Improving Group Orthogonal Matching Pursuit Performance with Iterative Feedback

Henning F. Schepker, Carsten Bockelmann and Armin Dekorsy
Department of Communications Engineering
University of Bremen, Bremen, Germany
email: {schepker, bockelmann, dekorsy}@ant.uni-bremen.de

***Abstract*—Machine-to-Machine communication requires new physical layer concepts to meet future requirements. In previous works it has already been shown that Compressive Sensing (CS) detectors are capable of jointly detecting both activity and data in multi-user detection (MUD). For this we propose a new generalized Group Orthogonal Matching Pursuit algorithm that allows the use of additional side information regarding the sparsity structure. As a specific example, we exploit the information of a sparsity-aware Viterbi decoder in an iterative feedback loop to improve the activity detection. A significant improvement of the activity detection is already achieved by executing only a single additional detection and decoding step.**

I. Introduction

The field of wireless Machine-to-Machine (M2M) communication is expected to grow tremendously in the future. This calls for new and adapted physical layer concepts, as system requirements differ from common applications such as high data rate access. Uplink transmission in a sensor network, as an example of a M2M communication application, is in general characterized by a large number of sensor nodes that only on occasion transmit a small amount of data, e.g., event driven or time controlled. This type of transmission is called *sporadic*, as each transmitter is inactive for most of the time.

A new detection paradigm developed in recent years is *Compressive Sensing* (CS) [1], [2], which is gaining more attention in the communication technology community. In a nutshell, CS shows that, assuming a signal has a sparse representation, this signal can be detected reliably even in highly under-determined systems. Since sporadic multi-user transmission can be interpreted as the transmission of sparse multi-user signals, we can apply CS detectors for multi-user detection (MUD), which is shown for a CDMA transmission in [3]–[5]. The main advantage of this CS MUD is that a joint detection of both node activity and transmitted data is performed, reducing the need to signal node activity. This property is especially beneficial for sensor nodes, where it can improve battery life or reduce complexity.

The previous research on CS MUD was primarily focused on symbol level detection without taking channel coding into account. In this paper, we investigate how the presence of a channel decoder can be exploited to improve the physical layer activity detection. More specifically, we introduce a weighted version of the Group Orthogonal Matching Pursuit (GOMP) algorithm [6] and use this algorithm in an iterative feedback loop which exploits the information from the channel decoder to improve activity detection.

This work was supported in part by the German Research Foundation (DFG) under grant DE 1858/1-1. Part of this work has been performed in the framework of the FP7 project ICT-317669 METIS, which is partly funded by the European Union. The authors would like to acknowledge the contributions of their colleagues in METIS, although the views expressed are those of the authors and do not necessarily represent the project.

II. Machine-to-Machine Scenario

We consider a M2M scenario, where K sensor nodes communicate with a central aggregation node, typically denoted as a star topology. Additionally, the transmissions from the sensor nodes are sporadic, i.e., the sensor nodes are only active on occasion. As a model for sensor node activity, we assume that each sensor node is active for a short time period with a given activity probability p_a. Further, we assume that this activity probability is identical for all sensor nodes and rather small, i.e., $p_a \ll 1$. For a large number of nodes K this is a valid assumption for practical applications.

The basic transmitter setup of the sensor nodes is depicted in Fig. 1. We assume that an active node k_a transmits a data frame of N_B information bits $\mathbf{b}_{k_\mathrm{a}} \in \{0,1\}^{N_\mathrm{B}}$. For simplicity, we assume that all nodes transmit the same number of information bits N_B, the system could easily be adapted to model unequal information bit sizes. For error protection, the information bits are encoded by a channel code of code rate R_C yielding code word vector $\mathbf{c}_{k_\mathrm{a}} \in \{0,1\}^{N_\mathrm{C}}$. After random interleaving, the code bits are mapped to BPSK symbols yielding a symbol vector $\mathbf{d}_{k_\mathrm{a}} \in \mathcal{A}^{N_\mathrm{C}}$ that contains the symbols of a frame, where $\mathcal{A}$ denotes the BPSK alphabet. The restriction to BPSK just simplifies the notation without loss of generality. Other modulation schemes can easily be applied. Due to the restriction to BPSK, we will consider a real-valued model. Further, as an inactive node k_i does not transmit any data, we model the transmitted symbols as zero symbols, i.e., $\mathbf{d}_{k_\mathrm{i}} \in \{0\}^{N_\mathrm{C}}$. Thus, the transmitted symbols for all nodes are taken from the so-called *augmented* alphabet $\mathcal{A}_0 = \{\mathcal{A} \cup 0\}$, which is the BPSK alphabet $\mathcal{A}$ augmented and extended by the zero symbol to indicate inactivity. Thus, each sensor transmits frames of N_C consecutive symbols drawn from the augmented alphabet $\mathcal{A}_0$.

