Design of a Gigabit DSL modem using super orthogonal complete complementary codes

Jacques H. van Wyk* and Louis P. Linde

Department of Electrical, Electronic and Computer Engineering, University of Pretoria, Lynnwood Road, Pretoria 0002, South Africa

*Correspondence: J. H. van Wyk, Department of Electrical, Electronic and Computer Engineering, University of Pretoria, Lynnwood Road, Pretoria 0002, South Africa. E-mail: jhvanwyk@up.ac.za

Abstract

This paper describes the preliminary design and simulation towards a gigabit digital subscriber line modem that provides multi-user interference-free communication in a far-end crosstalk dominated environment by incorporating super orthogonal complete complementary spreading into the existing xDSL modem architecture. This is in contrast to existing vectoring and dynamic spectral management techniques that use joint processing and pre-processing of each user's signal to mitigate far-end crosstalk. A novel code allocation algorithm is introduced to provide all users with equal data rate ratios, even with bad line profiles and high-required data rates. Preliminary simulation results show that 1 Gbps aggregate throughput can be obtained for the system over a single pair of 0.5 mm copper wire over a distance from 180 to 150 m, depending on whether ADSL2+ or VDSL2 Profile 30a service bands are avoided, if present.

1. Introduction

Telcos have access to technologies such as ADSL2+ and VDSL2 to achieve 10/20/40 and even 100 Mbps performance, but lack affordable alternatives to fibre to the home and higher bit rate xDSL. Digital subscriber line (DSL) systems transmit data to and from various customer premises over bundles of copper wire encapsulated within a cable binder. The interference between adjacent lines (pairs) is known as crosstalk. Crosstalk limits the achievable data rates of DSL systems. Near-end crosstalk (NEXT) is the crosstalk seen by a receiver from other transmitters on the same side and far-end crosstalk (FEXT) is the crosstalk seen by a receiver from transmitters on the adjacent side. NEXT interference is removed by using frequency-division duplexing to separate upstream and downstream, although time-division duplexing can also be used. FEXT is the only significant form of system crosstalk that cannot be removed with the current DSL architecture [1], although vectoring and dynamic spectral management use clever signal processing to mitigate the effect of FEXT. Vectoring is usually implemented at the access node, where the loops' ends are co-located and the signals of all users are available for joint processing [2]. It is however not compatible with unbundled systems, as it cannot manage the signals of other service providers' modems [3]. The new

G.fast (G.9700 and G.9701) standard, approved by the International Telecommunications Union in December 2014, aims to provide higher bit rate alternatives for xDSL [4]. Shorter distances also means that FEXT becomes a dominant factor. Higher-data rates can be achieved by minimising or even removing the FEXT interference effect. Spreading techniques like multi-carrier direct sequence code division multiple access and concatenated direct sequence CDMA have been investigated for DSL systems [5, 6], but usually to replace the whole DSL architecture. In this paper, we show a novel technique where a 'spreading block' is added to the existing DSL architecture, using super orthogonal complete complementary (SOCC) spreading codes to create a multi-user interference (MUI) free environment, even in the presence of severe FEXT crosstalk. In Section 2, the properties of SOCC codes are discussed. In Section 3, the proposed system architecture will be discussed. In Section 4, the allocation of codes from the pool in each resource block (RB) is presented. A complexity analysis of the code allocation algorithm is performed in Section 5. The co-location of gigabit digital subscriber line (GDSL) with other xDSL services is described in Section 6. Finally, we present a conclusion of this paper in Section 7.

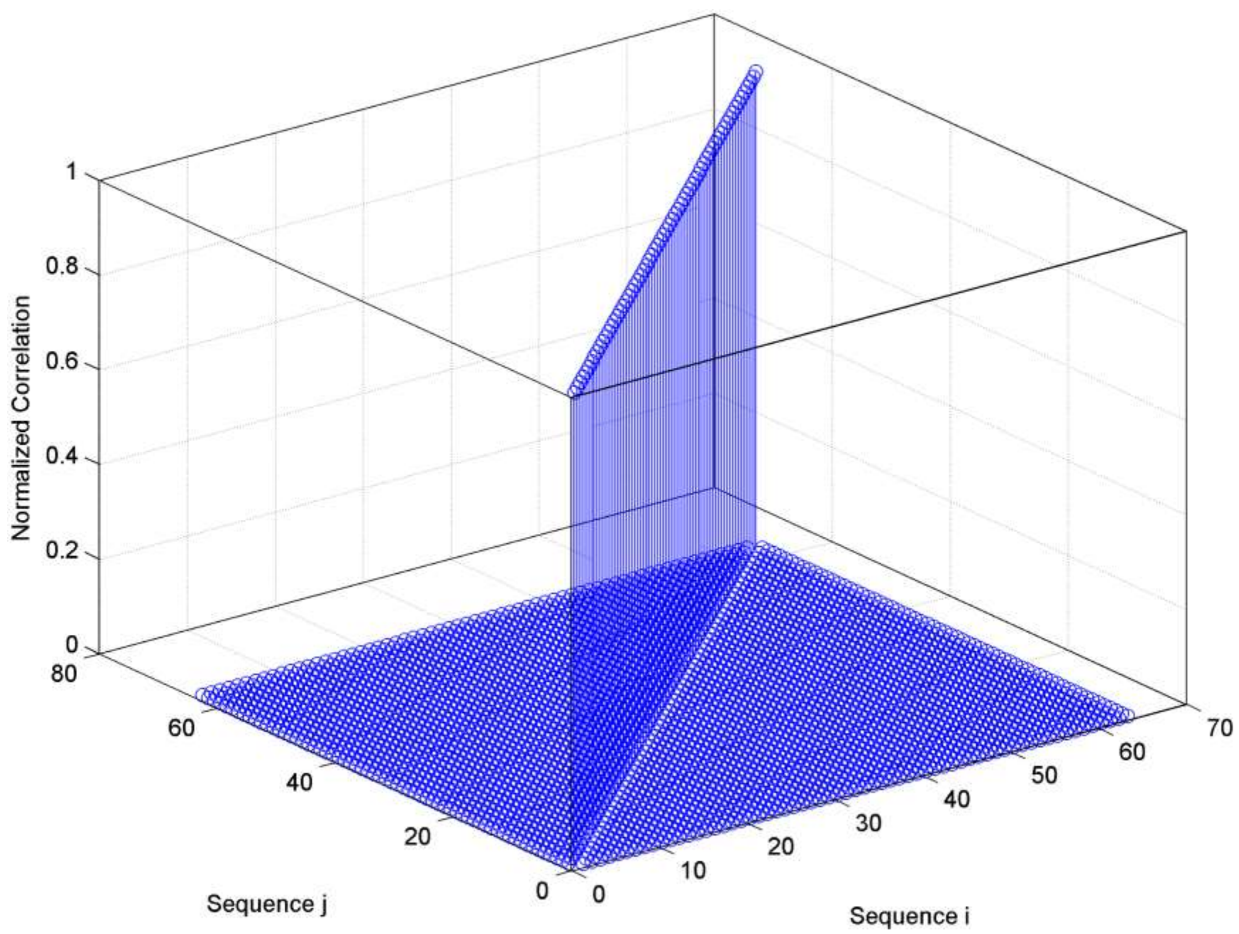

Figure 1. Correlation properties of super complementary codes (L = 64).

