	120	28

2. Section 8.4.9.2.3.4.2 Subblock interleaving

Table 329a – Parameters for subblock interleavers

Block size (bits) N_{EP}	N	Subblock Interleaver Parameters		Block size (bits) N_{EP}	N	Subblock Interleaver Parameters	
		m	J			m	J
48	24	3	3	960	480	8	2
72	36	4	3	1056	528	8	3
96	48	4	3	1080	540	8	3
144	72	5	3	1104	552	8	3
192	96	5	3	1152	576	8	3
216	108	5	4	1200	600	8	3
240	120	6	2	1224	612	8	3
288	144	6	3	1248	624	8	3
360	180	6	3	1296	648	8	3
384	192	6	3	1368	684	8	3
432	216	6	4	1392	696	8	3
480	240	7	2	1440	720	8	3
528	264	7	3	1488	744	8	3
576	288	7	3	1536	768	8	4
624	312	7	3	1584	792	8	4
648	324	7	3	1632	816	8	4
720	360	7	3	1656	828	8	4
768	384	7	3	1728	864	8	4
792	396	7	4	1776	888	8	4
816	408	7	4	1800	900	8	4
864	432	7	4	1824	912	8	4
912	456	8	2	1872	936	8	4
936	468	8	2	1920	960	9	2