In general, the received vector $\mathbf{y}$ at the aggregation node can be written as a multi-user model with

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{n}\,, \tag{1}$$

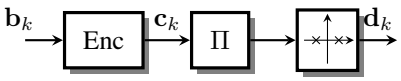

Fig. 1. Sensor node transmitter model of sensor k

Algorithm 1 Weighted Group Orthogonal Matching Pursuit (wGOMP)

$G^0 = \emptyset$, $\ell = 0$, $\mathbf{r}^0 = \mathbf{y}$
repeat
  $\ell = \ell + 1$
  $k_{\max} = \arg\max_k \sum_{j\in\Gamma(k)} \frac{|w_j \mathbf{A}_j^H \mathbf{r}^{\ell-1}|}{|\Gamma(k)|}$ with $k \in \overline{G}^{\ell-1}$
  $G^\ell = G^{\ell-1} \cup k_{\max}$
  $\hat{\mathbf{x}}^\ell_{\Gamma(G^\ell)} = \mathbf{A}^\dagger_{\Gamma(G^\ell)}\mathbf{y}$ **and** $\hat{\mathbf{x}}^\ell_{\Gamma(\overline{G}^\ell)} = \mathbf{0}$
  $\mathbf{r}^\ell = \mathbf{y} - \mathbf{A}\hat{\mathbf{x}}^\ell$
until $\ell = K_\mathrm{a}$

where the matrix $\mathbf{A} \in \mathbb{R}^{M\times KN_\mathrm{C}}$ models the various influences on the transmitted signal $\mathbf{x}$ and $\mathbf{n} \in \mathbb{R}^M$ is real-valued AWGN noise $\mathcal{N}(0, \sigma_n^2)$. The vector $\mathbf{x} \in \mathcal{A}_0^{KN_\mathrm{C}}$ contains the data from all nodes, i.e., it is the stacked vector of all node frames $\mathbf{d}_k$. The measurement dimension M depends on the system setup and the channel access technology being used.

III. Compressive Sensing

The theory of Compressive Sensing (CS) is focused on the recovery of sparse signals even from under-determined equation systems [1], [2]. Ideally, this reconstruction is done by solving non-convex optimization problems. As this is NP-hard, convex relaxations are considered instead, e.g., [7]. In addition to convex optimization, several algorithms have been proposed to efficiently determine a good estimation of the convex solution, e.g., iterative thresholding or greedy algorithms.

In this paper, we will focus on CS detection using greedy algorithms. These algorithms are in general more efficient, but less accurate than solving the convex optimization problem. While the Orthogonal Matching Pursuit (OMP) algorithm [8], [9] is a commonly used greedy algorithm, it is not well suited for our scenario. This is due to fact that node activity is constant for a frame, and thus all elements from one node are either all zero or non zero. This property is called *block-sparse* or *group-sparse*, where group k is given by the position of the elements of $\mathbf{d}_k$ in $\mathbf{x}$. One of the variants of the OMP called Group OMP (GOMP) [6] makes use of this property and is therefore better suited. In order to improve the detection performance further, we modify the GOMP such that we can use side information to improve the activity detection.

A. Weighted Group Orthogonal Matching Pursuit (wGOMP)

In order to discuss the weighted GOMP (wGOMP), we first explain the notation: G is a set of group-indices and $\overline{G}$ the complementary set. $\Gamma(k)$ specifies the vector-indices corresponding to group k and $\Gamma(G)$ specifies the vector-indices corresponding to any group in G. $\mathbf{A}_{\Gamma(G)}$ specifies the columns with vector-indices in $\Gamma(G)$ and $\mathbf{x}_{\Gamma(G)}$ the corresponding elemens of $\mathbf{x}$. Additionally, $\mathbf{x}^\ell$, $\mathbf{A}^\ell$ and G^ℓ specify the respective variable during the ℓ^{th} iteration. Herein, $\mathbf{A}^\dagger$ is the Moore-Penrose pseudoinverse of $\mathbf{A}$, and $\mathbf{A}^H$ the Hermitian matrix of $\mathbf{A}$.