2. Super Orthogonal Complete Complementary codes

According to Chen [7], SOCC codes are perfectly orthogonal codes in the sense that they offer zero cross-correlation between any two codes for any relative shift in both synchronous and asynchronous transmission channels. A CDMA system can thus achieve multiple access interference free operation. In addition, SOCC codes can offer an ideal auto-correlation property such that their auto-correlation will be zero for all relative shifts except zero shift in both synchronous and asynchronous transmission modes. In Figure 1, the normalised correlation properties of SOCC codes of length L=64 are shown. The diagonal represents the auto-correlation of each sequence (sequence i and j, i=j) and the rest of the graph the cross-correlation between sequences (sequence i and j, $i \neq j$).

SOCC codes are based on complete complementary codes (CCC), which is based on the principle of using smaller element codes to make up a flock (or spreading code). The set size (number of flocks) of a CCC set is always equal to the flock size (number of element codes in one flock) and is $M = \sqrt{L_e}$. The processing gain (PG) of CCCs is equal to $L_e\sqrt{L_e}$, where L_e is the length of the element code. The length of the total CCC sequence is $L = L_e\sqrt{L_e}$. SOCC codes can support much more users in the same code set under a fixed PG value than CCC by making the element code size smaller (thus increasing the number of element codes, which is related to increasing the set size). Thus, a CCC code consisting of four flocks (four user codes), each consisting of four element codes of length L_e=16(PG = 16 × 4=64) can have an equivalent SOCC code with eight flocks (eight users codes) each consisting of eight element codes of length L_e=8, as well as other combinations, all with the same PG = 64. We will be using a SOCC code with 64 flocks (64 user codes), consisting of 64 element codes of length L_e=1. Each code can thus be uniquely allocated. The construction of CCC and SOCC codes are described in Chen [7] and will not be described here.

3. Proposed System Architecture

Figure 2 shows the GDSL transmitter [8], based on discrete multi-tone, with the proposed SOCC spreader highlighted. Figure 3 shows the corresponding GDSL receiver. With a subchannel spacing of 51.75 kHz (12 × 4.3125 kHz) and using a 4096 FFT structure, a total system bandwidth of 211.968 MHz is required. This bandwidth is suggested for future G.fast systems [4]. During typical xDSL modem initialisation, the background noise or quiet line noise is first measured. This provides a reference point for the background noise present in the line before any data (signal) are transmitted. The next step is for the central office (CO) transmitter to transmit equal power (−60 dBm/Hz) pilot tones in each subchannel, which is received and measured by the customer side receiver. Based on the received signal (plus noise) and the background noise, a signal to noise ratio (SNR) profile is obtained. The SNR profile is typically averaged over a couple of seconds (described in G.994.2 standard) to reduce the effect of additive white Gaussian noise. Any other static noise or interferer (like ADSL2+ services present in the binder) will reduce the SNR profile at the affected frequencies. Figure 4 shows a 3D plot of the SNR profile for the proposed system as a function of line length (0–0.3 km) and frequency (0 to 212 MHz) when no other interferers are present. According to the IEC EN55022:1998 [9] standard, a quasi-peak limit for radiation caused by an electric appliance is measured at a distance of 10 m. This limit of

30 dB(μV/m) is measured in any 6 dB band of 9 kHz width in the frequency range above 30 MHz. A transmit power spectral density below −60 dBm/Hz ensures that the 30 dB(μV/m) limit is on average not violated. A power spectral density of −60 and −140 dBm/Hz were used for the transmitter and background noise, respectively. This increases the power required from the usual 14.5 to 23.26 dBm (212 mW) for the 212 MHz bandwidth. Channel attenuation models for 0.5 mm copper lines were also used, based on two-port ABCD modelling [8]. FEXT and NEXT modelling is also described in [8].

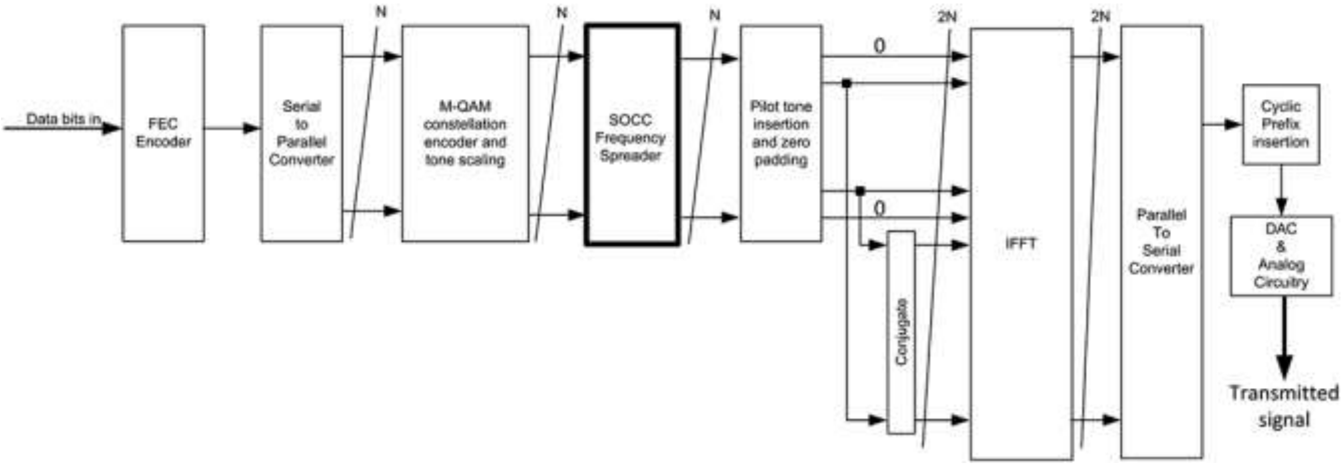

Figure 2. Gigabit digital subscriber line transmitter with proposed super orthogonal complete complementary (SOCC) spreader (highlighted).

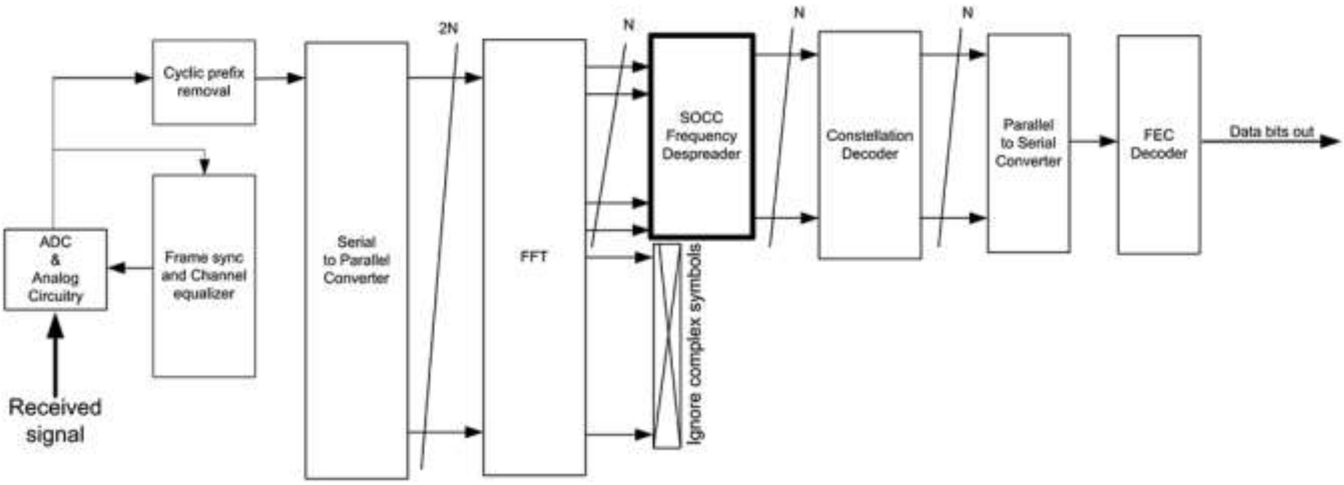

Figure 3. Gigabit digital subscriber line receiver with proposed SOCC Despreader (highlighted).

