EURASIP Journal on Wireless Communications and Networking

RESEARCH Open Access

Compressive sensing based random access for machine type communications considering tradeoff between link performance and latency

Yujing He, Wan Chen, Ishtiaq Ahmad, Lin Shi and Kyung Hi Chang*

Abstract

Machine type communications (MTC) in the next generation of mobile communication systems require a new random access scheme to handle massive access with low signaling overhead and latency. The recently developed compressive sensing multi-user detection (CS-MUD) supports joint activity and data detection by exploiting the sparsity of device activity. In this paper, we adopt the CS-based random access scheme by assigning the unique identification sequences to distinguish the different sensor nodes and employ group orthogonal matching pursuit least square (GOMP-LS) and weighted iteration (WI) GOMP algorithms based on the conventional GOMP with respect to high-reliability and low-latency applications. In addition, to further reduce computational complexity and latency, we introduce a low complexity WIGOMP with inverse Cholesky factorization (WIGOMP-ICF). Based on the simulation results and analysis, we can observe that the proposed three algorithms are promising to support different services requirements for MTC by considering high reliability, low latency, and computational complexity.

Keywords: Random access, Compressive sensing, Machine type communication, Latency

1 Introduction

In order to support a number of devices for the internet of things (IoT), there has been a growing interest in machine type communications (MTC). MTC is also a critical issue in next generation of mobile communication systems [1]. The applications of MTC are diverse from health care to smart grid, where a huge number of devices exist in the system, but only a few of them are active at a particular timing instance [2, 3]. In LTE, establishing a connection requires a relatively complex handshaking procedure [4, 5]. Such an approach is suitable for a system serving only a few high-activity users, but it becomes very cumbersome for MTC traffic, where large amounts of low-activity users intermittently transmit a small number of packets [6, 7]. Since traffics in MTC are sporadic, the CS-based random access has been widely considered to support a number of devices with low-signaling overhead and low latency for MTC [8, 9].

CS is an alternative approach to Shannon/Nyquist sampling, where a sparse or compressible signal can be sampled at a rate much less than the Nyquist rate [10, 11]. In [8], a CS-based random access scheme was proposed to exploit the sparsity of active devices for multi-user detection (MUD), where sensor nodes directly transmit their data to an access point (AP) without going through the request–grant procedure. The AP can not only detect the signals from active devices, but also identify them (through their unique signature sequences) simultaneously. To further improve performance, a CS-based random access scheme with multiple-sequence spreading was proposed by Abebe and Kang [12], and two novel approaches were also proposed by Schepker et al. [13], who introduced a channel decoder into the CS detector.

* Correspondence: khchang@inha.ac.kr
Department of Electronic Engineering, Inha University, Incheon, Korea

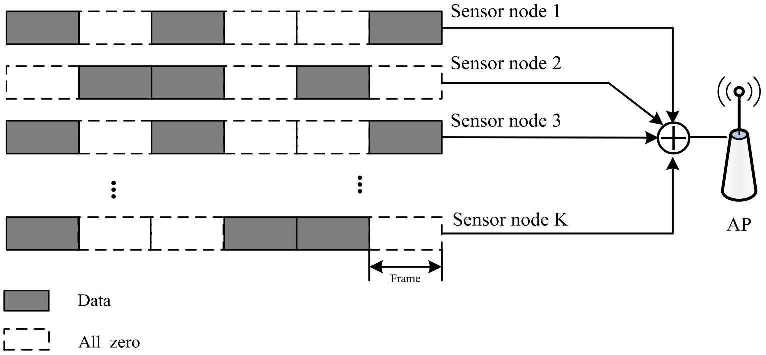


Fig. 1 Random access scenario of MTC

In this paper, we utilize the unique identification sequences to distinguish the different sensor nodes, which is attractive for MTC due to its flexibility in supporting various data rates as well as the quality of service. In [14, 15], the number of users is greater than the processing gains in an overloaded communication system, if there are a few active users, CS-based MUD can effectively detect their signals. In addition, we consider the quality of service (QoS) requirements of different applications in our proposed algorithms. First, we propose GOMP-LS as an improved GOMP algorithm to enhance link performance for high-reliability applications. However, GOMP-LS introduces additional processing delay, which is not suitable for low-latency applications. To satisfy low-latency applications requirements, WIGOMP algorithm is proposed as a tradeoff. In WIGOMP, the link performance is inferior to GOMP-LS, but complexity and latency are obviously lower than GOMP-LS. In addition, WIGOMP-ICF is proposed to further reduce the complexity by applying inverse Cholesky factorization.

