A LOW-COMPLEXITY ALGORITHM FOR UTILITY BASED SPECTRUM COORDINATION IN DSL SYSTEMS

Jeroen Verdyck[⋆] *Chris Blondia*[†] *Marc Moonen*[⋆]

[⋆] KU Leuven, Department of Electrical Engineering (ESAT)
STADIUS Center for Dynamical Systems, Signal Processing and Data Analytics
[†] University of Antwerp, Department of Mathematics-Computer Sciences
MOSAIC Modeling of Systems And Internet Communication

ABSTRACT

The static resource allocation which is usually assumed for the DSL physical layer leaves unused a significant portion of the achievable rate region. An alternative approach is to divide time into slots of short duration, and to change the resource allocation from each time slot to the next. A cross-layer scheduler then chooses a different resource allocation setting for each time slot by defining a utility function for each user n, and solving the corresponding network utility maximization (NUM) problem. For spectrum coordination, this NUM problem is non-convex and solving it is NP-Hard. This paper therefore introduces a fast algorithm, referred to as NUM-DSB, which converges to a local solution of the NUM problem. NUM-DSB can be applied to any NUM problem, regardless of the considered utility functions's characteristics. Simulation results show that NUM-DSB can compete with the state of the art algorithm for smooth non-convex network utility maximization.

***Index Terms*—** DSL, Spectrum Coordination, Cross-Layer Scheduling, Network Utility Maximization

1. INTRODUCTION

Dynamic spectrum management (DSM) techniques, which are used in digital subscriber line (DSL) networks to combat crosstalk, give rise to a rate region $\mathcal{R}$ which contains no single point that simultaneously maximizes the data rate of all users. Instead, there is a set of Pareto-optimal resource allocation settings that result in a data rate tuple on the edge of the rate region. DSL networks commonly use one such Pareto-optimal resource allocation for an extended period of time, thus leaving unused a significant portion of the rate region.

An alternative to this static resource allocation is to divide time into slots of short duration, and to change the resource allocation from one time slot to the next. A cross-layer scheduler then chooses one setting for each time slot in accordance with upper layer requirements. To this end, the cross-layer scheduler defines a non-decreasing utility function $U^n(\cdot)$ for each user n, and solves the corresponding network utility maximization (NUM) problem

$$\arg\max_{\boldsymbol{R}\in\mathcal{R}} \sum_n U^n(R^n), \tag{1}$$

where $\boldsymbol{R}$ is a vector containing the data rate R^n of each user n in the network. Examples of such cross-layer schedulers can be found in [1, 2, 3]. Many algorithms exist that solve problem (1), see e.g. [4, 5].

The DSM technique under consideration in this paper is spectrum coordination. For spectrum coordination, as well as for many other DSM techniques, problem (1) is non-convex, and finding its global optimum is NP-hard [6]. This is problematic, as a new NUM problem is to be solved for each time slot, and as it is desirable for time slots to be short.

This paper therefore introduces a fast algorithm which converges to a local solution of problem (1). The proposed strategy is to construct successive convex lower bound approximations of $\mathcal{R}$, which are denoted as $\tilde{\mathcal{R}}(\boldsymbol{s})$ where $\boldsymbol{s}$ corresponds to a specific resource allocation. It is demonstrated that for $\tilde{\mathcal{R}}(\boldsymbol{s})$, problem (1) is solved more easily. The resulting NUM-DSB algorithm thus consists of solving a sequence of NUM problems over different approximations $\tilde{\mathcal{R}}(\boldsymbol{s})$. NUM-DSB can be applied to any NUM problem, regardless of the characteristics of the utility functions.

The network performance gains that are enabled by a cross-layer scheduling algorithm that employs NUM-DSB have been demonstrated in [2]. In this paper, the performance of NUM-DSB itself is compared to the performance of the similar SJBR algorithm [7]. Results show that NUM-DSB, which can be applied to a wider variety of NUM problems

This research work was carried out at the ESAT Laboratory of KU Leuven, in the frame of 1) the Interuniversity Attractive Poles Programme initiated by the Belgian Science Policy Office: IUAP P7/23 ‘Belgian network on stochastic modeling analysis design and optimization of communication systems’ (BESTCOM) 2012-2017, 2) Research Project FWO nr. G.0912.13 ’Cross-layer optimization with real-time adaptive dynamic spectrum management for fourth generation broadband access networks ’, 3) VLAIO O&O Project nr. HBC.2016.0055 ’5GBB Fifth generation broadband access’. The scientific responsibility is assumed by its authors.

than SJBR, needs significantly fewer iterations to converge to a local solution of (1).