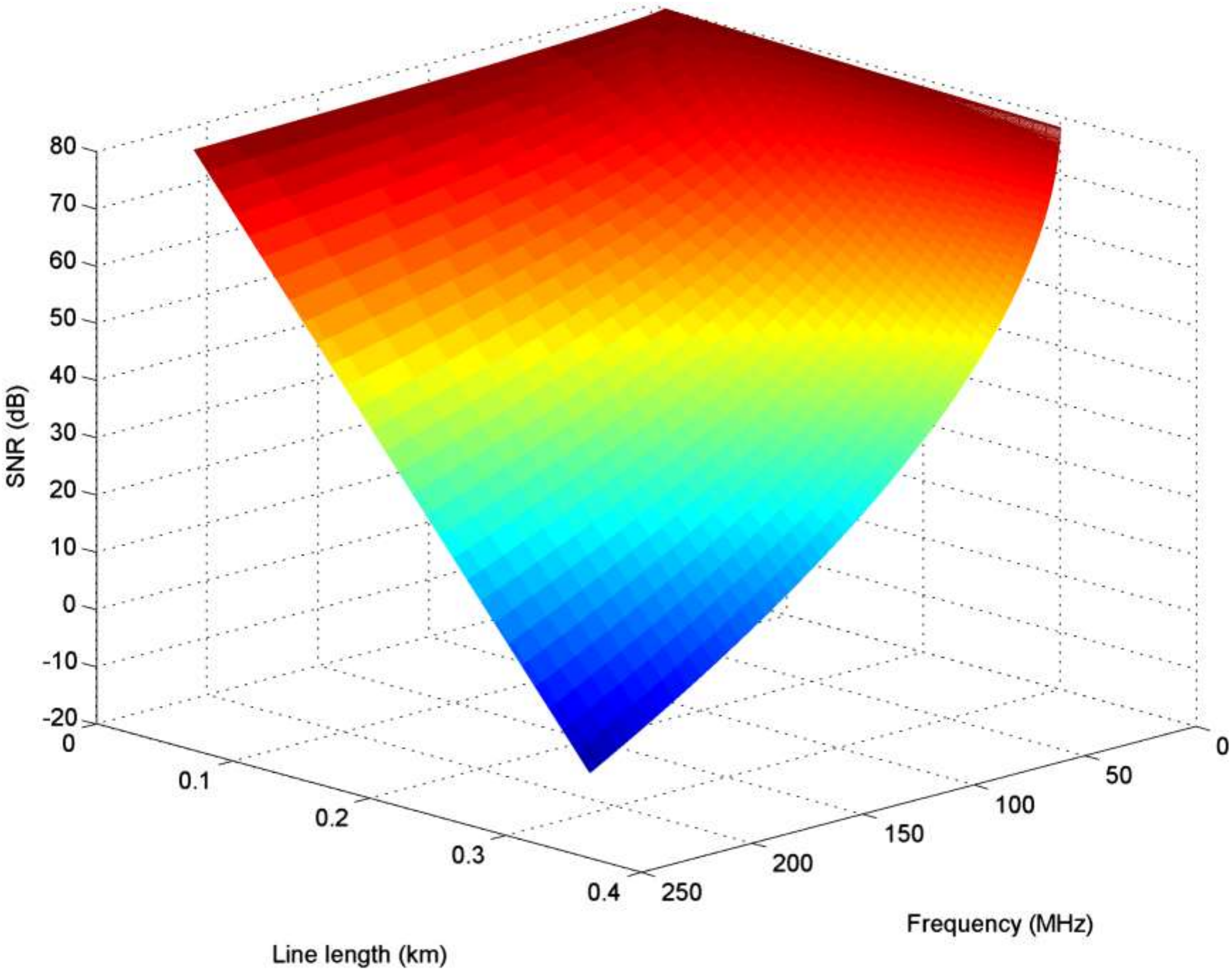

Figure 4. Signal-to-noise ratio profile versus line length and frequency.

A SNR margin (Γ) is also added to the measured SNR (thus lowering the effective SNR available) to provide a certain BER compared with an uncoded system. For typical QAM and PAM systems used, a SNR margin of 9.8 dB with a symbol error probability of 10^{-7} is used [8]. The allocated bits per subchannel (also referred to as 'the standard bit allocation' in this paper) is calculated as follows:

$$\bar{b} = \frac{1}{2}log_2\left(1 + \frac{SNR}{\Gamma}\right) \quad (1)$$

where $\bar{b}$ is the bits allocated per subchannel, *SNR* is the measured SNR and Γ is the SNR margin. Figure 5 shows a 3D plot of the allocated bits per subchannel for the proposed system as a function of line length and frequency. Considering a 300-m line, it is evident from this graph that a bandwidth of 100 MHz is sufficient, since the bits allocated above 100 MHz is zero. The example in Figure 6 below is a 2D slice at 200 m of Figure 5. In order to be able to use spreading over *L* subchannels, the bit allocation should be the same over these *L* subchannels. To illustrate this principle, consider the bit allocation for a 200-m line, using L=64 spreading, as shown in Figure 6. Using Equation (1), the standard bit allocation b (red circles) is obtained. The standard bit allocation is adapted using the following procedure to obtain the spreading bit allocation b_{spread} (blue crosses):

Algorithm 1

```
for k = 1 step L to FFT_size,
   temp = max of b(k to k+L-1);
   if temp > 15,
     temp = 15;
   end
   if temp < 2,
     temp = 0;
   end
   b_spread(k to k+L-1) = temp;
   while ∑_{n=k}^{k+L-1} b_spread(n)-0.5 >
   ∑_{n=k}^{k+L-1} b(n),
     b_spread(k:k+L-1) = b_spread
     (k:k+L-1) - 1;
   end
end
```

