Near-Optimal Multi-user Greedy Bit-Loading for Digital Subscriber Lines

Alastair McKinley and Alan Marshall

Queens University Belfast, Belfast BT7 1NN, Northern Ireland
amckinley03@qub.ac.uk

Abstract. This work presents a new algorithm for Dynamic Spectrum Management (DSM) in Digital Subscriber Lines. Previous approaches have achieved high performance by attempting to directly solve or approximate the multiuser spectrum optimisation problem. These methods suffer from a high or intractable computational complexity for even a moderate number of DSL lines. A new method is proposed which is a heuristic extension of the single user greedy algorithm applied to the multi-user case. The new algorithm incorporates a novel cost function that penalises crosstalk as well as considering the *usefulness* of a tone. Previous work has proved the performance of the new algorithm in simple 2-user scenarios. In this work we present new results which demonstrate the performance of the algorithm in larger DSL bundles. Simulation results are presented and it is shown that the new method achieves results within a few percent of the optimal solution for these scenarios.

Keywords: Digital Subscriber Lines, Dynamic Spectrum Management, greedy bit-loading.

1 Introduction

Crosstalk is a major limiting factor in xDSL systems. In fact, crosstalk noise is the dominant noise source in DSL. Recently, new techniques which fall under the category of Dynamic Spectrum Management (DSM) have been proposed which seek to mitigate the effects of crosstalk. Although it is possible in theory to almost eliminate crosstalk [1], this is often impractical due to local loop unbundling, as signal level co-ordination is required between co-located modems. Where crosstalk cancellation is not possible, other techniques have focused on new bit-loading algorithms which are multiuser aware [2] [3] [4] [5] [6] [7] [8]. Rather than eliminate crosstalk, these techniques seek to reduce the effect of crosstalk, improving the performance of a binder group.

All of the aforementioned techniques require some form of centralised calculation, with varying computational demands depending on the particular algorithm. In particular the Optimal Spectrum Balancing algorithm (OSB) [2] and Iterative Spectrum Balancing [5] (ISB) require a significant amount of time to produce a solution for a large number of lines. The new algorithm presented here attempts to solve the spectrum balancing problem using a different approach to these methods.

C. Wang (Ed.): AccessNets, LNICST 6, pp. 224–239, 2009.

The rest of the paper is laid out as follows. Section 2 of this paper describes the system model used for simulation. Section 3 outlines the multi-user spectrum balancing problem and related work in this area. Section 4 gives a synopsis of the multi-user greedy algorithm [4], on which the method presented here is based. Section 5 presents the new algorithm "Near-Optimal Multi-User Greedy Bit-Loading" and details the enhancements made to the algorithm in [4] to arrive at the new method. Section 6 contains the new contribution of this work, demonstrating the performance of the new algorithm described here for a number of larger DSL bundles.

2 System Model

The system model adopted in this paper is an xDSL system based on DMT modulation, with N users and K tones per user. For simplicity, it is assumed that NEXT is eliminated by frequency division duplexing (FDD) and transmission is considered in the downstream direction only. Crosstalk noise is purely from FEXT coupling and occurs on a tone by tone basis.

The Signal-Noise Ratio for user n on tone k is computed as follows:

$$SNR_n(k) = \frac{p_n(k)|h_{nn}(k)|^2}{\sum\limits_{j \neq n} p_j(k)|h_{jn}(k)|^2 + \sigma_n^2(k)} \tag{1}$$

Where $p_n(k)$ is the transmit power spectral density (PSD) of user k on tone n. The term $|h_{nn}(k)|^2$ represents the direct channel power transfer function of the DSL line for user n on tone k. The term $|h_{jn}(k)|^2$ represents the crosstalk power transfer function (FEXT in this case) of user j into user n on tone k. The crosstalk gains are calculated according to standard models [9]. $\sigma_n^2(k)$ is the received background noise power for user n on tone k. It includes thermal noise plus background noise from other systems (e.g. ISDN, HDSL).

It is assumed that each modem has a maximum bit-loading of $b_{max} = 15$. It is further assumed that modems can only support integer bit-loading, which is the case in current implementations. The achievable bit-loading on tone k for user n is given by

$$b_n(k) = \log_2\left(1 + \frac{SNR_n(k)}{\Gamma}\right) \tag{2}$$

where Γ is the 'SNR-gap' which is a function of the line code and target BER [10].

3 The Spectrum Balancing Problem

The spectrum balancing problem is expressed in a number of different ways, depending on rate and power constraints. They are generally categorised into Rate Adaptive (RA) and Fixed Margin (FM) methods [10]. Rate adaptive algorithms attempt to maximise user's bit rates under power budget constraints.

Fixed Margin algorithms minimise the power required to transmit a target bit rate at a fixed performance margin.