2. DSL SYSTEM MODEL & CROSS-LAYER SCHEDULING

Consider an N-user DSL system with K orthogonal sub channels or tones. As spectrum coordination is considered, each of these tones k is modeled as an interference channel

$$\boldsymbol{y}_k = H_k \boldsymbol{x}_k + \boldsymbol{z}_k. \tag{2}$$

In (2), $\boldsymbol{x}_k = \left[x_k^1, \ldots, x_k^N\right]^T$ is a vector containing the transmitted signal of each user. Also, let $\boldsymbol{x}^n = [x_1^n, \ldots, x_K^n]^T$ and $\boldsymbol{x} = \left[\boldsymbol{x}^{1T}, \ldots, \boldsymbol{x}^{NT}\right]^T$. Similar vector notation will be used for other signals, as well as for variables and functions introduced later such as the bit loading, total power consumption, and data rate. Furthermore, $\boldsymbol{y}_k$ and $\boldsymbol{z}_k$ contain the received signal and noise of each user. The average power of x_k^n is given as $s_k^n = \Delta_f \mathcal{E}\left\{|x_k^n|^2\right\}$, with $\mathcal{E}\{\cdot\}$ the expected value operator and Δ_f the tone spacing. Also, let $\sigma_k^n = \Delta_f \mathcal{E}\left\{|z_k^n|^2\right\}$ be the average noise power received by user n on tone k. Finally, H_k is the $N \times N$ channel matrix, where $[H_k]_{n,m} = h_k^{n,m}$ is the transfer function between the transmitter of user m and the receiver of user n, evaluated on tone k. The maximum achievable bit loading for user n on tone k, given the transmit powers $\boldsymbol{s}_k$, is calculated as

$$b_k^n(\boldsymbol{s}_k) = \log_2\left(1 + \frac{1}{\Gamma}\frac{|h_k^{n,n}|^2 s_k^n}{\sum_{n\neq m}|h_k^{n,m}|^2 s_k^m + \sigma_k^n}\right), \tag{3}$$

with Γ the SNR gap to capacity. The data rate and total power consumption of user n are respectively calculated as

$$R^n(\boldsymbol{b}^n) = f_s \sum_k b_k^n \qquad P^n(\boldsymbol{s}^n) = \sum_k s_k^n,$$

where f_s is the symbol rate.

The total transmit power of each user is limited to P^{tot}. The transmit spectrum of each user additionally has to satisfy the spectral mask constraint $\boldsymbol{s}^n \leq \boldsymbol{s}^{\text{mask}}$. The set of all possible power loadings of user n can thus be described as

$$\mathcal{S}^n = \left\{\boldsymbol{s}^n \in \mathbb{R}_+^K \mid \; P^n(\boldsymbol{s}^n) \leq P^{\text{tot}} \text{ and } \boldsymbol{s}^n \leq \boldsymbol{s}^{\text{mask}}\right\}. \tag{4}$$

The set of all possible power loadings of the whole multi-user system is $\mathcal{S} = \mathcal{S}^1 \times \ldots \times \mathcal{S}^N$. The resulting set of achievable bit loadings is

$$\mathcal{B} = \left\{\boldsymbol{b} \in \mathbb{R}_+^{N\times K} \mid \exists \boldsymbol{s} \in \mathcal{S} : \boldsymbol{b} \leq \boldsymbol{b}(\boldsymbol{s})\right\} \tag{5}$$

Finally, the rate region of a DSL system that employs spectrum coordination can be defined as

$$\mathcal{R} = \left\{\boldsymbol{R} \in \mathbb{R}_+^N \mid \exists \boldsymbol{b} \in \mathcal{B} : R^n = R^n(\boldsymbol{b}^n)\right\}. \tag{6}$$

The rate region $\mathcal{R}$ of a DSL network, for which tone spacing is small relative to the coherence bandwidth of the power transfer function, is a convex set [8].