The rest of this paper is organized as follows. In Section 2, the system model of CS-based random access for MTC is presented. In Section 3, the conventional GOMP algorithm is introduced and three improved GOMP algorithms considering a tradeoff between link performance and latency for various applications are proposed. Section 4 gives the simulation results and analyzes the latency and complexity of each algorithm and the simulation results for link performance are also discussed in detail. Finally, Section 5 concludes the paper.

2 System methodology for CS-based random access model

In this paper, we consider a random access MTC system that consists of one AP and K sensor nodes. Each sensor sends small packets independently to the AP with a certain low-active probability. We employ the proposed CS-based random access scheme to reconstruct the original signals at the AP. The proposed scheme reduces the signaling overhead and latency compared to the conventional LTE

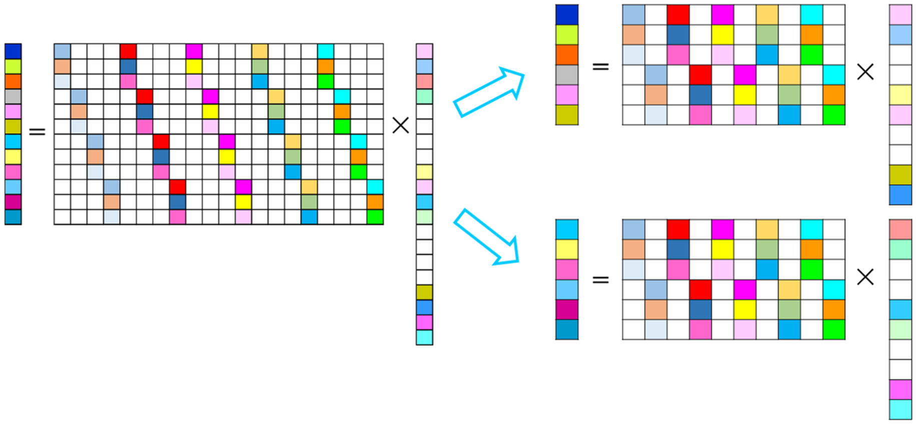


Fig. 2 Dividing a complicated problem into two simple sub-problems, where $K = 5$, $M = 3$, $N_c = 4$, and $v = 2$

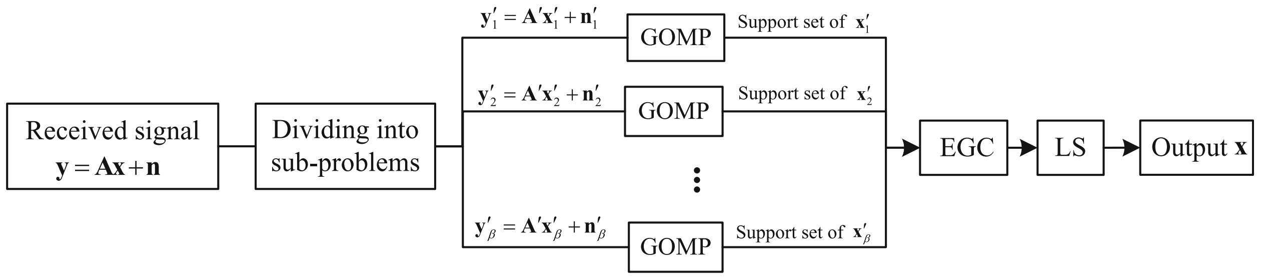

Received signal
$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{n}$
Dividing into sub-problems
$\mathbf{y}'_1 = \mathbf{A}'\mathbf{x}'_1 + \mathbf{n}'_1$
GOMP
Support set of $\mathbf{x}'_1$
$\mathbf{y}'_2 = \mathbf{A}'\mathbf{x}'_2 + \mathbf{n}'_2$
GOMP
Support set of $\mathbf{x}'_2$
$\mathbf{y}'_\beta = \mathbf{A}'\mathbf{x}'_\beta + \mathbf{n}'_\beta$
GOMP
Support set of $\mathbf{x}'_\beta$
EGC
LS
Output $\mathbf{x}$