In a multi-user channel, the spectrum balancing problem can be stated as follows:

$$\begin{aligned} max\ R_1\ s.t.\ R_n &\geq R_n^{target} \\ s.t. \sum_{k=1}^{K} p_n(k) &\leq P_n^{budget} \end{aligned} \tag{3}$$

Equation (3) states that the rate of line 1 is to be maximised, subject to all other lines meeting or exceeding their rate targets and that all lines are within their respective power budgets.

3.1 Related Work

Many DSM algorithms have been proposed which address the multi-user spectrum balancing problem. Iterative Waterfilling (IW) [11] is a DSM technique which is distributed, whereby each modem calculates its own power spectral densities. Optimal Spectrum Balancing (OSB) [2] is a centralised DSM algorithm, where the power spectral densities are calculated centrally by a Spectrum Management Centre (SMC). In OSB, the spectrum balancing problem is reformulated using a dual-decomposition, transforming the power constrained optimisation problem into an unconstrained optimisation of the langrangian sub-problem. OSB provides better performance than IW at the cost of increased co-ordination and a higher computational complexity. The inner loop of the algorithm is an exhaustive search over all bit loading combinations on a particular tone. As such, the running time of OSB is exponential in the number of users N. Practically this means that OSB is intractable for more than 5-6 DSL lines. Iterative Spectrum Balancing (ISB) [5] [12] is a sub-optimal approach based on OSB, which reduces the computational complexity of the inner loop of ISB to $O(N^2)$ at the cost of the optimality of the solution. The exhaustive search of the inner loop is replaced by an iterative search on each individual line which converges to at least a local optimum solution, and often to the global optimum for a tone. The SCALE [3] algorithm takes a different approach, attempting to maximise the per tone lagrangian through an iterative convex relaxation procedure. SCALE has been shown to achieve close to optimal performance in networks with a moderate number of DSL lines and has a complexity similar to ISB [13].

Band-Preference techniques [6] [7] are a relatively new idea which follow from the results of SCALE. The main idea in Band-Preference is that each modem executes a “scaled water-filling” algorithm on a preset number of bands within its own spectrum. An SMC calculates the scaling factors for each band on each modem and communicates these results to the individual modems. Increasing the number of bands increases the performance of the algorithm. With unity band sizes, the result converges to the optimum solution [7]. An advantage of Band-Preference Algorithms is the possibility of using hooks in currently available standards (ADSL2,VDSL) to implement them.

4 Multi-User Greedy Bit Loading

The core of the new algorithm presented here is based upon the multiuser greedy algorithm [4], and is a heuristic extension of the single user greedy loading algorithm [14]. The single user greedy algorithm is a conceptually simple algorithm, where each line loads one bit on to the tone on which it will cost the least power to do so, until the line's power budget is fully utilised. In effect this is the discrete version of the classic waterfilling algorithm. This idea is extended to its multi-user counterpart in [4]. The entire binder group is considered as a whole and bits are added incrementally to the channel and line with the lowest energy cost. The cost metric is defined as the power required to add one bit to a particular line/tone and the sum of the power increases required on all other lines on that tone in order to bring those lines back within the performance margin given the increased crosstalk. We will denote this term Δp. The objective of the multi-user greedy algorithm is to minimise the total power required to transmit a total rate-sum. This algorithm does not consider fairness between users, or guarantee a minimum data rate for each user.

In order to calculate the cost metric, the power vector to support a particular bit vector must be calculated. The achievable bit-loading on line n tone k is as follows:

$$b_n(k) = \log_2\left(1 + \frac{p_n(k)|h_{nn}(k)|^2}{\Gamma\left(\sigma_n^2(k) + \sum_{j\neq n} p_j(k)|h_{jn}(k)|^2\right)}\right) \tag{4}$$

Re-arranging the formula (4) letting $f(b_n(k)) = \Gamma(2^{b_n(k)} - 1)$ we obtain:

$$p_n(k) - f(b_n(k))\sum_{j\neq n} p_j(k)\frac{|h_{jn}(k)|^2}{|h_{nn}(k)|^2} = f(b_n(k))\frac{\sigma_n^2(k)}{|h_{nn}(k)|^2} \tag{5}$$

For a particular tone k, equation (5) is an N-dimensional linear system of equations, where N is the number of lines. This can be written in matrix form:

$$A(k)P(k) = B(k) \tag{6}$$

where

$$A(k)_{ij} = \begin{cases} 1, & \text{for } i = j \\ \frac{-f(b_i(k))|h_{ji}|^2}{|h_{ii}|^2}, & \text{for } i \neq j \end{cases} \tag{7}$$

$$P(k) = [p_1(k)\ldots p_i(k)\ldots p_N(k)]^T \tag{8}$$

$$B(k) = \left[\frac{f(b_1(k))\sigma_1^2}{|h_{11}|^2}\ldots\frac{f(b_i(k))\sigma_i^2}{|h_{ii}|^2}\ldots\frac{f(b_N(k))\sigma_N^2}{|h_{NN}|^2}\right]^T \tag{9}$$