The wGOMP, given in Algorithm 1, iteratively determines the support of $\mathbf{x}$, i.e., the location of the non-zero elements. During each iteration the wGOMP determines the group $k_{\max}$ that has the highest average correlation to the previous residual $\mathbf{r}^{\ell-1}$. Afterwards, the wGOMP calculates a new least-square (LS) estimate for $\hat{\mathbf{x}}$ based on the current and all previous group choices, and updates the residual $\mathbf{r}^\ell$. For simplicity we assume that the algorithm is terminated after a number of iterations equal to the number of active nodes K_a. For implementation, an appropriate termination criterion needs to be found that is well suited for the current system.

The wGOMP modifies the GOMP by introducing weights w_j for each correlation in the group selection step. These weights w_j promote the choice of nodes that are likely to be active due to side information, even though they may have low correlation. The weights w_j are set based on the available information in the system, e.g., activity in previous transmissions. As the weights are specific to each system, we will discuss further details for an exemplary system in the next section.

IV. CDMA Chip-rate System Model

It has been shown that CS is able to perform MUD for the CDMA chip-rate model written as a multi-user problem in (1) [4]. Therefore, we will use CDMA as a technique to facilitate sporadic and simultaneous medium access. For simplicity, we assume a synchronous CDMA uplink transmission with Pseudo Noise (PN) sequences, where the spreading factor N_S, i.e., the number of chips per information symbol, is identical for all sensor nodes. We assume that PN sequences are constant for a frame. Other medium access schemes are also possible, but the combination of those with CS is beyond the scope of this paper.

Based on [10], the multi-user chip-rate model can be written in the form of (1), where the measurement dimension M is given by $N_\mathrm{C}N_\mathrm{S}$. Here, $\mathbf{A} \in \mathbb{R}^{N_\mathrm{S}N_\mathrm{C}\times KN_\mathrm{C}}$ models the spreading, and the channel influences. As a channel model, we assume that the spread symbols are distorted by a node specific frequency selective channel $\mathbf{h}_k \in \mathbb{R}^{L_\mathrm{h}}$ of length L_h, which is constant for a whole frame. The chip-rate model (1) includes ISI (Inter-Symbol-Interference) within one frame, due to frequency selective channels. For simplicity, we do not capture frame start and end processing in this model. Thus, we omit the last $L_\mathrm{h} - 1$ received values and set the model to contain only $N_\mathrm{S}N_\mathrm{C}$ measurements.

A. Detection Model

As the size of the multi-user vector $\mathbf{x}$ is given by KN_C, the computational complexity quickly gets prohibitive for larger code word sizes N_C or many nodes K, necessitating a restricted model for detection purposes. Therefore, (1) is

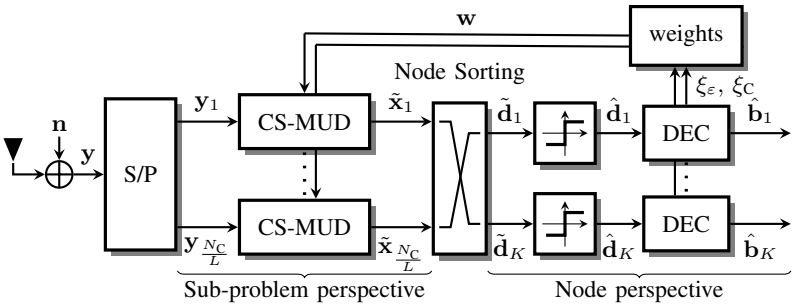

Fig. 2. Detection structure at the aggregation node. Interleaver included in the decoder block for space reasons.

divided into $\nu = 1, \ldots, N_C/L$ sub-problems, each considering L consecutive transmit symbols per node. Thus, each sub-problem is determined by the system matrix $\mathbf{A}_\nu \in \mathbb{R}^{N_S L \times KL}$, which is the same for all ν, as both PN sequences and channels are assumed to be constant for an entire frame. In order to simplify the detection model, we neglect the ISI between sub-problems in the detection. The transmit vector $\mathbf{x}_\nu \in \mathcal{A}_0^{KL}$ summarizes L transmit symbols for each of the K nodes in the ν^{th} sub-problem as

$$\mathbf{x}_\nu = [d_{1,L(\nu-1)+1}, \ldots, d_{1,L\nu}, \ldots , d_{K,L(\nu-1)+1}, \ldots, d_{K,L\nu}]^T . \quad (2)$$

The sub-problem dimension L is chosen in the system design, and defines a tradeoff between reduced complexity and detection accuracy.