Fig. 3 Block diagram of GOMP-LS

random access scheme. Moreover, we utilize a unique spreading sequence to distinguish the sensor nodes. Thus, the sensor nodes can efficiently utilize the resources by transmitting the data on the same resource block. Furthermore, we divide the original problem into several sub-problems in order to reduce the computational complexity of the CS-based random access model.

2.1 CS method details

CS is an emerging theory based on the fact that the salient information of a signal can be recovered from much fewer samples than the requirements of the Nyquist rate. According to CS theory, considering a real-valued discrete time signal $\mathbf{z}$ with a finite length N,$\mathbf{z}$ can be expressed on an orthonormal basis $\boldsymbol{\Psi} = [\boldsymbol{\psi}_1, \boldsymbol{\psi}_2, \ldots, \boldsymbol{\psi}_N]$ as

$$\mathbf{z} = \sum_{i=1}^{N} \alpha_i \boldsymbol{\psi}_i = \boldsymbol{\Psi}\mathbf{x} \tag{1}$$

where $\mathbf{x}$ is the coefficient sequence of $\mathbf{z}$, $\mathbf{x}_i = \langle \mathbf{z}, \boldsymbol{\psi}_i \rangle$. When only S of the coefficients $\mathbf{x}_i$ are nonzero, and $S \ll N$, the signal $\mathbf{z}$ is compressible and has a sparse representation, called$S$ - sparse, and $\mathbf{x}$ is the sparse representation of $\mathbf{z}$[16].

The sparse signal $\mathbf{z}$ can be a compact measurement under $M \times N$ $(M < N)$ measurement matrix Φ, and the sampled vector $\mathbf{y}$ can be expressed as

$$\mathbf{y} = \boldsymbol{\Phi}\mathbf{z} = \boldsymbol{\Phi}\boldsymbol{\Psi}\mathbf{x} = \mathbf{A}\mathbf{x} \tag{2}$$

where $\mathbf{A} = \boldsymbol{\Phi}\boldsymbol{\Psi}$ is called the sensing matrix. In general, because M is less than N, it is impossible to recover $\mathbf{x}$ from $\mathbf{y}$ in this underdetermined equation. However, if $\mathbf{x}$ is S - sparse and $S < M$, it can be recovered by the following l_1 − norm minimization:

$$\bar{\mathbf{x}} = \arg \min_{\mathbf{x}} \|\mathbf{x}\|_{l_1} \ \ s.t. \ \ \mathbf{y} = \mathbf{A}\mathbf{x} \tag{3}$$

Several algorithms, which can exactly recover a sparse signal with high probability, have been proposed to solve this convex problem, including matching pursuit (MP) [17], orthogonal matching pursuit (OMP) [18], GOMP [19], and chaining pursuit (CP) [20]. OMP is an efficient iterative greedy algorithm that can recover the support set of a sparse signal by selecting the column most correlated with the current residuals in each iteration.