This can be solved for $P(k)$ via direct inversion or by LU/QR decomposition. $P(k)$ is a vector which contains the power value required on each line to support

Algorithm 1. The Original Multi-User Greedy Loading Algorithm

function *main*
    $init()$
    repeat
        $n, k = \text{argmin } C[n][k]$ ▷ user n, tone k
        increment $b_n(k)$
        if $P_n > P_n^{budget} \;\; \forall n$ **then**
            decrement $b_n(k)$
            $F[n][k] = 1$ ▷ Set tone as full
        end if
        if $b_n(k) == maxbits$ **then**
            $F[n][k] = 1$ ▷ Set tone as full
        end if
        if $R_n == R_n^{target}$ **then**
            $F[n][\forall k] = 1$ ▷ Set user n tones full
        end if
        $C[n][k] = cost\_function(n, k)$ ▷ $\forall n, k$
    until all tones full
end function

function *init*
    for all n, k **do**
        $F[n][k] = 0$ ▷ Matrix of flags set to 0
        $C[n][k] = cost\_function(n, k)$
    end for
end function

function $cost\_function(n, k)$
    $b(\hat{k}) = b(k) + e(n)$
    $P(\hat{k}) = A(k)^{-1} B(\hat{k})$
    $cost = \sum_{n=1}^{N} \left(p_n(\hat{k}) - p_n(k) \right)$
    return *cost*
end function

a vector of bits $b(k)$ on tone k. (Note $B(k)$ is a function of $b(k)$). Also, the solution can only be computed in this way if it is assumed that each tone is independent, i.e. no adjacent channel interference.

Knowing the structure of equation (6), we can formulate the multi-user greedy algorithm shown in algorithm 1. In algorithm 1, F is an $N \times K$ matrix of flags which is initialised to 0 and set to 1 when a tone on a particular user is declared full. Also, C is an $N \times K$ matrix containing the cost to add one bit to a tone k on user n. The vector $e(n)$ is a zero vector of length N with element n set to 1. As previously discussed, the cost is the sum of the total extra power required on all lines when a bit is added to user n tone k.

4.1 Performance

The original multi-user greedy algorithm does not consider fairness between users or guarantee a minimum data rate for any user. Due to this fact, it performs poorly in some scenarios, such as the near-far scenario shown in figure 1. In this network configuration, there is a large amount of crosstalk from the "strong" line, i.e. the shorter RT-fed line into the "weak" line, i.e. the longer CO-fed line. Crosstalk from the RT line dominates the noise spectrum of the CO line, greatly reducing its potential performance. Crosstalk avoidance through some DSM technique can vastly improve performance in this scenario. Figure 2 compares the downstream power spectral density for the two line scenario shown in figure 1 using the multi-user greedy algorithm and the optimal spectrum balancing algorithm [2]. The RT line rate target is set to 4Mbps. In this scenario, the multi-user greedy algorithm achieves a rate of 1.084Mbps on the CO line. In contrast, the OSB algorithm achieves a rate of 3.484Mbps on the CO line. It is clear that the multi-user greedy algorithm does not perform well in this scenario. The following section will outline some changes to the multi-user greedy algorithm which dramatically improves its performance.

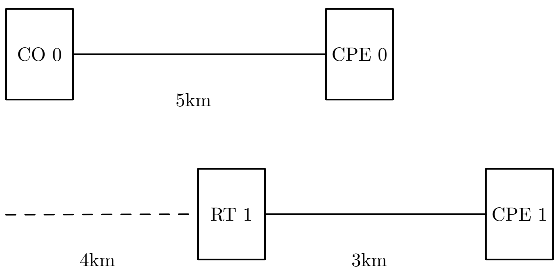

Fig. 1. A two user DSL network which exhibits the near-far effect

5 Near-Optimal Multi-user Greedy Bit-Loading

A simple observation from figure 2 is that in the optimal solution, the RT line only uses its higher frequency tones. This is because, to achieve a higher bit-rate on the CO line, the RT line will yield to the CO line at low frequencies, as the RT line is short enough that it can use its higher frequencies for data transmission. In the case of the multi-user greedy algorithm, we note that the cost function is based on the lowest incremental power to add a bit to a particular tone. When the cost matrix is first initialised, each tone contains zero bits. Therefore, when a bit is added to a tone, it will incur no extra cost due to generated crosstalk. This means that the highest gain channels will always be chosen first, which in the case of figure 1 will be the lowest frequency tones on the RT line. In doing so, the multi-user greedy algorithm will immediately choose the wrong tones for loading as compared to the optimal solution. It is now postulated that a new cost function which accounts for crosstalk effects in a different way may achieve better results.