A common assumption in CS literature is that the system matrix has *unit norm columns*, i.e., columns with identical ℓ_2-norm. However, the system matrix $\mathbf{A}_\nu$ does not have this property as the channel coefficients $\mathbf{h}_k$ only have average values that satisfy $\|\mathbf{h}_k\|_2^2 = 1$. To solve this problem, we modify the system matrix in each of the $\nu = 1, \ldots, N_C/L$ CS detection sub-problems to have unit norm columns, as shown in [11], such that

$$\mathbf{y}_\nu = \tilde{\mathbf{A}}_\nu \tilde{\mathbf{x}}_\nu + \mathbf{n}_\nu , \quad (3)$$

where $\mathbf{n}_\nu \in \mathbb{R}^{N_S L}$ is real-valued AWGN noise $\mathcal{N}(0, \sigma_n^2)$, $\mathbf{y}_\nu \in \mathbb{R}^{N_S L}$ denotes the received sub-vector at the aggregation node, and $\tilde{\mathbf{x}}_\nu = \tilde{\mathbf{H}}_\nu \mathbf{x}_\nu$. Here, $\tilde{\mathbf{H}}_\nu$ is a diagonal matrix containing the column ℓ_2-norms of $\mathbf{A}_\nu$, and $\tilde{\mathbf{A}}_\nu$ is the matrix $\mathbf{A}_\nu$ with each column divided by its ℓ_2-norm. The values in $\tilde{\mathbf{x}}_\nu$ of node k are scaled according to the column ℓ_2-norm, such that $\tilde{\mathcal{A}}_{0,k} = \{0, \pm\|\mathbf{A}_{\nu,k}\|_2\}$. For simplicity, we treat the elements as being continuous during the detection and then quantize the estimated frame for each node to $\hat{\mathbf{d}}_k \in \mathcal{A}_0^{N_C}$.

B. Applying wGOMP

In this system example, we assume that there is no correlation in node activity between transmissions and that the only a-priori information is that each node is active with probability p_a. With additional information, the initial weights need to be set accordingly.

To improve the detection, the wGOMP is used in an iterative feedback loop as shown in Fig. 2. During the first cycle, an initial wGOMP with weights $w_j = 1 \ \forall j$ is applied, which is identical to applying GOMP. Then the results of the individual sub-problems are sorted and stacked by nodes, and quantized to $\hat{\mathbf{d}}_k \in \mathcal{A}_0^{N_C}$ for each node. Afterwards, the channel decoder is used to estimate the probability that a given node was active. This information is then mapped to weights and used in the wGOMP detection of the following cycle. This iterative feedback is continued until either a maximum number of cycles is reached or the feedback process has converged. This convergence occurs if the difference of the weights in the current and the previous cycle is sufficiently small, such that the weights will not result in different activity detection.

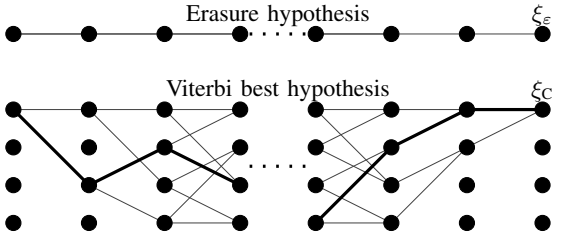

Fig. 3. Trellis diagram for a sparsity-aware Viterbi detector.

To determine the exact probability that a node was active using soft-decoding of the channel code, we need to know the soft values regarding the activity for each element. While we can calculate the soft values for elements detected as active, there is no information about how reliable elements are which were detected as inactive. The reason for this is that the wGOMP returns hard-decisions regarding the node activity, i.e., inactive elements are always zero, regardless whether the node had a high or a low correlation. Therefore, we cannot use soft-decoding to determine activity probabilities.