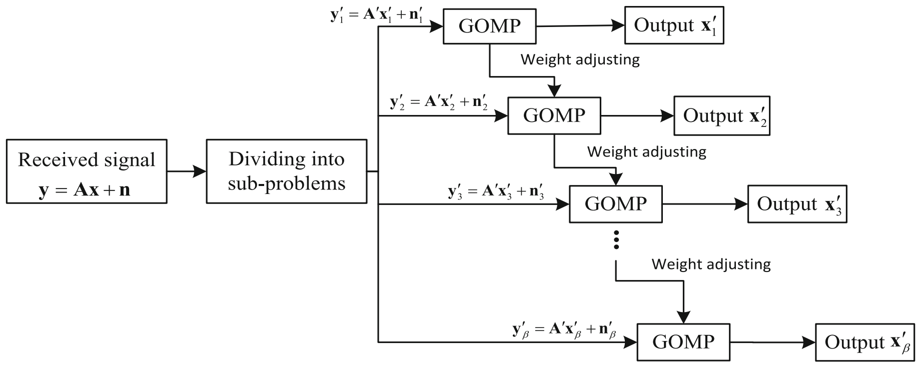

Received signal
$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{n}$
Dividing into sub-problems
$\mathbf{y}'_1 = \mathbf{A}'\mathbf{x}'_1 + \mathbf{n}'_1$
GOMP
Output $\mathbf{x}'_1$
Weight adjusting
$\mathbf{y}'_2 = \mathbf{A}'\mathbf{x}'_2 + \mathbf{n}'_2$
GOMP
Output $\mathbf{x}'_2$
Weight adjusting
$\mathbf{y}'_3 = \mathbf{A}'\mathbf{x}'_3 + \mathbf{n}'_3$
GOMP
Output $\mathbf{x}'_3$
Weight adjusting
$\mathbf{y}'_\beta = \mathbf{A}'\mathbf{x}'_\beta + \mathbf{n}'_\beta$
GOMP
Output $\mathbf{x}'_\beta$

Fig. 4 Block diagram of WIGOMP

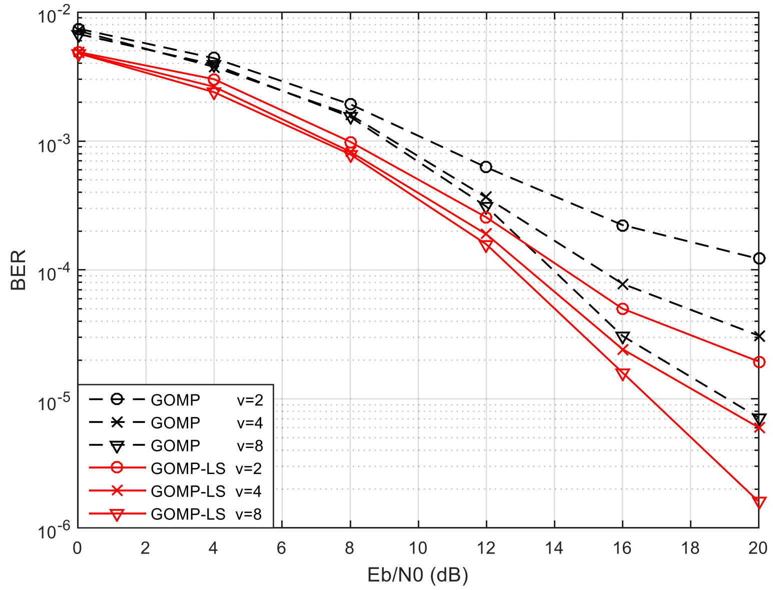


Fig. 5 BER performance of GOMP and GOMP-LS algorithms varying group size

2.2 CS-based random access model

In this paper, we consider a frame-synchronized system with one AP and K sensor nodes. In this system, the AP is capable of sophisticated signal processing, while sensor nodes with a simple structure produce small packets occasionally and send them to the AP in the recent frame starting time, as shown in Fig. 1. We assume that each sensor node has data to send in a frame independently, with a certain probability, p_a, called the activity probability [21, 22] (in practical communications system, especially a massive MTC system, $p_a \ll 1$).

We assume there are N_c symbols in a frame. The transmitted signal of the kth $(k = 1, 2, \ldots, K)$ sensor node is $\mathbf{x}_k = [x_1^{(k)}, x_2^{(k)}, \ldots, x_{N_c}^{(k)}] \in \mathfrak{R}^{N_c \times 1}$. $x_n^{(k)}$ is the nth$(n = 1, 2, \ldots, N_c)$ transmission symbol of the kth sensor node. Furthermore, the inactive sensor nodes that do not have data to send seem that all the transmitting data are 0, while the active sensor nodes, which have data