A DSL system typically uses a fixed operating point for the physical layer for an extended period of time. This operating point can be selected such that some minimal rate requirements are satisfied [9], or such that some degree of fairness among users is achieved [10]. Due to this static resource allocation, a significant portion of the rate region is left unused and hence the DSL network is not used to its full potential.

An alternative to this static resource allocation is to divide time into slots of short duration, and to change the resource allocation from one time slot to the next. For each time slot t, the cross-layer scheduler decides on the specific power allocation $\boldsymbol{s}$ and resulting rate tuple $\boldsymbol{R}$ to be used. To this end, the cross-layer scheduler assigns a utility function $U^n(\cdot)$ to each user n, and chooses the physical layer setting such that it maximizes the sum of all utilities in the DSL network (1).

3. ALGORITHM DEVELOPMENT

As the rate region of a DSL system is convex [8], the solution to problem (1) can be calculated by solving a sequence of weighted rate sum maximization (WRS) problems [11]

$$\arg\max_{\boldsymbol{R}\in\mathcal{R}} \boldsymbol{\omega}^T \boldsymbol{R}, \tag{7}$$

where $\boldsymbol{\omega} = [\omega^1, \ldots, \omega^N]^T$ is a vector of weights. In the case of spectrum coordination, as well as for many other DSM techniques, problem (7) is non-convex on account of the bit loading being a non-convex function of the power allocation (3). Trying to find a globally optimal solution to problem (7), and by extension to problem (1), therefore results in algorithms of exceedingly high complexity. This is problematic as a new NUM problem is to be solved for each time slot, and as it is desirable for time slots to have a short duration in order to be able to adapt fast to changing upper layer requirements.

Inspired by the distributed spectrum balancing (DSB) algorithm for spectrum coordination [12], the proposed solution is to construct successive per-user convex lower bound approximations of the rate region for which problem (7) can be solved more easily. Approximations $\tilde{\mathcal{R}}(\boldsymbol{s})$ are constructed by defining an approximation for the bit loading that is a convex function of the power allocation $\boldsymbol{s}^n$. For each approximation of the rate region, the following problem is solved

$$\arg\max_{\boldsymbol{R}\in\tilde{\mathcal{R}}(\boldsymbol{s})} \sum_n U^n(R^n). \tag{8}$$

By iteratively constructing a new approximation of the rate region at the solution of the previous iteration, a local solution of problem (1) is found. The resulting algorithm is summarized in Algorithm 1, and is referred to as Distributed Spectrum Balancing for Network Utility Maximization (NUM-DSB). NUM-DSB can be applied to any NUM problem, regardless of the characteristics of the utility functions.

Algorithm 1. NUM-DSB

Initialize $\mathbf{s}^{(0)} \in \mathcal{S}$
for $\ell = 0, 1, \ldots$ **do**
  Select a user n and construct $\tilde{\mathcal{R}}(\boldsymbol{s}^{(\ell)})$
  Set $\boldsymbol{s}^{n(\ell+1)} = \boldsymbol{s}^{n\star}$, with $\boldsymbol{s}^{n\star}$ obtained from (8)
  Set $\boldsymbol{s}^{m(\ell+1)} = \boldsymbol{s}^{m(\ell)} \quad \forall m \neq n$
end for

In Subsection 3.1, it is explained how $\tilde{\mathcal{R}}(\boldsymbol{s})$ is constructed. In Subsection 3.2, an algorithm is described that solves problem (8).

3.1. Approximation of the Rate Region

In each iteration ℓ of the NUM-DSB algorithm, a user n constructs its own convex lower bound approximation of the rate region $\mathcal{R}$. Given the current iterate $\boldsymbol{s}^{(\ell)}$, the approximation of $\mathcal{R}$ is denoted as $\tilde{\mathcal{R}}(\boldsymbol{s}^{(\ell)})$. Let it be clear that, although this is not reflected in notation, $\tilde{\mathcal{R}}(\boldsymbol{s}^{(\ell)})$ is specific to user n. In order to construct $\tilde{\mathcal{R}}(\boldsymbol{s}^{(\ell)})$, it is assumed that all other users do not change their power allocation, i.e. $\boldsymbol{s}^{m(\ell+1)} = \boldsymbol{s}^{m(\ell)}, \forall m \neq n$. No approximation is used for the calculation of the bit loading of user n, i.e.