A good approximation can be calculated using a hard-decision Viterbi decoder. This detector is already capable of handling erasures, i.e., active elements that were detected as inactive, within the code word, but does not consider the case that the node was inactive for the duration of the code word. Therefore, we expand the Viterbi decoder to not only compute the Euclidean distance ξ_C of the most likely code word, but also the Euclidean metric ξ_ε for the erasure hypothesis. This Euclidean metric is the distance to the all erasure word in BPSK notation, i.e., $\xi_\varepsilon(\hat{\mathbf{d}}_k) = \|\hat{\mathbf{d}}_k\|_2$, which gives an indication how likely node k was inactive. The decoder then chooses the hypothesis with the smaller Euclidean distance. We call this decoder the *sparsity-aware Viterbi decoder*, shown

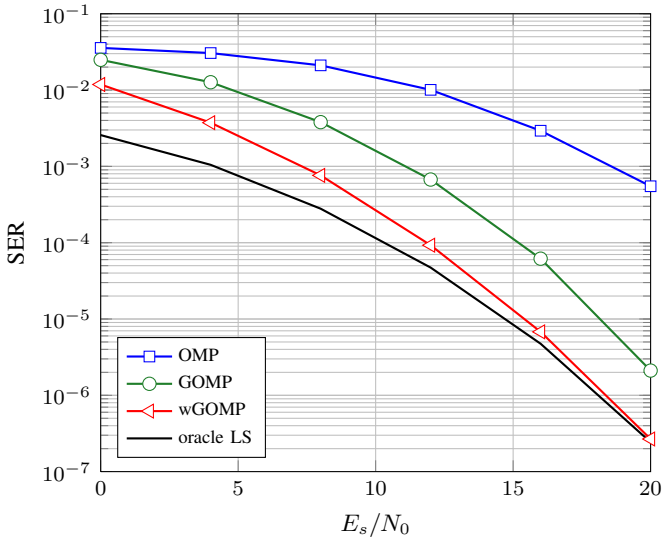

Fig. 4. Symbol Error Rate over the Augmented Alphabet.

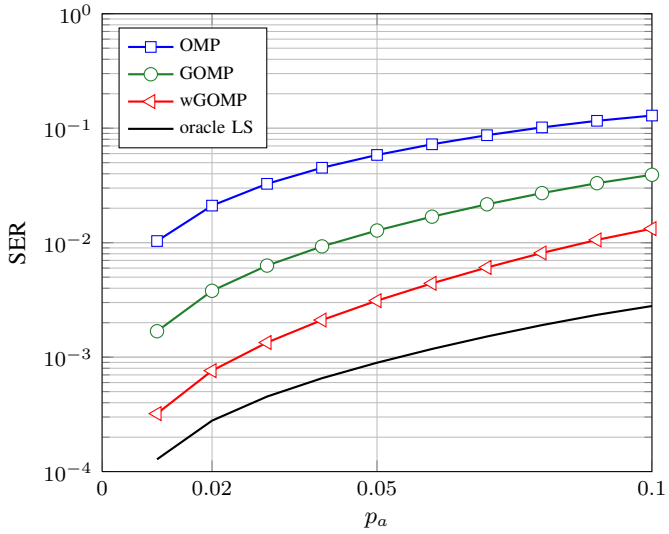

Fig. 5. Symbol Error Rate for a range of activity probabilities $p_a = 0.01, \ldots, 0.1$ and fixed $E_s/N_0 = 8$dB.

in Fig. 3. With these Euclidean distances as side information, we compute the weights w_j indicating the activity probability as

$$w_j = 0.5 + \frac{\xi_{\mathrm{C}}(\hat{\mathbf{d}}_k) - \xi_{\varepsilon}(\hat{\mathbf{d}}_k)}{2N_{\mathrm{C}}} \; \forall \; j \in \Gamma(k)\,. \tag{4}$$

Since the Euclidean distances are in the range of 0 to N_{C}, due to hard-decision, the normalization with $2N_{\mathrm{C}}$ ensures weights w_j in the range of 0 to 1. Using the information of the decoder we can only compute weights per node, as the decoder only has information on the code word level.

V. Simulation Results

In this section, we will discuss simulation results for the wGOMP in the iterative feedback structure described in Fig. 2, and compare them with simulation results for OMP and GOMP. The wGOMP was executed with a maximum number of feedback cycles equal to five, unless otherwise noted. Additionally, the symbol error rate (SER) for the best case performance and thus the lower bound for these algorithms is given by least-squares estimation for a known support of $\mathbf{x}_\nu$. We call this ideal detector the *oracle LS*.