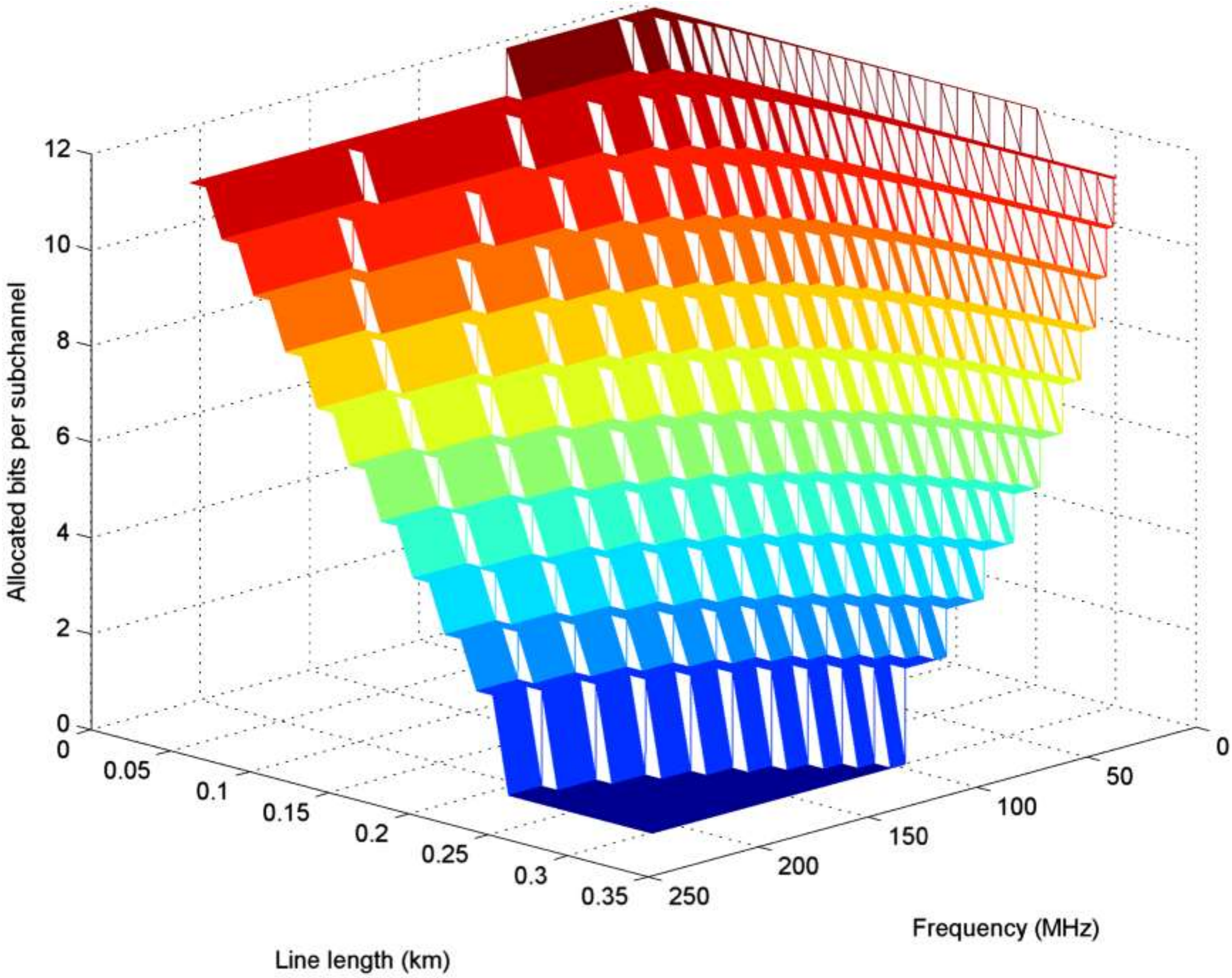

Figure 5. Theoretical bit allocation per subchannel versus line length and frequency.

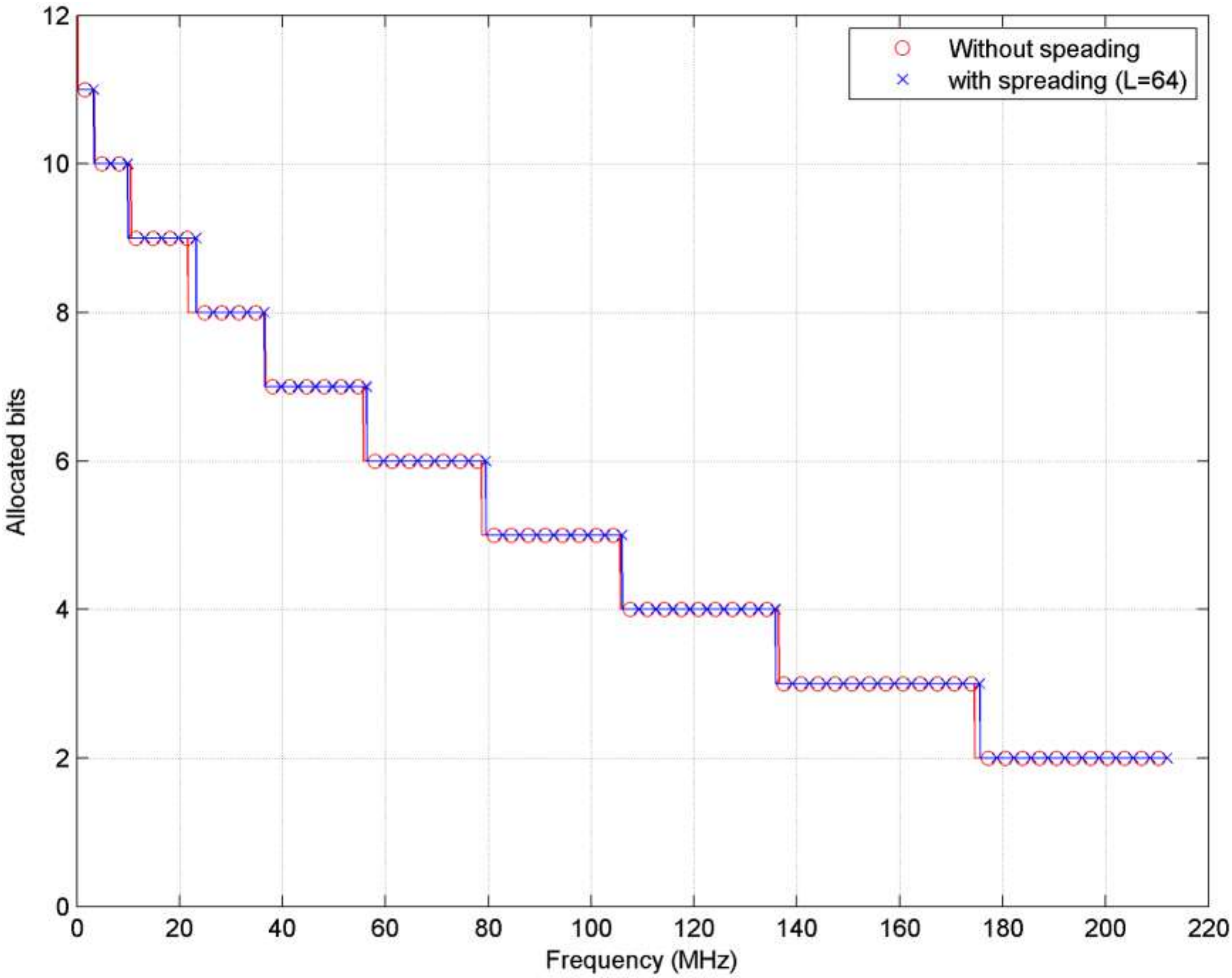

Figure 6. Bit allocation per subchannel for line length of 200 m without spreading (red circles) and with L = 64 spreading (blue crosses) versus frequency.

It should be observed that there is little difference between the two bit allocations. The difference becomes more pronounce for L=256. Figures 7 and 8 show a 3D plot of the spreading bit allocation per subchannel for L=64 and L=256, respectively, as a function of line length and frequency. Spreading bit allocation creates RBs by aggregating (joining) L subchannels, where L is the length of the spreading code used. Within each of these RBs, a pool of unique SOCC codes is available to be allocated to users. It is important to note that the spectral efficiency of the pool of SOCC codes is 1. When combined with the M-QAM modulation based on the bit allocation of each sub-band, higher spectral efficiencies are obtained. Figure 9 shows the throughput rate with a 10% total overhead assumed for error correction coding and timing overheads. From this graph, 1-Gbps aggregate throughput can be obtained over a 180-m piece of 0.5-mm copper wire.