$$\tilde{\boldsymbol{b}}^n(\boldsymbol{s}^n; \boldsymbol{s}^{(\ell)}) = \boldsymbol{b}^n([\boldsymbol{s}^{1(\ell)^T}, \ldots, \boldsymbol{s}^{nT}, \ldots, \boldsymbol{s}^{N(\ell)^T}]^T). \quad (9)$$

The bit loading of all other users $m \neq n$ is however approximated with a lower bound hyperplane, i.e.

$$\tilde{\boldsymbol{b}}^m(\boldsymbol{s}^n; \boldsymbol{s}^{(\ell)}) = \boldsymbol{b}^m(\boldsymbol{s}^{(\ell)}) + \boldsymbol{\beta}^m(\boldsymbol{s}^{(\ell)}) \circ \left(\boldsymbol{s}^n - \boldsymbol{s}^{n(\ell)}\right). \quad (10)$$

$A \circ B$ denotes the Hadamard product of matrices A and B, and $\beta_k^m(\boldsymbol{s}_k{}^{(\ell)})$ is the directional derivative of $b_k^m(\cdot)$ along the n^{th} vector in the standard basis of $\mathbb{R}^n$ evaluated at $\boldsymbol{s}_k{}^{(\ell)}$.

As the utility functions $U^n(R^n)$ may be undefined for negative values of R^n, the approximation of the data rate for each user should have a non-negative value. This requirement is enforced by adding an additional constraint which guarantees that the value of the approximate bit loading $\tilde{b}_k^m$ remains positive. Keeping in mind that $\beta_k^m(\boldsymbol{s}_k{}^{(\ell)}) < 0$, the appropriate constraint is

$$s_k^n \leq \hat{s}_k = s_k^{n(\ell)} - \max_{\substack{m \neq n: \\ b_k^m(\boldsymbol{s}_k{}^{(\ell)}) \neq 0}} \frac{b_k^m(\boldsymbol{s}_k{}^{(\ell)})}{\beta_k^m(\boldsymbol{s}_k{}^{(\ell)})}. \quad (11)$$

The resulting set of all possible power loadings and corresponding set of achievable approximate bit loadings are

$$\tilde{\mathcal{S}}^n(\boldsymbol{s}^{(\ell)}) = \{\boldsymbol{s}^n \in \mathcal{S}^n \mid \boldsymbol{s}^n \leq \hat{\boldsymbol{s}}\} \quad (12)$$

$$\tilde{\mathcal{B}}(\boldsymbol{s}^{(\ell)}) = \left\{\boldsymbol{b} \in \mathbb{R}_+^{N \times K} \mid \exists \boldsymbol{s}^n \in \tilde{\mathcal{S}}^n(\boldsymbol{s}^{(\ell)}) : \boldsymbol{b} \leq \tilde{\boldsymbol{b}}(\boldsymbol{s}^n; \boldsymbol{s}^{(\ell)})\right\}$$

Finally, the approximate rate region is defined as

$$\tilde{\mathcal{R}}(\boldsymbol{s}^{(\ell)}) = \left\{\boldsymbol{R} \in \mathbb{R}_+^N \mid \exists \boldsymbol{b} \in \tilde{\mathcal{B}}(\boldsymbol{s}^{(\ell)}) : R^n = R^n(\boldsymbol{b}^n)\right\}. \quad (13)$$

Algorithm 2. CG algorithm for problem (8)

Initialize $\mathbf{R}^{(0)} \in \tilde{\mathcal{R}}(\boldsymbol{s}^{(\ell)})$
for $i = 0, 1, \ldots$ **do**
  Compute $\mathbf{r}' = \arg\max_{\mathbf{r} \in \tilde{\mathcal{R}}(\boldsymbol{s}^{(\ell)})} \mathbf{r}^T \nabla U(\tilde{\mathbf{R}}^{(i)})$
  Set $\boldsymbol{s}^{n(i+1)} = (1 - \gamma^{(i)})\boldsymbol{s}^{n(i)} + \gamma^{(i)}\boldsymbol{s}'$
  Set $\tilde{\boldsymbol{R}}^{(i+1)} = \mathbf{R}(\tilde{\boldsymbol{b}}(\boldsymbol{s}^{n(i)}; \boldsymbol{s}^{(\ell)}))$
end for

The same approximation of the rate region can straightforwardly be applied to other non-convex resource allocation problems, such as joint spectrum and signal coordination for upstream DSL [13].