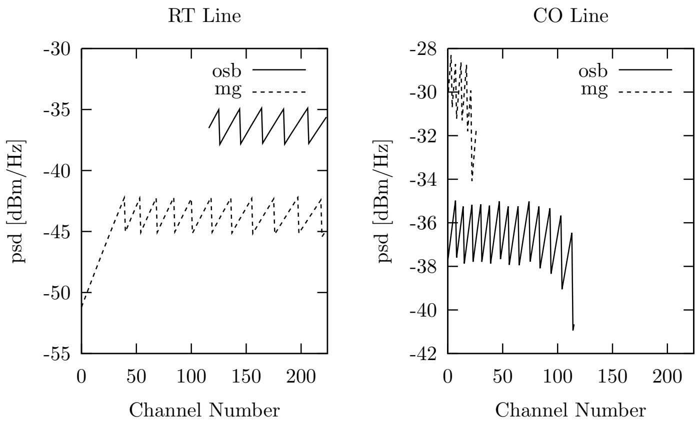

Fig. 2. PSD of RT line at CO line with OSB and multi-user greedy algorithms

5.1 A New Cost Function

The new cost function is fundamentally based on the same cost function as the original multi-user greedy algorithm, i.e. the incremental power cost to add one bit to a particular line and tone Δp. This is because this metric has some inherently good properties. Firstly, it generally chooses the tones with the highest gains and, secondly, it also chooses tones that do not cause large power increases on other lines due to crosstalk.

In addition to Δp, two new terms are included. The first is a crosstalk penalty term denoted by $w_n\beta$. The second is a factor $\gamma_n(k)$ which measures the relative "usefullness" of a tone relative to all other tones on that line. This is used to adjust the crosstalk penalty depending on how important that tone is on a particular line. The $w_n\beta$ term is an "inter-line" cost adjustment and the $\gamma_n(k)$ is an "intra-line" cost adjustment. The penalty term is summed over all victim tones (a victim tone is a tone which incurs crosstalk due to loading bits on line n) when calculating the cost for a tone.

Crosstalk Penalty Term. The crosstalk penalty term is calculated in the following manner. When calculating the cost for a particular tone k on line n, use the current bit value $b_n(k)$ and assume that the other lines on tone k are set to a reference PSD. In the simulations shown here -40 dBm/Hz was used. Given these initial parameters, calculate the vector of bits on tone k, b_k. Now increment the value of $b_n(k)$ and recalculate the vector b_k. From the two vectors calculated, a penalty term for each victim tone can be constructed as follows:

$$penalty = w_n \times \overbrace{(\underset{b_k(k)}{b_k(j))} - \underset{b_n(k)+1}{b_k(j)}\ )}^{\beta=\text{``lost bits''}} \qquad \text{for } \ j = 1 \ldots N \ \ j \neq n \tag{10}$$

The term w_n in equation (10) is a weight which determines how much a crosstalking line is penalised due to its own generated crosstalk. Adjusting this value will determine the bit rates on each line. The second term in equation (10) is marked "lost bits", as it effectively is the number of bits lost on line j tone k after adding an extra bit on line n tone k. In this way a major pitfall of the original multi-user greedy algorithm is avoided, as crosstalking lines are penalised even if victim lines do not yet contain any bits.

Relative Usefullness Adjustment. The crosstalk penalty term is further adjusted by a factor which determines how useful a tone k on line j is relative to the rest of its tones. The rationale for this is that intuitively, if a line is crosstalking heavily into another line's strongest tones, it should be more heavily penalised for adding extra bits. Conversely, if a crosstalking line is heavily crosstalking into a "weak" line's worst tones, then the penalty should be less as the "weak" line will not use these tones in any case. To calculate $\gamma_n(k)$ for each line n and tone k we first assume zero FEXT into the line and calculate the total bits attainable on each tone k given a reference PSD (-40dBm/Hz) and also the average number of bits attainable on line n. The total number of bits attainable of line n tone k is given by $b_n(k)^{ref}$ and the average bits on line n by $\bar{b_n}$. This adjustment is calculated for each line n and tone k as follows:

$$\gamma_n(k) = \frac{b_n(k)^{ref}}{\bar{b_n}} \qquad \text{for} \quad k = 1 \ldots K, n = 1 \ldots N \tag{11}$$

Cost Matrix Re-Initialisation. Another adjustment made to the multi-user greedy algorithm is the re-initialisation of the cost matrix C during loading. When a line n reaches its rate target R_n^{target} or its power budget P_n^{budget}, there is no longer any need to penalise a neighbouring line for crosstalking into that line. A $user\_frozen$ flag is set for each line that has finished loading bits. This is used in the cost function calculation to determine whether to penalise a crosstalking line or not. Although additional crosstalk into a line which has reached its power budget may cause the line to break its power constraint, this will be caught in the main loop of the algorithm, causing its flag in the matrix F to be set to 1. This re-initialisation ensures that no lines are unfairly penalised for crosstalking into lines that either cannot load any more bits due to reaching their power budget, or will not load any more bits because they have reached their rate target.