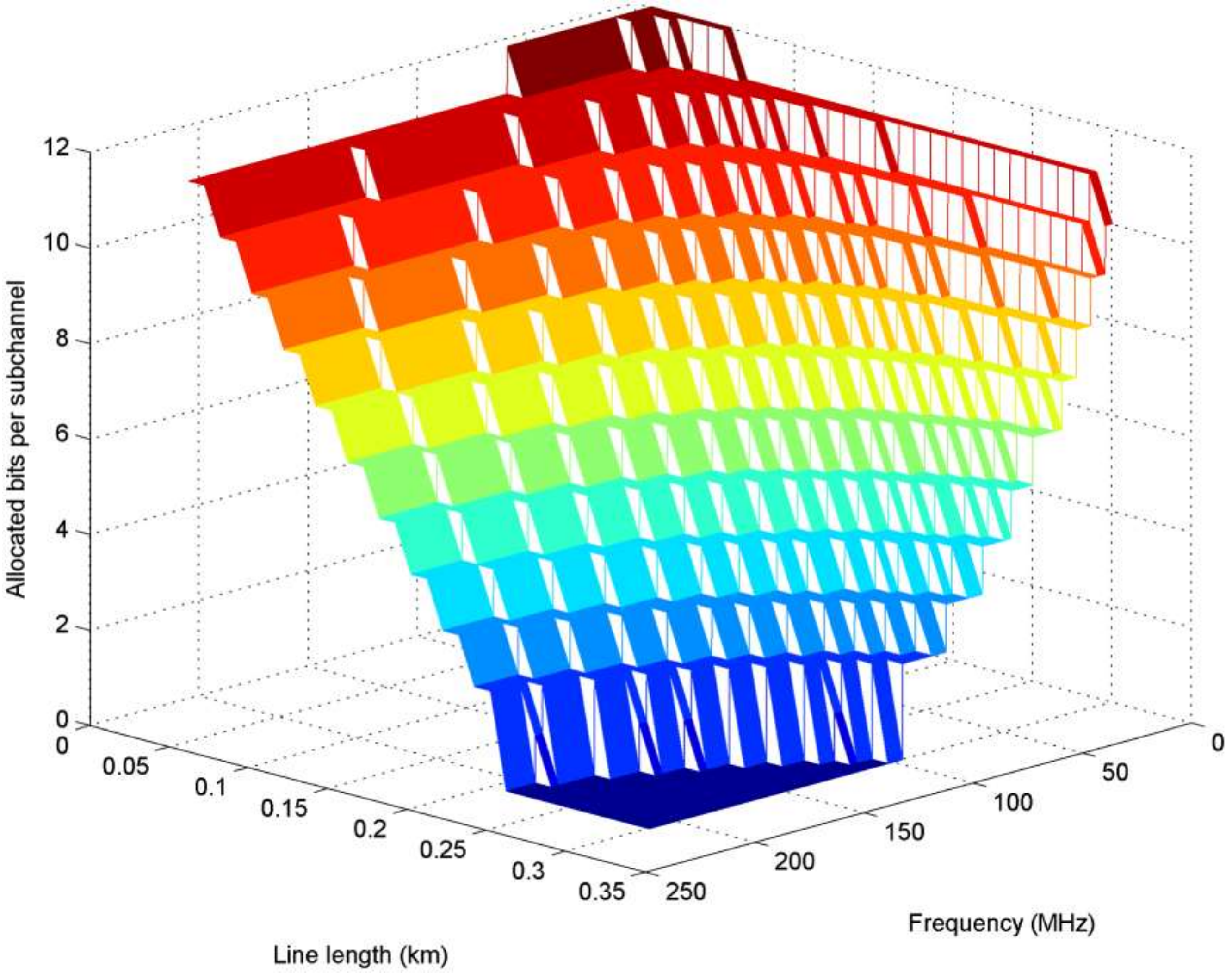

Figure 7. Bit allocation per subchannel with L = 64 spreading versus line length and frequency.

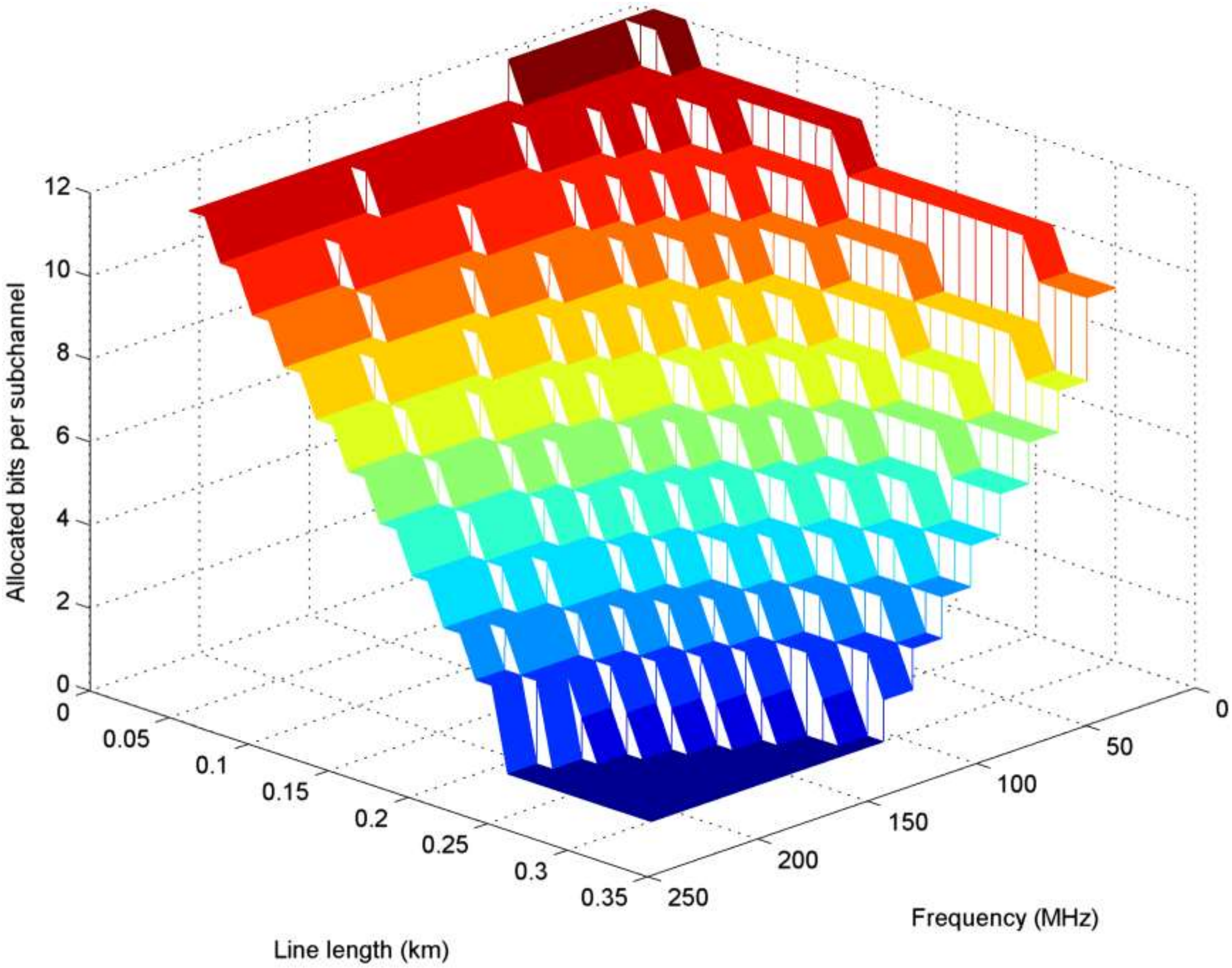

Figure 8. Bit allocation per subchannel with *L* = 256 spreading versus line length and frequency.