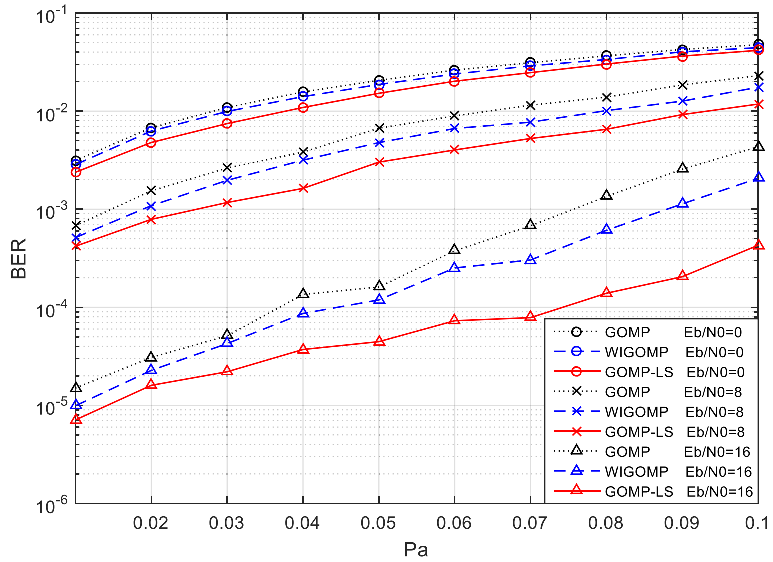


Fig. 6 Activity probability versus BER

to send, transmit symbols from the discrete finite modulation alphabet Q. Therefore, on the receiver side, the detected symbols are from the augmented modulation alphabet, i.e., $x_n^{(k)} \subset \{Q \cup 0\}$.

In this random access system, all the sensor nodes transmit their data in the same time-frequency resource block. To identify all the sensor nodes, a unique spreading sequence is assigned permanently to each sensor node. The spreading sequence for the kth sensor node is $\mathbf{s}^{(k)} \in \mathfrak{R}^M$, where M is the spreading factor. After modulation and spreading, transmitted signals are distorted by the frequency-selective fading channel and are received by the AP. We assume that the channel, with length L, is invariable for a whole frame, and perfectly known by the receiver. Therefore, the ith received symbol at the AP can be expressed as

$$\begin{aligned} y_i = \begin{bmatrix} y_{i,1} \\ y_{i,2} \\ \vdots \\ y_{i,M} \end{bmatrix} &= \sum_{k=1}^{K} \mathbf{H}_k \mathbf{s}^{(k)} x_{\mathrm{i}}^{(k)} \\ &= \left[\mathbf{H}_1 \quad \mathbf{H}_2 \quad \cdots \quad \mathbf{H}_k \right] \begin{bmatrix} \mathbf{s}^{(1)} & 0 & \cdots & 0 \\ 0 & \mathbf{s}^{(2)} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{s}^{(K)} \end{bmatrix} \begin{bmatrix} x_{\mathrm{i}}^{(1)} \\ x_{\mathrm{i}}^{(2)} \\ \vdots \\ x_{\mathrm{i}}^{(k)} \end{bmatrix} \end{aligned} \tag{4}$$

where $\mathbf{H}_\mathrm{k}$ is the convolution matrix for the channel of the kth sensor node, and

$$\mathbf{H}_k = \begin{bmatrix} h_{k,0} & 0 & \cdots \\ h_{k,1} & h_{k,0} & \\ \vdots & \vdots & \ddots \\ h_{k,L-1} & h_{k,L-2} & \\ 0 & h_{k,L-1} & \\ \vdots & \vdots & \ddots \\ 0 & 0 & \end{bmatrix} = circ(\mathbf{h}_k) \in \mathfrak{R}^{M \times M} \tag{5}$$

The received signal in one frame can be expressed as

$$\mathbf{y} = \sum_{k=1}^{K} \mathbf{H}_k \mathbf{S}^{(k)} \mathbf{x}_k + \mathbf{n} \tag{6}$$

where

$$\mathbf{S}^{(k)} = \begin{bmatrix} \mathbf{s}^{(k)} & 0 & \cdots & 0 \\ 0 & \mathbf{s}^{(k)} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{s}^{(k)} \end{bmatrix} \in \mathfrak{R}^{MN_c \times N_c} \tag{7}$$

is the spreading matrix of the kth sensor node, and $\mathbf{n}$ is the additive white Gaussian noise. $\mathbf{y} = [\mathbf{y}_1^\mathrm{T}, \ \mathbf{y}_2^\mathrm{T}, \ \ldots, \ \mathbf{y}_{N_c}^\mathrm{T}] \in \mathfrak{R}^{MN_c}$ is the received signal at the AP.