The revised cost function is shown in algorithm 2.

5.2 Algorithm Complexity

Although the algorithm here is slightly more complex than the multi-user greedy algorithm, tests showed that results for 50 line bundles could be achieved in a reasonable time (on the order of minutes) on standard PC hardware running non-optimised code. For the 8 line simulations presented in the results section, the running time was on the order of seconds. It was noted that most of the execution

Algorithm 2. Revised Cost Function

function $cost\_function(n, k)$
 $\hat{b(k)} = b(k) + e(n)$
 $\hat{P(k)} = A(k)^{-1}\hat{B(k)}$
 $\Delta p = \sum_{n=1}^{N} \left(\hat{p_n(k)} - p_n(k)\right)$
 $p_m(k) = -40dBm/Hz \text{ for } m = 1 \ldots N, m \neq n$
 Calculate $\underset{p(k),b_n(k)}{b(k)}$
 Calculate $\underset{p(k),b_n(k)+1}{b(k)}$
 for $m = 1 \ldots M, m \neq n$ **do**
  if $(!user\_frozen[m])$ **then**
   $\beta_m = \underset{p(k),b_n(k)}{b_m(k)} - \underset{p(k),b_n(k)+1}{b_m(k)}$
   $penalty+ = w_n \beta_m \gamma_m(k)$
  end if
 end for
 $cost = \Delta p \times penalty$
 return $cost$
end function

time of the algorithm was taken up by the PSD vector calculation of equation (6). As this step requires the inversion of a matrix, it requires $O(N^3)$ operations. A significant optimisation of this step is outlined in [4]. The cost function requires an evaluation of $P(k)$ given $b(k)$ and subsequently an evaluation of $P(k)$ given $b(k) + e(n)$. As the addition of $e(n)$ only makes a small rank update to $A(k)$, the calculation of successive inverses of $A(k)$ can be optimised using the matrix inversion lemma[1]. This reduces the complexity of calculating successive inverses of $A(k)$ to $O(N^2)$ operations. This optimisation was not implemented in the simulations here and it is expected that it would produce a significant decrease in execution time. There is also another possible optimisation presented in [15] where multiple bits are added at once to reduce the need for cost matrix updates without performance degradation. This optimisation was not used in the simulations presented here, but should also bring a significant performance increase.

6 Simulation Results

In this section the new algorithm is tested in two different scenarios by simulation. All DSL line diameters are assumed to be 0.5mm (24-AWG). The coding gain is set at 3dB and the noise margin at 6dB. The gap Γ is set to 9.8dB in all scenarios and a power budget of 110mW is assumed on each line. All lines are using DMT ADSL and transmission is in the downstream direction. The crosstalk damping terms w_n are chosen to meet the rate targets for each scenario.

[1] $(A + bc^T)^{-1} = A^{-1} - (\frac{1}{1+c^T A^{-1} b})(A^{-1}b)(c^T A^{-1})$.

Previous work [16] considered the performance of the new algorithm in 2-user network scenarios. The results presented here expand on previous results and in particular demonstrate the suitability of the new algorithm to larger bundles.

6.1 2-User Near Far Scenario

The first scenario is the near far scenario shown in figure 1. For this scenario, only self-FEXT is considered. The rate target for the RT line was set at 4.1Mbps. Figure 3 shows the resulting PSD graph against channel number beside the optimal solution obtained by the OSB algorithm. The corresponding bit-rates for this scenario are shown in table 1. In this case, the new algorithm actually achieves the same result as the optimal solution given by the OSB algorithm.

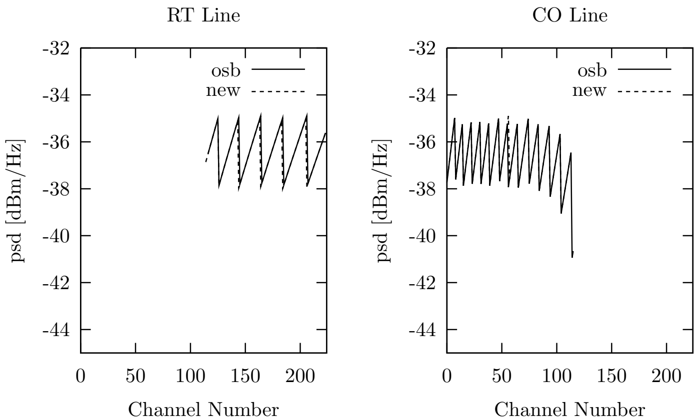

Fig. 3. PSD of RT line and CO line with OSB and new algorithm for scenario in figure 1

Table 1. Bit rates for OSB, Multi-User Greedy and new algorithm for the scenario in figure 1

	OSB	New Algorithm	Multi-User Greedy
CO Line	3.484Mbps	3.484Mbps	1.052Mbps
RT Line	4.1Mbps	4.1Mbps	4.1Mbps

6.2 8-User Central Office Scenario

In this scenario, eight DSL users are connected to a DSLAM in the Central Office. Four of the eight users lines are 3 kilometres in length, whilst the other four are 2 kilometres in length. The scenario is illustrated in figure 4. For this scenario, alien crosstalk is included as given by model A in [17]. The Iterative Spectrum Balancing algorithm (ISB) is used as a comparison to the new algorithm presented here, as OSB is intractable for 8 lines. Iterative Spectrum balancing has be shown to achieve near optimal results with a significant reduction in complexity relative to OSB. OSB with branch and bound [18] is tractable for an 8 line scenario, however due to time constraints it was not implemented here.