An important feature of $\tilde{\mathcal{R}}(\boldsymbol{s})$ is that the approximation of the achievable rate is a lower bound on the actually achieved rate, which is due the lower bound hyperplane approximation used for the bit loading of users $m \neq n$. Combined with the fact that $U^n(\cdot)$ is monotonically increasing by definition, it can be concluded that

$$\sum_n U^n\big(R^{n(\ell)}\big) = \sum_n U^n\big(R^n\big(\tilde{b}_k^n(s_k^{n(\ell)}; \boldsymbol{s}_k^{(\ell)})\big)\big)$$
$$\leq \sum_n U^n\big(R^n\big(\tilde{b}_k^n(s_k^{n\star}; \boldsymbol{s}_k^{(\ell)})\big)\big) \leq \sum_n U^n\big(R^{n(\ell+1)}\big). \quad (14)$$

It is thus seen that each iteration of NUM-DSB increases the objective function value of problem (1).

3.2. Solving problem (8)

The algorithm presented here to solve problem (8) is based on the conditional gradient (CG) method, and can be shown to converge to the optimal solution of (8) if the utility functions $U^n(\cdot)$ are concave and continuously differentiable [14]. The derivation of the algorithm demonstrates that solving problem (8) is computationally far less demanding than directly solving (1). The conditional gradient algorithm for problem (8) is outlined in Algorithm 2.

In case the considered utility functions are not concave or not smooth, NUM-DSB can still be combined with other existing WRS based algorithms for NUM. Examples include a subgradient based dual decomposition algorithm that can be applied to NUM problems with non-smooth utility functions, or the monotonic optimization (MO) algorithm for NUM problems with non-concave utility functions [11].

Algorithm 2 constructs a sequence of linear problems of the form

$$\boldsymbol{s}^{n\prime} = \arg\max_{\boldsymbol{s}^n \in \tilde{\mathcal{S}}^n(\boldsymbol{s}^{(\ell)})} \quad \boldsymbol{\omega}^T \boldsymbol{R}\big(\tilde{\boldsymbol{b}}(\boldsymbol{s}^n; \boldsymbol{s}^{(\ell)})\big), \quad (15)$$

where $\boldsymbol{\omega}$ is a vector of positive weights. Positivity of these weights is guaranteed by the fact that the objective functions $U^n(\cdot)$ are increasing by definition. As problem (15) is convex, it can be solved by applying a dual decomposition method, i.e. by dualizing the total power constraints $\sum_k s_k^n \leq P^{\text{tot}}$, solving the resulting Lagrange dual problem, and extracting the solution to problem (15). The Lagrangian of problem (15) is given by

$$\mathcal{L}(\boldsymbol{s}^n, \lambda) = \boldsymbol{\omega}^T \boldsymbol{R}\big(\tilde{\boldsymbol{b}}(\boldsymbol{s}^n; \boldsymbol{s}^{(\ell)})\big) - \lambda\Big(\sum_k s_k^n - P^{\text{tot}}\Big), \quad (16)$$

and the resulting Lagrange dual problem of (8) is

$$\arg\min_{\lambda \geq 0} \left[g(\lambda) = \max_{\mathbf{0} \leq \boldsymbol{s}^n \leq \hat{\boldsymbol{s}}} \mathcal{L}(\boldsymbol{s}^n, \lambda) \right]. \quad (17)$$

Problem (17) is a convex problem in a single variable λ, and can be solved using a simple bisection method.