For simplicity, Eq. (6) can be rewritten as

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{n} \tag{8}$$

where $\mathbf{x} = [\mathbf{x}_1^\mathrm{T}, \ \mathbf{x}_2^\mathrm{T}, \ \ldots, \ \mathbf{x}_K^\mathrm{T}] \in \mathfrak{R}^{KN_c}$ is a stacked vector containing all the symbols from all Ksensor nodes in one frame, and the measurement matrix of the transmitted signal $\mathbf{A} = [circ(\mathbf{h}_1)\mathbf{S}^{(1)}, \ circ(\mathbf{h}_2)\mathbf{S}^{(2)}, \ \ldots, \ circ(\mathbf{h}_K)\mathbf{S}^{(K)}] \in \mathfrak{R}^{MN_c \times KN_c}$ combines the influence of spreading matrix $\mathbf{S}^{(\mathrm{k})}$ and channel matrix $\mathbf{H}_\mathrm{k}$.

By solving problem (8) with $l_1 -$ norm minimization of (3) via the CS algorithm, the non-zero values, as well as their positions for $\mathbf{x}$, are estimated [23]. The positions with non-zero values for $\mathbf{x}$ correspond to the indexes of the active sensor nodes. The non-zero values are the data transmitted by active sensor nodes.

In this system model, $\mathbf{x}$ is a sparse vector; that is, only a few elements of $\mathbf{x}$ are non-zero because of the fact that the activity probability of each sensor node, p_a, is much less than 1. On the receiver side, the sparse vector $\mathbf{x}$ can be estimated from the under-determined Eq. (8) via CS theory. Unfortunately, the computational complexity of CS algorithms keeps increasing exponentially with the length of $\mathbf{x}$. In Eq. (8), the length of $\mathbf{x}$ is KN_c, which is too large to estimate using the CS algorithm. Therefore, to decrease complexity, the original problem can be divided into $\beta = N_c/\nu$ sub-problems [24]. The number of consecutive transmission symbols in each sub-problem is v, called the group size. The computational complexity increases with group size, but the performance of CS algorithms decreases simultaneously. Each sub-problem uses the same measurement matrix, $\mathbf{A}' \in \mathfrak{R}^{\mathrm{M}\nu \times K\nu}$, which is the sub-matrix of the original measurement matrix $\mathbf{A}$. To simplify the model, we neglect the inter-symbol interference (ISI) between adjacent sub-problems [25]. The ith ($\mathrm{i} = 1, \ 2, \ \ldots, \ \beta$) sub-problem of the original problem (8) can be expressed as

$$\mathbf{y}_\mathrm{i}' = \mathbf{A}'\mathbf{x}'_\mathrm{i} + \mathbf{n}_\mathrm{i}' \tag{9}$$