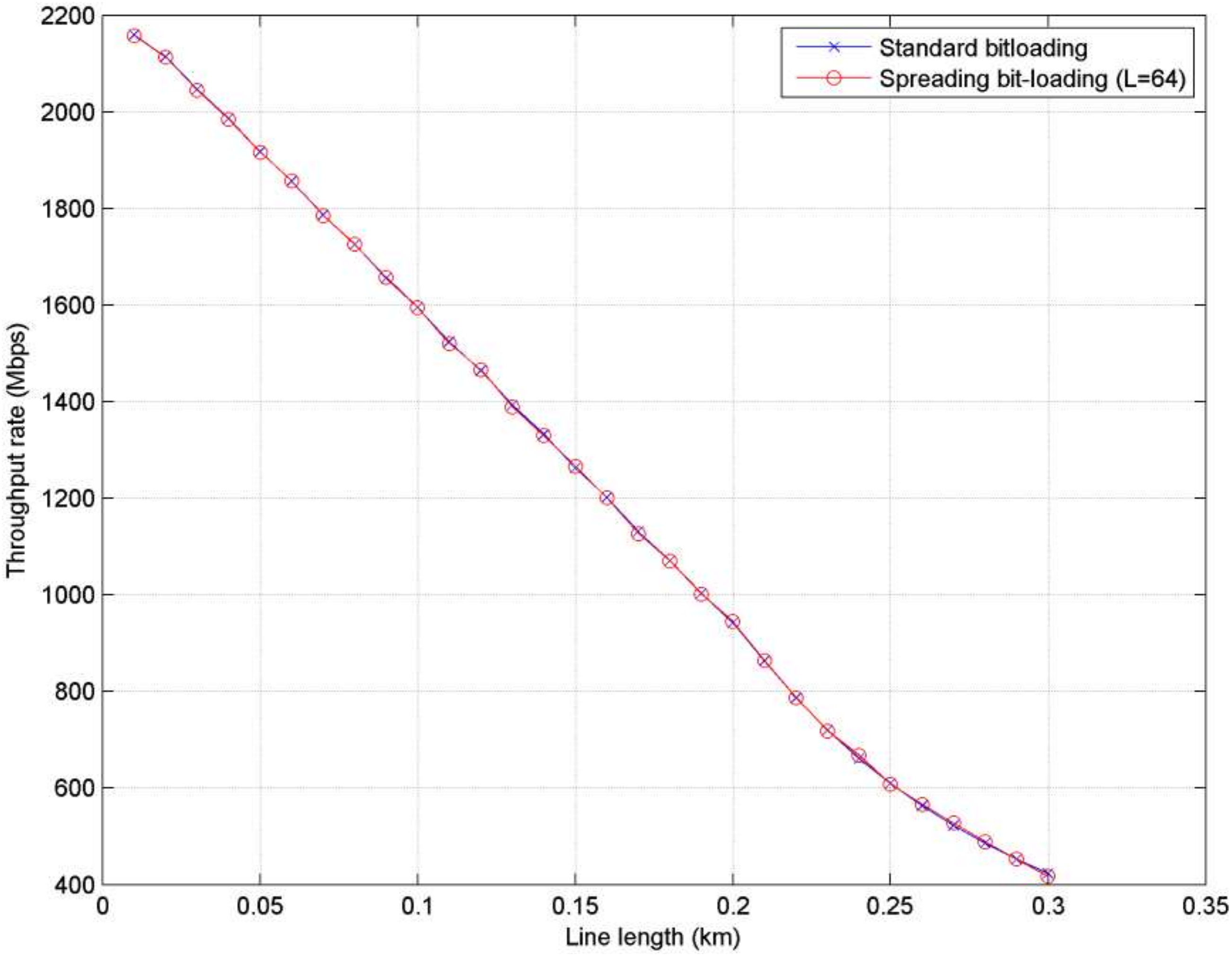

Figure 9. Throughput rate (Mbps) without spreading (red circles) and with L = 64 spreading (blue crosses) versus line length.

4. Resource block code allocation

In order to ensure MUI-free operation, codes are uniquely allocated to users in each RB. A novel algorithm was developed that uses the required rate (total data) of each user and the bits per RB profile to allocate codes such that each user at the end obtain the same ratio of actual rate (data) to required rate (data). In extreme cases, users with bad or long line profiles (having a bit allocation profile that quickly becomes zero for higher frequencies) can also have comparable good rate ratio performance, because more codes are allocated to them in lower frequency RBs. Users with better line profiles are allocated more codes in their higher-frequency RBs. Users will also only use spectrum where codes have been allocated. The algorithm works as follows:

Algorithm 2

Step 1: Specify the required rate (Mbps) of each user in the system as *DR_left*(i), for i = 1 to K (K the number of users)

Step 2: Obtain the bit allocation profiles based on RBs for each user as *BpRB*(i,m), for i = 1 to K and for m = 1 to M (M the total number of RBs)

Step 3: Iteratively calculate the codes allocated for each RB:

```
for m = 1 to M
 for i = 1 to K
   if BpRB(i,m) ≠ 0
    DR_used(i,m) = DR_left(i,m)
     - RB_DR(i,m-1)
    Determine ∑_{l=m+1}^{M} BpRB(i,l)
    Determine BpRB(i,m)_ratio from 5
    Codes_RB(i,m)
     =round(BpRB(i,m)_ratio.L)
    Determine RB_DR(i,m)
    if RB_DR(i,m) > DR_left(i,m)
      Codes_RB(i,m)=⌊RB_DR(i,m)/DR_left
     (i,m)+1⌋
      Determine RB_DR_adj(i,m)
```

Example:Consider five users, each with three RBs and a required rate as indicated in Table 1.

Table 1. Five users' profile (BpRB) and required data rate.

User	RB 1	RB 2	RB 3	Required data rate
1	8	6	2	8
2	12	12	11	20
3	12	12	11	10
4	6	5	3	8
5	3	2	0	8

User 1 has an average line profile 8, 6, 2 BpRB. Users 2 and 3 have good profiles but different required rates. Users 4 and 5 have the same required rate as User 1, but with worse line profiles. User 5's line profile is especially bad, and under existing DSL conditions, it will not nearly obtain its required data rate. We will show that the proposed algorithm adapts codes allocation to ensure that actual data rates for all users lead to the same ratio of actual rate to required rate.

The algorithm status for RB 1 is shown in Table 2. For RB 1, the data rate left is equal to the required rate. Because User 5 has the worst line profile with a high-required rate, it was assigned a high-BpRB weight (0.4534), leading to 29 codes being assigned to it. Users 2 and 4 have the same BpRB weight (0.1620) and corresponding 10 codes being assigned. User 2 however has a higher BpRB of 12, thus leading to a higher RB data rate. The algorithm status for RB 2 is shown in Table 3. For RB 2, User 5 is at its last RB where data can be allocated. The system is however not yet fully loaded; thus, it can allocate all remaining data of User 5 to RB 2. The algorithm status for RB 3 is shown in Table 4. For User 5, the BpRB is 0; thus, no codes are allocated. Codes are allocated as shown in column Codes but then adjusted based on the data rate left to the codes shown in column adj Codes. A summary of the data rates is shown in Table 5. All users are allocated a sufficient number of codes to allow the required data rate to be met. Table 6 shows a case of a fully loaded system (required data rate of all users adjusted to 20 Mbps). The actual rate of all users decreases so that the ratio of all users is on average 46%.

Table 2. RB 1 algorithm status.

User	Data rate left	BpRB left sum	BpRB weight	Codes	RB data rate	adj Codes	adj RB data rate
1	8	16	0.142	9	3.726	9	3.726
2	20	35	0.162	10	6.21	10	6.21
3	10	35	0.081	6	3.726	6	3.726
4	8	14	0.162	10	3.105	10	3.105
5	8	5	0.453	29	4.502	29	4.502

Table 3. RB 2 algorithm status.