The Lagrange dual function $g(\lambda)$ is evaluated by maximizing $\mathcal{L}(\boldsymbol{s}^n, \lambda)$ independently on each tone k, i.e. by solving K optimization problems of the form

$$\arg\max_{0 \leq s_k^n \leq \hat{s}_k} \boldsymbol{\omega}^T \tilde{\boldsymbol{b}}_k(s_k^n; \boldsymbol{s}_k^{(\ell)}) - \lambda s_k^n. \quad (18)$$

As problem (18) corresponds to a convex problem in a single variable, its solution either satisfies the optimality condition $\boldsymbol{\omega}^T \frac{\partial}{\partial s_k^n} \tilde{\boldsymbol{b}}_k(s_k^n; \boldsymbol{s}_k^{(\ell)}) = \lambda$, or lies on the boundary of the feasible interval $[0, \bar{s}_k]$ with $\bar{s}_k = \min(s_k^{\text{mask}}, \hat{s}_k)$. Therefore, problem (18) is solved by the following expression.

$$\left[\frac{\omega^n / \log(2)}{\lambda - \sum_{m \neq n} \omega^m \beta_k^m(\boldsymbol{s}_k^{(\ell)})} - \Gamma \frac{\sum_{m \neq n} |h_k^{nm}|^2 s_k^{m(\ell)} + \sigma_k^n}{|h_k^{nn}|^2} \right]_0^{\bar{s}_k} \quad (19)$$

Convergence of Algorithm 2 is straigforwardly established from the results in [14] as follows. If the step size is chosen as $\gamma^{(i)} = \frac{2}{2+i}$, then

$$U(\tilde{\boldsymbol{R}}^*) - U\big((1-\gamma^{(i)})\tilde{\boldsymbol{R}}^{(i)} + \gamma^{(i)}\boldsymbol{r}'\big) \leq \mathcal{O}(1/i), \quad (20)$$

where $\tilde{\boldsymbol{R}}^*$ is the solution to problem (8) [14]. As the approximation of the bit loading is a concave function of the power allocation $\boldsymbol{s}^n$, it is readily seen that $U\big(\mathbf{R}^{(i+1)}\big) \geq U\big((1-\gamma^{(i)})\tilde{\boldsymbol{R}}^{(i)} + \gamma^{(i)}\boldsymbol{r}'\big)$. Therefore, the following result holds for Algorithm 2

$$U\big(\boldsymbol{R}^*\big) - U\big(\mathbf{R}^{(i+1)}\big) \leq \mathcal{O}(1/i). \quad (21)$$

4. SIMULATION RESULTS

The performance of NUM-DSB is compared to the performance of the SJBR algorithm of [7]. Like NUM-DSB, SJBR solves a sequence of convex approximations of problem (1). However, it differs from NUM-DSB in that it can only be applied if the utility functions of problem (1) are smooth [7]. The performance of the Gauss-Seidel variant of SJBR is compared to the performance of NUM-DSB.

Table 1. G.Fast parameter settings

Parameter	Value	Parameter	Value
$P^{n,\text{tot}}$	4 dBm	K	2047
f_s	48 kHz	Δ_f	51.75 kHz
Γ	12.6 dB	a^n	1 $\forall n \in \mathcal{N}$

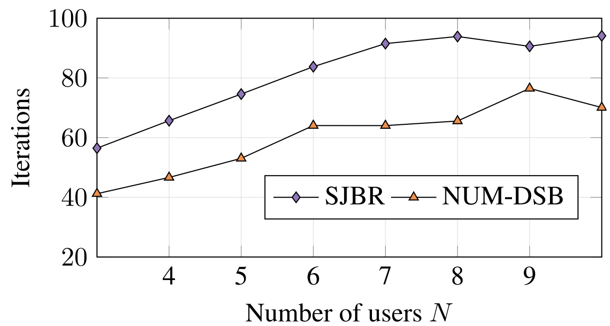

Fig. 1. Average number of iterations versus number of users in the DSL system. Gauss-Seidel execution is considered for both algorithms. All users sequentially update their transmit spectrum once per iteration.

The DSL network under consideration connects 10 users to a distribution point. The distance to the distribution point ranges from 110m for user 1 up to 200m for user 10, increasing with 10m for each consecutive user. The DSL networks for which $N \leq 10$ consist of the first N users of the above 10-user network. Parameter settings for the DSL system are summarized in Table 1. The considered utility functions are those of the minimal delay violation scheduler from [2], i.e. $U^n(R^n) = \frac{-a^n}{R^n}$. Spectral mask constraints are not included.

In Figure 1, the average number of outer iterations needed for convergence are presented for both NUM-DSB and SJBR. The algorithms are terminated when the decrease of the sum utility between two iterations relative to the sum of the objective function value for these two iterations is smaller than 10^{-6}. The average number of iterations is calculated over 100 random initializations of the transmit spectra $\boldsymbol{s}$. It should be noted that a single iteration of the SJBR algorithm has a lower complexity that an iteration of the NUM-DSB algorithm. However, the results show that NUM-DSB, which can also be applied to a wider variety of NUM problems than SJBR, needs significantly fewer iterations to converge.

