Modified Interleaver Update 1/20/19

Benjamin Davis
Computer Engineering
California Polytechnic State University
San Luis Obispo, CA
benjamin.j.davis96@gmail.com

I. Problem With Current Interleaving Algorithm

The interleaving jamming scheme for 802.11 proposed by Vo-Huu et al. takes advantage of the current 802.11 interleaving scheme. This scheme is developed in such a way as to ensure that any two bits which were adjacent in the original coded data sequence are going to be a multiple of three subcarriers apart. This allowed for effective jamming of the entire signal through only jamming a third of the entire set of digital subcarriers (DSC). In order to prevent the attack from being performed this easily it is necessary to develop an interleaving scheme which does not have the same interval repetition property.

II. Proposal

In order to prevent this attack it is necessary to abandon the current first round permutation which essentially takes the coded input stream in column-major order and outputs them in row-major order. Instead I propose to first take the input data and store it in 48 blocks which are the size of the bits per subcarrier (*b*) so that the first *b* bits are in subcarrier one, the second *b* bits are in subcarrier two, etc. Then use an array of *b* offsets in order to shift the *i*th bit of each subcarrier into the first available space in the appropriate subcarrier. This output can then be fed into the current second round of permutation as defined by 802.11. This in essence results in replacing the current first-round of permutation with a new algorithm which splits adjacent bits by a non-constant number of subcarriers.

The pseudo-code for the algorithm as tested uses an array of offsets generated by linear congruential generator defined by

$$X_{n+1} = \lfloor 3.1 * X_n + 13 \rfloor \, mod \; 48$$

with a seed of $X_0 = 14$. This results in the offset array [8 37 31 13 5 28 3 22 33 19 23 36]. This offset array exhibits appropriate behavior for this application as the difference between any given subcarriers is not constant. It also contains 12 offsets which is more than the *b* value for any current 802.11 modulation scheme (QAM256 being the highest with *b* = 8). So by using this offset array the following pseudo code can be developed for the proposed first round permutation algorithm. (Note the symbol '<=' is being used as the bit string concatenation operator).

```
Input:
  Bits per Subcarrier: b
  Coded Bit String: $X_0X_1…X_m$
     Where m = b*48
Output:
  DSC Array: newDSC
Start:
  offset = [8 37 31 … 23 36]
  initDSC[48]
  for i = 0 to m-1
     initDSC[floor(i/b)] <= $X_i$

  newDSC[48]
  for s = 1 to 48
     for i = 1 to b
        dsc = (s + offset[i]) % 48
        newDSC[dsc] <= initDSC[s][i]
```

Here it is important to note that because the offsets being applied do not depend upon the initial subcarrier each new subcarrier is ensured to have the same number of bits. Simple brute force analysis over all possible values of *b* show that this algorithm solves the problem of constant DSC intervals for adjacent bits.

III. Analysis

This proposed algorithm solves some of the problem which is exploited by interleaving jamming. In avoiding constant DSC intervals it forces the attacker to determine which modulation scheme is being used in order to determine which channels contain adjacent bits in the coded bit stream.

Additionally, in the jamming scheme proposed by Vo-Huu et al. jamming every third DSC results in a contiguous block of *b**16 bits being jammed. This sort of behavior is avoided in this interleaving scheme, and it can be verified via brute force analysis that more than 16 DSC need to be jammed for all values of *b* except 1 (BPSK) (Table 1). It can also be demonstrated using brute force that in order to jam *b* substrings of 16 adjacent bits more than a third of the DSCs must be jammed (25 for *b*=4).

Bits per Subcarrier (*b*)	Minimum DSC jammed for *b**16 adjacent bits jammed
1	16
2	32
4	42
6	48
8	48

Table 1. Minimum DSC for Equivalent Jamming (Calculated via Brute Force)

Overall, these two benefits of this interleaving algorithm make it a strong candidate to replace the current 802.11 first round of interleaving.