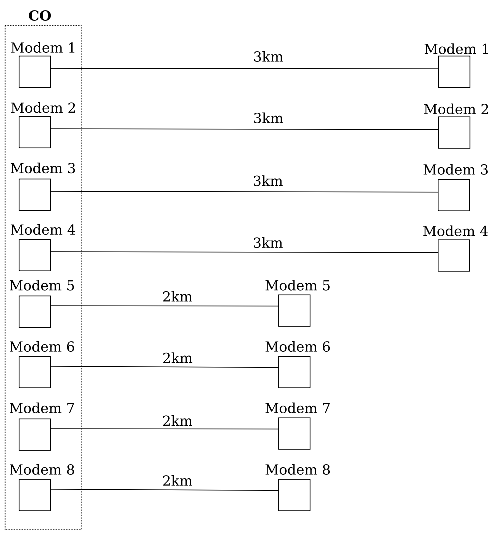

Fig. 4. An 8-User Scenario with all modems co-located at the Central Office

Table 2. Data rates for scenario in figure 4 comparing ISB to the new algorithm

	ISB	New Algorithm
User 1	2.912 Mbps	2.892 Mbps
User 2	2.932 Mbps	2.9 Mbps
User 3	2.912 Mbps	2.896 Mbps
User 4	2.904 Mbps	2.896 Mbps
User 5	4.172 Mbps	4.164 Mbps
User 6	4.172 Mbps	4.164 Mbps
User 7	4.148 Mbps	4.164 Mbps
User 8	4.168 Mbps	4.164 Mbps

The results are shown in table 2. The performance of the new algorithm is shown to to approach the performance of the ISB algorithm very closely. The average drop in data rate for each user is just 0.34% in comparison to ISB. In figure 5 the resulting PSDs are shown for one of the 3km lines and one of the 2km lines for both ISB the new greedy algorithm.

6.3 8-User Mixed Deployment Scenario

This scenario consists of an eight user network, with two remote terminals located 500 metres and 3 kilometres from the central office respectively. This is illustrated in figure 6. Once again, the ISB algorithm is used for comparison and alien crosstalk is included as in the eight user central office scenario. The results are shown in table 3. It can be seen that the results given by the new algorithm closely approach the performance of ISB. In this case, the average drop in data

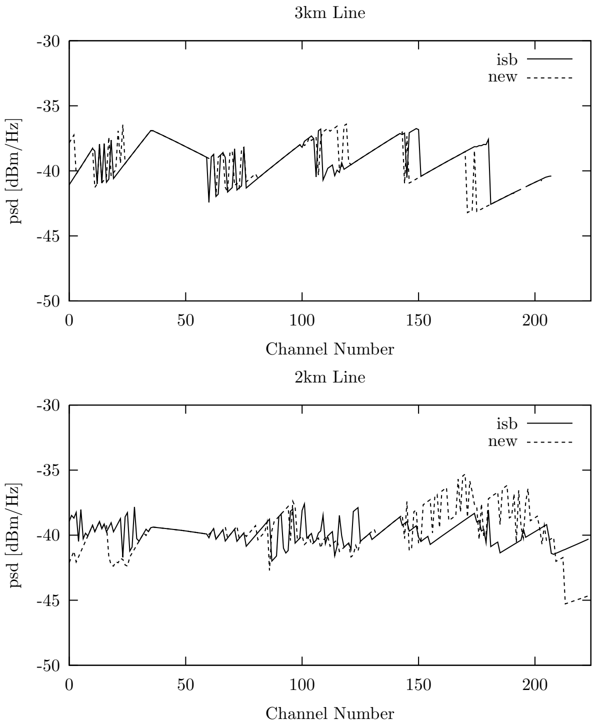

Fig. 5. Resulting Power Spectral Densities for ISB and the new algorithm for a 3km line and 2km line in scenario 4