As a simulation setup, we focus on the CS detection in an overloaded CDMA system. More specifically, we consider a transmission from $K = 128$ sensor nodes, where each node transmits using a PN sequence of length $N_{\mathrm{S}} = 32$, thus the system is overloaded by a factor of four. Unless otherwise noted, we assume that sensor nodes are only active with a probability of $p_a = 0.02$, such that the number of active nodes is on average much smaller than K. Furthermore, the frame length is $N_{\mathrm{C}} = 104$ symbols using a standard terminated $[7;5]_8$ convolutional code (code rate $R_{\mathrm{C}} = 1/2$, constraint length $L_{\mathrm{C}} = 3$). The channel is modeled by $L_{\mathrm{h}} = 6$ i.i.d. Rayleigh distributed taps with an exponential decaying power delay profile. The CS detection uses $L = 8$ consecutive symbols per sub-problem.

The SER over the augmented alphabet $\mathcal{A}_0$ contains both errors due to incorrect activity detection and errors due to incorrect data detection. Fig. 4 shows that the GOMP has a lower SER than the OMP for the entire SNR region, as it uses the information that each node is active for an entire sub-problem during the detection. Additionally, Fig. 4 shows that the wGOMP is able to further improve on the performance of the GOMP by the use of iterative feedback of frame activity information based on channel decoding. When comparing the wGOMP to the oracle LS, we can see that the wGOMP converges on this lower bound in the high SNR region. This comparison also indicates that the errors of OMP, GOMP and wGOMP are primarily activity errors.

In addition to these SER results, Fig. 5 shows the SER of the detectors for fixed $E_s/N_0 = 8$dB and a range of $p_a = 0.01, \ldots, 0.1$. Here, the main observation is that wGOMP scales slightly worse with p_a than GOMP. The reason for this is that subsequent wGOMPs in the iterative feedback loop are based on the performance of the initial GOMP, and a worse performance for the GOMP also means less reliable information for the weights.

On the symbol level, there are two types of activity errors. On the one hand errors for estimating an active symbol as inactive, called *missed detection*, and on the other hand estimating an inactive symbol as active, called *false alarm*. For both the Missed Detection Rate (MDR) and the False Alarm Rate (FAR) shown in Fig. 6 results indicate similar behavior as for the SER. The GOMP has a better performance than the OMP for the entire SNR region, and especially for high SNR. The wGOMP improves on the performance of the GOMP by approximately 4dB.

It should be noted that the main drawback of the wGOMP is a longer runtime than the GOMP. The relatvie runtime of the wGOMP compared to the GOMP is proportional to the number of feedback cycles necessary for convergence. Fig. 6 shows the difference between two feedback cycles and the previous setup of five feedback cycles, denoted as wGOMP. These results show that when executing only one additional

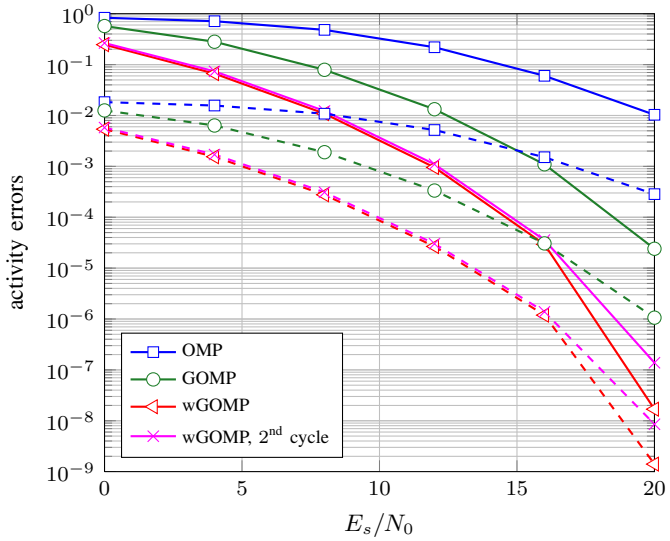

Fig. 6. Missed Detection Rate (solid) and False Alarm Rate (dashed).