User	Data rate left	BpRB left sum	BpRB weight	Codes	RB data rate	adj Codes	adj RB data rate
1	4.274	8	0.142	9	2.795	9	2.795
2	13.79	23	0.159	10	6.21	10	6.21
3	6.274	23	0.072	5	3.105	5	3.105
4	4.895	8	0.162	10	2.588	10	2.588
5	3.498	2	0.464	30	3.105	30	3.105

Table 4. RB 3 algorithm status.

User	Data rate left	BpRB left sum	BpRB weight	Codes	RB data rate	adj Codes	adj RB data rate
1	1.480	2	0.298	19	1.967	15	1.553
2	7.58	11	0.277	18	10.247	14	7.97
3	3.169	11	0.116	7	3.985	6	3.416
4	2.308	3	0.309	20	3.105	15	2.323
5	0	0	0	0	0	0	0

Table 5. Data rate comparisons.

User	Required data rate	RB 1 data rate	RB 2 data rate	RB 3 data rate	Total data rate	Ratio %
1	8	3.726	2.795	1.553	8.073	100.91
2	20	6.21	6.21	7.97	20.39	101.95
3	10	3.726	3.105	3.416	10.247	102.47
4	8	3.105	2.588	2.329	8.021	100.27
5	8	4.502	3.105	0	7.607	95.09

Table 6. Data rate comparisons for a fully loaded system.

User	Required data rate	RB 1 data rate	RB 2 data rate	RB 3 data rate	Total data rate	Ratio %
1	20	4.14	3.105	3.105	10.35	51.75
2	20	3.105	1.863	3.416	8.384	41.92
3	20	3.105	2.484	3.416	9	45.02
4	20	3.416	2.5875	3.416	9.42	47.09
5	20	5.123	3.83	0	8.953	44.76

5. Complexity of the proposed code allocation algorithm

In theoretical analysis of algorithms, it is common to estimate their complexity in the asymptotic sense, using for instance the Big-O notation. The complexity can also be based on a time analysis, but this will be dependent on the speed of the processor being used. Big-O is not dependent on processing speed. From Algorithm 2, Step 3, there are two for loops, one for the number of RBs (M) and one for the number of users (K). The complexity of the system is thus $O(MK)$, $0 < K,M \leq 64$. Looking more from an Engineering perspective, the number of multiply-accumulate operations can be used. From a more detailed analysis, the algorithm uses $15MK$ multiplications, $3MK + 4M(K\text{-}1)$ additions, $2MK$ rounding operations and $9MK$ decision (if) operations. The algorithm will only be executed at specific times (every 1, 10 or 100 ms), when the codes allocated to users are changed and updated.

Table 7. Comparison of xDSL service occupied bandwidth and GDSL starting subchannel and corresponding starting frequency.

xDSL service	Bandwidth used (MHz)	GDSL starting subchannel	Frequency (MHz)
ADSL	1.104	22	1.139
ADSL2+	2.208	45	2.277
VDSL2 Profile 8a/b/c	8.832	171	8.849
VDSL2 Profile 12a/b	12	232	12.006
VDSL2 Profile 17a	17.664	342	17.699
VDSL2 Profile 30a	30	581	30.007

GDSL, gigabit digital subscriber line.

6. Co-location with other xDSL services

The proposed system is designed to provide MUI-free and FEXT-free communication for all GDSL transceivers. If other services like ADSL / ADSL2+ / VDSL / VDSL2 and its variants are also present within the same binder as the GDSL services, the GDSL modem will not be able to remove their FEXT. The best work-around is to completely avoid the effected frequency areas and not allocate any bits. This is accomplished by changing the starting subchannel of the spreading bit allocation algorithm. Table 7 summarises the applicable xDSL standards, their occupied bandwidth, the GDSL starting subchannel and the starting frequency. ADSL2+ and VDSL2 profiles are obtained from ITU G.992.5[10] and G.993.2[11] standards, respectively.

Figure 10 shows how the spreading bit allocation is adjusted when a VDSL2 Profile 30a service is also present. The throughput rate is also affected, as shown in Figure 11. It should be observed that 1-Gbps aggregate throughput is obtained over 150 m of 0.5 mm wire.

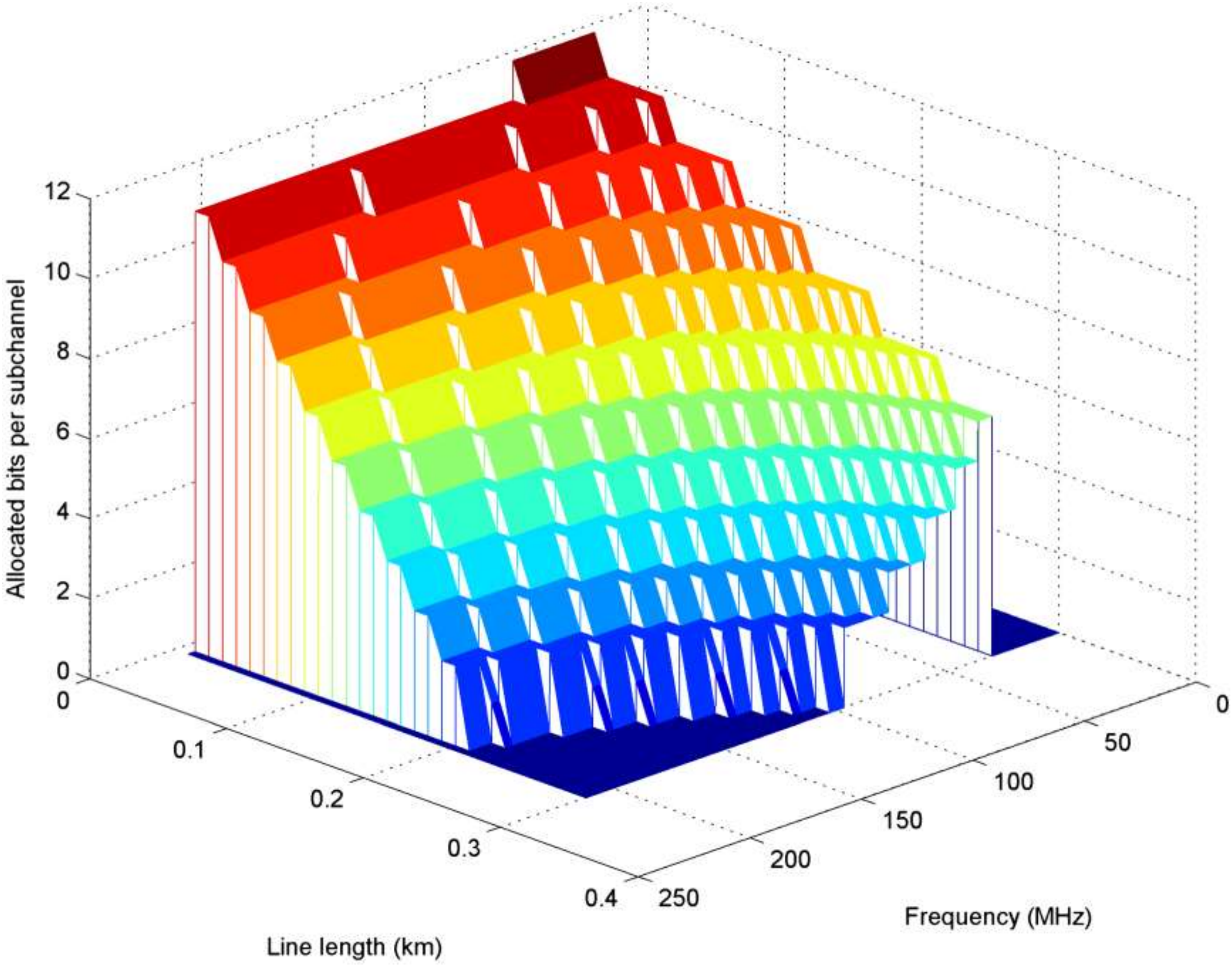

Figure 10. Bit allocation per subchannel with *L* = 64 spreading when VDSL2 Profile 30a services are present versus line length and frequency.