5. CONCLUSION

The novel NUM-DSB algorithm for NUM based spectrum coordination has been presented. NUM-DSB can be employed regardless of the specific characteristics of the utility functions. Simulations have confirmed that NUM-DSB converges exceedingly fast, which enables its use in the computationally demanding context of cross-layer scheduling.

6. REFERENCES

[1] Paschalis Tsiaflakis, Yung Yi, Mung Chiang, and Marc Moonen, “Throughput and Delay Performance of DSL Broadband Access with Cross-Layer Dynamic Spectrum Management,” *IEEE Transactions on Communications*, vol. 60, no. 9, pp. 2700–2711, sep 2012.

[2] Jeremy Van Den Eynde, Jeroen Verdyck, Chris Blondia, and Marc Moonen, “Delay Performance Enhancement for DSL Networks through Cross-Layer Scheduling,” in *Proc. of the 37th WIC Symposium on Information Theory in the Benelux*, Louvain-La-Neuve, 2016, pp. 1–8.

[3] Guocong Song, Ye Li, and L.J. Cimini, “Joint channel- and queue-aware scheduling for multiuser diversity in wireless OFDMA networks,” *IEEE Transactions on Communications*, vol. 57, no. 7, pp. 2109–2121, jul 2009.

[4] Xiaojun Lin, N.B. Shroff, and R Srikant, “A tutorial on cross-layer optimization in wireless networks,” *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 8, pp. 1452–1463, aug 2006.

[5] Mung Chiang, Steven H. Low, a. Robert Calderbank, and John C. Doyle, “Layering as Optimization Decomposition: A Mathematical Theory of Network Architectures,” *Proceedings of the IEEE*, vol. 95, no. 1, pp. 255–312, jan 2007.

[6] Zhi-Quan Luo and Shuzhong Zhang, “Dynamic Spectrum Management: Complexity and Duality,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 2, no. 1, pp. 57–73, feb 2008.

[7] Gesualdo Scutari, Francisco Facchinei, Peiran Song, Daniel P. Palomar, and Jong-Shi Pang, “Decomposition by Partial Linearization: Parallel Optimization of Multi-Agent Systems,” pp. 1–14, 2013.

[8] Raphael Cendrillon, Wei Yu, Marc Moonen, Jan Verlinden, and Tom Bostoen, “Optimal multiuser spectrum balancing for digital subscriber lines,” *IEEE Transactions on Communications*, vol. 54, no. 5, pp. 922–933, may 2006.

[9] Martin Wolkerstorfer, Driton Statovci, and Tomas Nordström, “Dynamic spectrum management for energy-efficient transmission in DSL,” *2008 11th IEEE Singapore International Conference on Communication Systems, ICCS 2008*, pp. 1015–1020, 2008.

[10] Thierry Sartenaer, Jerome Louveaux, and Luc Vandendorpe, “Balanced capacity of wireline multiple access channels with individual power constraints,” *IEEE Transactions on Communications*, vol. 56, no. 6, pp. 925–936, jun 2008.

[11] Johannes Brehmer, *Utility Maximization in Nonconvex Wireless Systems*, vol. 5 of *Foundations in Signal Processing, Communications and Networking*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

[12] Paschalis Tsiaflakis, Moritz Diehl, and Marc Moonen, “Distributed Spectrum Management Algorithms for Multiuser DSL Networks,” *IEEE Transactions on Signal Processing*, vol. 56, no. 10, pp. 4825–4843, oct 2008.

[13] Paschalis Tsiaflakis, Rodrigo B Moraes, and Marc Moonen, “A low-complexity algorithm for joint spectrum and signal coordination in upstream DSL transmission,” in *2011 18th IEEE Symposium on Communications and Vehicular Technology in the Benelux (SCVT)*. nov 2011, pp. 1–6, IEEE.

[14] Martin Jaggi, “Revisiting Frank-Wolfe: Projection-Free Sparse Convex Optimization,” in *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, 2013, vol. 28, pp. 427–435.