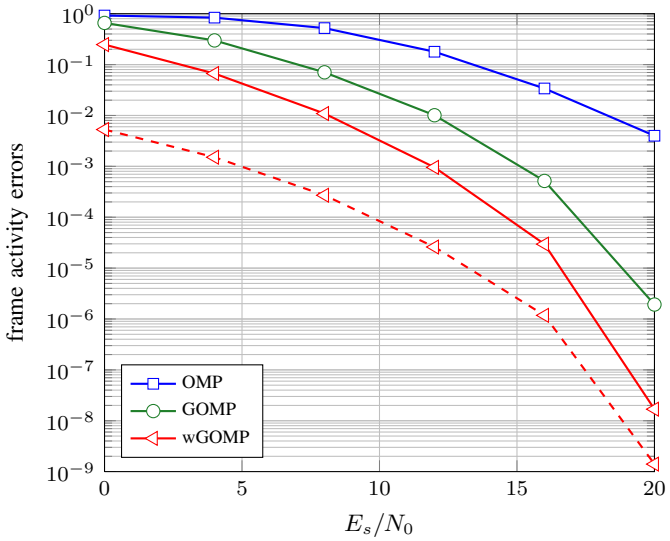

Fig. 7. Frame Missed Detection Rate (solid) and Frame False Alarm Rate (dashed).

wGOMP detection after the initial detection, the activity detection performance is almost fully converged. Therefore, we only need to execute more than two feedback cycles in the high SNR region and for very low error rates.

On the frame level, the activity decision is done by the sparsity-aware Viterbi decoder. Fig. 7 shows the Frame Missed Detection Rate (FMDR) for the detectors. When comparing these results with the symbol level errors in Fig. 6, we see that both the OMP and the GOMP have slightly lower error rates at the frame level. The reason for this is that each sub-problem perceives a different noise realization, and therefore the symbol level errors will be distributed differently among nodes in each sub-problem. Therefore, it is less likely that these errors cause the entire frame to be decided as inactive. For the wGOMP we can note that the FMDR is not only lower than both the OMP and GOMP, but also equal to the MDR. The reason for this is that the wGOMP enforces the frame level behavior on the symbol level, and thus promotes correlation in the errors.

For the Frame False Alarm Rate (FFAR) also shown in Fig. 7, it should be noted that for the OMP and the GOMP no FFAR could be measured in our simulation of 10^5 code words. Thus, we can only state that the FFAR of the OMP and GOMP are lower than the FFAR of the wGOMP. This is due to the distribution of the false alarm errors among the different nodes, such that each of those nodes has a low probability of causing a frame error. For the wGOMP, we can note the same behavior as for the FMDR, i.e., the FFAR is identical to the FAR. This means, that a drawback of the wGOMP is an increased FFAR. However, how severe this drawback is depends on the error handling at the higher layers and on the system design.

VI. Conclusion

In this paper, we investigated CS MUD using greedy algorithms in a M2M context. We introduced the weighted GOMP (wGOMP) detector as a new CS detector that is based on the GOMP. By exploiting side information determined by Viterbi decoding in an iterative feedback loop we were able to show that the activity detection of the wGOMP is significantly improved compared to a GOMP. In the high SNR region, the wGOMP even converges on the lower SER bound, while only requiring one additional greedy CS detection for this performance. In future works, it should be investigated which stopping criterion performs best in regard to a wGOMP used in such a feedback loop.

References

[1] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, April 2006.

[2] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, February 2006.

[3] H. Zhu and G. B. Giannakis, "Exploiting sparse user activity in multiuser detection," *IEEE Transactions on Communications*, vol. 59, no. 2, pp. 454–465, February 2011.

[4] H. F. Schepker and A. Dekorsy, "Sparse multi-user detection for CDMA transmission using greedy algorithms," in *8th International Symposium on Wireless Communication Systems*, Aachen, Germany, November 2011.

[5] H. Schepker and A. Dekorsy, "Compressive sensing multi-user detection with block-wise orthogonal least squares," in *IEEE 75th Vehicular Technology Conference*, Yokohama, Japan, May 2012.

[6] A. Majumdar and R. K. Ward, "Fast group sparse classification," *Electrical and Computer Engineering, Canadian Journal of*, vol. 34, no. 4, 2009.

[7] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *Journal of the Royal Statistical Society, Series B*, vol. 67, pp. 301–320, 2005.

[8] Y. Pati, R. Rezaiifar, and P. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," *Signals, Systems and Computers*, vol. 1, pp. 40–44, November 1993.

[9] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Transactions on Information Theory*, vol. 53, no. 12, pp. 4655–4666, December 2007.

[10] S. Verdú, *Multiuser Detection*. Cambridge, U.K.: Cambridge Univ. Press, November 1998.

[11] S. Rangan, A. Fletcher, and V. Goyal, "Asymptotic analysis of MAP estimation via the replica method and applications to compressed sensing," *IEEE Transactions on Information Theory*, vol. 58, no. 3, pp. 1902 –1923, March 2012.