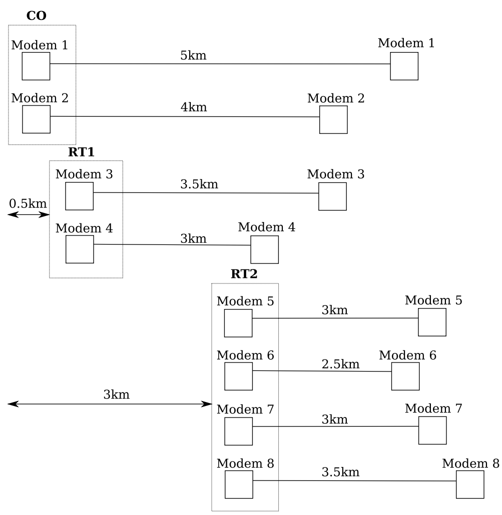

Fig. 6. An 8-User Scenario with two remote terminals

Table 3. Data rates for mixed deployment scenario in figure 6 comparing ISB to the new algorithm

	ISB	New Algorithm
User 1	0.240 Mbps	0.238 Mbps
User 2	0.772 Mbps	0.772 Mbps
User 3	1.528 Mbps	1.520 Mbps
User 4	2.600 Mbps	2.600 Mbps
User 5	0.744 Mbps	0.744 Mbps
User 6	1.064 Mbps	1.060 Mbps
User 7	0.744 Mbps	0.744 Mbps
User 8	0.528 Mbps	0.524 Mbps

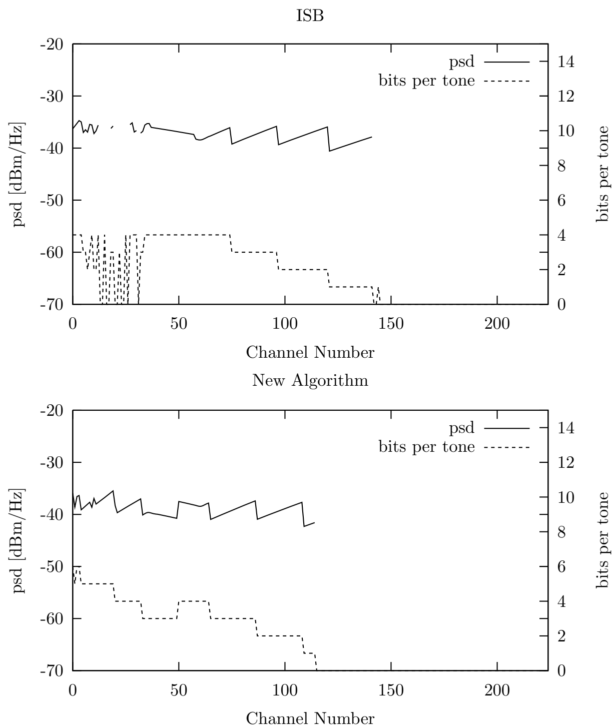

Fig. 7. Resulting Power Spectral Densities and bit allocation for ISB and the new algorithm for user 3 for the scenario in figure 6

rate for each user is 0.31%. The resulting PSD and bits per tone graphs for user 3 are shown in figure 7.

6.4 Discussion of Results

It is noted from figure 7 that the PSD graph of the new algorithm is more continuous than that of ISB. This is also true for the PSD graphs of the other 6 users

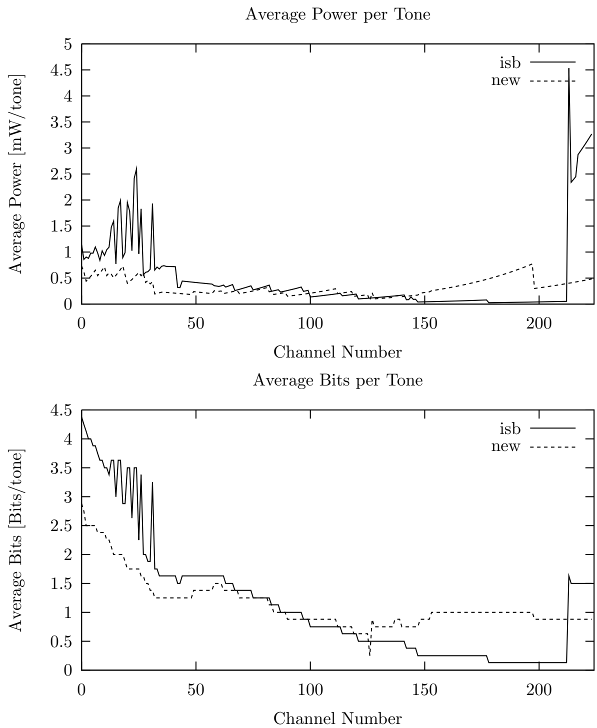

Fig. 8. Average Power and Average Bits per Tone for both ISB and new algorithm for scenario in figure 6

not shown here. This is further illustrated in figure 8 which shows the average power and average bits per tone for all lines in the scenario in figure 6 on each tone. It is clear from figure 8 that the PSD and hence the bit allocation of each user is flatter over all channels. It is speculated that this may make the algorithm particularly suitable for calculating the scaling factors in Band-Preference algorithms [7] [6]. In Band-Preference algorithms each modem executes a "scaled waterfilling" algorithm on each of its sub bands. This results in a relatively flat PSD on each of the sub bands. This has the effect of approximating a spectrum balancing solution given by, OSB, ISB or SCALE for example. With reference to figure 7 it relatively easy to see that the spectrum of the new algorithm could be better approximated with just two bands (one flat band at low frequencies and off at higher frequencies) than the ISB solution.