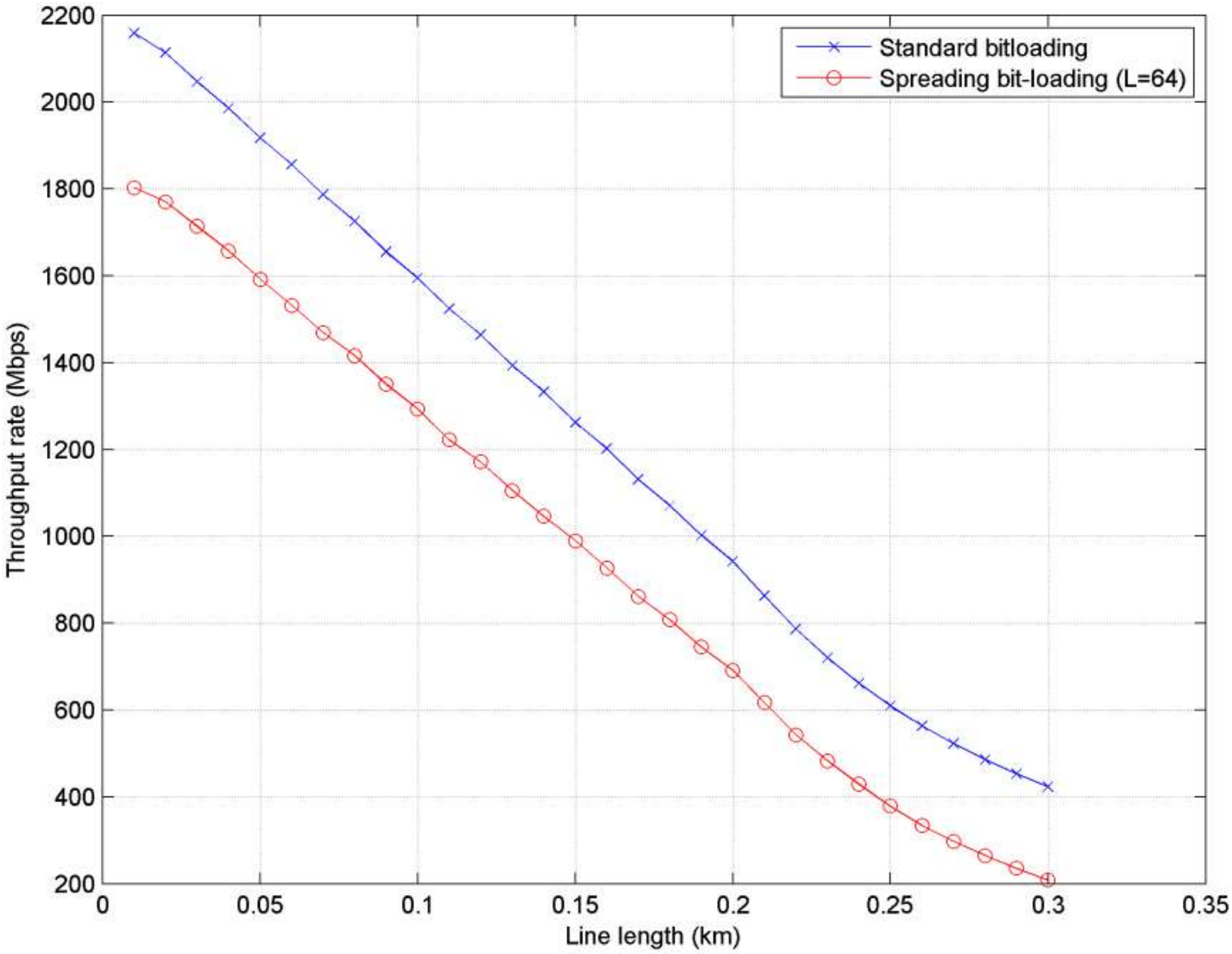

Figure 11. Throughput rate (Mbps) without spreading (red circles) and with *L* = 64 spreading (blue crosses) when VDSL2 Profile 30a services are present versus line length.

7. Conclusion

In this paper, a spreading block was introduced into existing DSL architecture to provide MUI-free communication even in a FEXT dominated environment. A novel code allocation algorithm was demonstrated, which provides all users with equal data rate ratios, even with bad line profiles and high-required data rates. When only GDSL users are present, 1-Gbps aggregate throughput can be obtained over a 180 m piece of 0.5 mm copper wire. In the presence of other xDSL services, distance is reduced to 150 m for 1 Gbps or about 800 Mbps for 180 m as a worse case scenario for VDSL2 Profile 30a.

Acknowledgements

This work is based on the research supported in part by the National Research Foundation of South Africa (Grant reference TP1207183332) and our industry partners Telkom and Bytes Universal Systems. The grantholder acknowledges that opinions, findings and conclusions or recommendations expressed in any publication generated by the NRF

supported research are that of the author(s), and that the NRF and our industry partners accept no liability whatsoever in this regard.

References

1. Leung C, Huberman S, Ho-Van K, Le-Ngoc T. Vectored DSL: potential, implementation issues and challenges. *IEEE Communications & Tutorials Q4* 2013; 15 (4): 1907–1923.
2. Forouzan AR, Moonen M, Timmers M, Guenach M, Maes J. On the achievable bit rates of DSL vectoring techniques in presence of alien crosstalkers. In *Proceedings of IEEE GLOBECOM,* Anaheim, California, 2012; 3086–3091.
3. *Very-high-bit-rate digital subscriber line 2*, 2015. https://en.wikipedia.org/wiki/Very-high-bit-rate_ digital_subscriber_line_2#Vectoring.
4. ITU-T. *Fast access to subscriber terminals (G.fast) –physical layer specification*, 2014. https://www.itu.int/rec/T-REC-G.9701/en.
5. Mallier S, Nouvel F, Baudais J, Gardan D, Zeddam A. Multi-carrier CDMA over copper lines - comparison of performances with the ADSL system. In *Proceedings of The First IEEE International Workshop on Electronic Design, Test and Applications*, Christchurch, New Zealand, 2002; 450–452.
6. Crussiere M, Baudais J, Helard JF. Improved throughput over wirelines with adaptive MC-DS-CDMA. In *Proceedings of IEEE Ninth International Symposium on Spread Spectrum Techniques and Applications*, Manaus, Amazon, Brazil, 2006; 143–147.
7. Chen HH. *The Next Generation CDMA Technologies.* John Wiley & Sons, Ltd.: Chichester, England, 2007.
8. Starr T, Cioffi JM, Silverman PJ. *Understanding Digital Subscriber Line Technology*. Prentice Hall: Upper Saddle River, NJ, USA, 1999.
9. IEC. Information technology equipment – radio disturbance characteristics limits methods of measurement. *Technical Report EN55022:1998 (CISPR 22:1997*), 1998.
10. ITU-T. *Asymmetric digital subscriber line 2 transceivers (ADSL2)- Extended bandwidth ADSL2 (ADSL2plus),* 2009. http://www.itu.int/rec/T-REC-G.992.5/e.
11. ITU-T. *Very high speed digital subscriber line transceivers 2 (VDSL2),* 2011. https://www.itu.int/rec/T-REC-G.993.2/en.