Furthermore, it appears that the new algorithm is able to leverage greater capacity at the higher frequency tones compared to ISB, and consequently requires less average power at lower frequencies. In practice it is expected that this will result in reduced adjacent channel interference compared to ISB.

7 Conclusion

New results for the algorithm developed in [16] have been presented. These demonstrate that the algorithm performs close to the optimum solution in three very different DSL bundle scenarios. The complexity is low enough to achieve results within a realistic time for a large number of DSL lines without significant optimisation.

Future work will attempt to formalise the calculation of the crosstalk damping weights w_n, investigate further complexity reductions and assess the suitability of the algorithm in relation to Band-Preference techniques.

Acknowledgment

The authors would like to thank DEL N.I. and Asidua Ltd. for their funding and support.

References

1. Ginis, G., Cioffi, J.: Vectored transmission for digital subscriber line systems. IEEE Journal on Selected Areas in Communications 20(5), 1085–1104 (2002)
2. Cendrillon, R., Moonen, M., Verlinden, J., Bostoen, T., Yu, W.: Optimal multiuser spectrum management for digital subscriber lines. In: IEEE International Conference on Communications, vol. 1, pp. 1–5 (2004)
3. Papandriopoulos, J., Evans, J.: Low-complexity distributed algorithms for spectrum balancing in multi-user dsl networks. In: IEEE International Conference on Communications, ICC 2006, June 2006, vol. 7, pp. 3270–3275 (2006)
4. Lee, J., Sonalkar, R., Cioffi, J.: Multiuser bit loading for multicarrier systems. IEEE Transactions on Communications 54(7), 1170–1174 (2006)
5. Cendrillon, R., Moonen, M.: Iterative spectrum balancing for digital subscriber lines. In: 2005 IEEE International Conference on Communications, ICC 2005, May 16-20, vol. 3, pp. 1937–1941 (2005)
6. Lee, W., Kim, Y., Brady, M.H., Cioffi, J.M.: Band-preference dynamic spectrum management in a dsl environment. In: Global Telecommunications Conference, GLOBECOM 2006, pp. 1–5. IEEE, Los Alamitos (2006)
7. Papandriopoulos, J., Evans, J.S.: Band preference design algorithms for improved iterative water-filling. In: Global Telecommunications Conference, GLOBECOM 2007, November 26-30, pp. 2899–2903. IEEE, Los Alamitos (2007)
8. Statovci, D., Nordstrom, T., Nilsson, R.: The normalized-rate iterative algorithm: A practical dynamic spectrum management method for dsl. Eurasip Journal on Applied Signal Processing 2006, 17 (2006)
9. Baldemair, R., Horvat, M., Nordstrom, T.: Proposed method of crosstalk calculations in a distributed environment. ETSI ETC TM6 Plenary Meeting (2003)
10. Starr, T., Cioffi, J.M., Silverman, P.J.: Understanding Digital Subscriber Line Technology. Prentice-Hall, Englewood Cliffs (1999)
11. Yu, W., Ginis, G., Cioffi, J.: Distributed multiuser power control for digital subscriber lines. IEEE Journal on Selected Areas in Communications 20(5), 1105–1115 (2002)

12. Lui, R., Yu, W.: Low-complexity near-optimal spectrum balancing for digital subscriber lines. In: IEEE International Conference on Communications, ICC 2005, May 16-20, vol. 3, pp. 1947–1951 (2005)
13. Cioffi, J.M.: Chapter 15 ee479 course notes - multiuser digital transmission systems, http://www.stanford.edu/class/ee479/
14. Campello, J.: Practical bit loading for dmt. In: IEEE International Conference on Communications, ICC 1999, vol. 2, pp. 801–805 (1999)
15. Akujuobi, C., Shen, J., Sadiku, M.: A new parallel greedy bit-loading algorithm with fairness for multiple users in a dmt system. IEEE Transactions on Communications 54(8), 1374–1380 (2006)
16. McKinley, A., Marshall, A.: A new penalty based algorithm for multi-user spectrum balancing in xdsl networks (submitted for inclusion in Networks 2008, Budapest, Hungary)
17. Oksman, V., Cioffi, J.: Noise models for vdsl performance validation (1999) ANSI-77E7.4/99.438R2
18. Tsiaflakis, P., Vangorp, J., Moonen, M., Verlinden, J.: A low complexity branch and bound approach to optimal spectrum balancing for digital subscriber lines. In: GLOBECOM - IEEE Global Telecommunications Conference, p. 4150666 (2007)