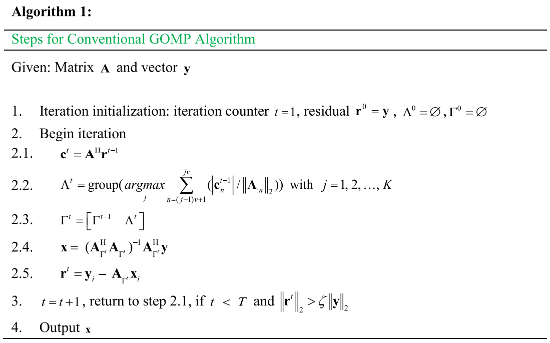
where $\mathbf{y}_i' = [\mathbf{y}_{\nu(i-1)+1}^\mathrm{T}, \ \ldots, \ \mathbf{y}_{\nu i}^\mathrm{T}] \in \mathfrak{R}^{M\nu}$ is composed of the $(\nu(i-1)+1)$th to (νi)th vectors of $\mathbf{y}$, and$\mathbf{x}_i' = [\mathbf{x}_{1((\nu(i-1)+1):\nu i)}^\mathrm{T}, \ \ldots, \ \mathbf{x}_{K((\nu(i-1)+1):\nu i)}^\mathrm{T}] \in \mathfrak{R}^{K\nu}$ is composed of $\{\mathbf{x}_{k((\nu(i-1)+1):\nu i)}^\mathrm{T}\}_{k=1,\ldots,K}$, where $\mathbf{x}_{k((\nu(i-1)+1):\nu i)}$ is a vector consisting of the $(\nu(i-1)+1)$th to (νi)th elements of $\mathbf{x}_k$. A simple example of a problem division is shown in Fig. 2, where user number $K = 5$, spreading factor $M = 3$, the number of symbols in one frame $N_c = 4$, and group size $\nu = 2$.

3 Proposed GOMP algorithms for CS-based random access model

In this section, we discuss the details of the conventional algorithm and three proposed algorithms for CS-based random access scheme. First, we provide the details of conventional GOMP algorithm and then discuss the three proposed algorithms such as the improved GOMP-LS algorithm for high-reliability applications, WIGOMP algorithm for low-latency applications, and WIGOMP-ICF for low complexity in detail.

3.1 Conventional GOMP algorithm procedure

In the CS-based random access system for MTC, the stacked vector $\mathbf{x}$ contains whole signals transmitted from all the sensor nodes. If a sensor node is inactive in a frame, it is seen as transmitting all zeros to the AP. Consequently, in the transmitted signal $\mathbf{x}$, we need to estimate block sparse, because the symbols belonging to a single sensor node are all non-zero or zero in a frame. Exploiting the block sparsity of $\mathbf{x}$, Majumdar and Ward [26] proposed a low-complexity CS algorithm, GOMP, as shown in Algorithm 1. In GOMP, the indexes of the entire group containing the highest correlation to residual are selected (Step 2.2) in each iteration, instead of only one index with the highest correlations selected in OMP. In Algorithm 1, K is the total number of groups, and ν is the number of elements in one group. The iteration stop criterion is decided by maximal iteration time T and minimum residual coefficient ζ. $\mathbf{A}_{:n}$ is the nth column of $\mathbf{A}$. $\mathbf{A}_{\Gamma^t}$ is a sub-matrix of $\mathbf{A}$ containing only the columns of $\mathbf{A}$ with indexes in Γ^t. As mentioned before, in a CS-based random access system, to decrease the complexity, the complicated problem in Eq. (8) is divided into several sub-problems, as shown in Eq. (9). Each sub-problem, $\mathbf{x}_i^{'}$, which contains a part of the elements in $\mathbf{x}$, is also block sparse. Solving each sub-problem via GOMP, the signal transmitted by sensor nodes is estimated by the AP.



Algorithm 1:

Steps for Conventional GOMP Algorithm

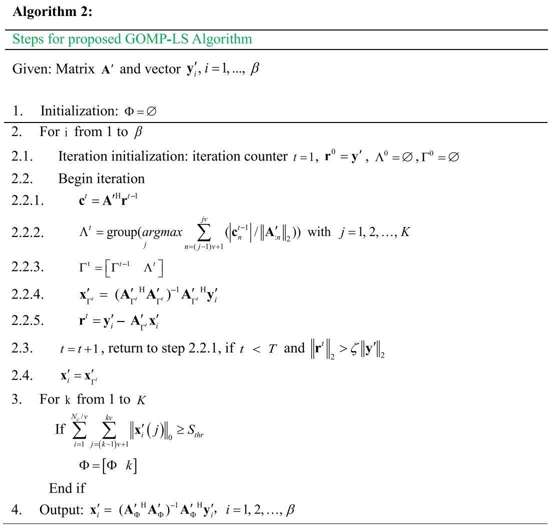
```
Given: Matrix $\mathbf{A}$ and vector $\mathbf{y}$

1.     Iteration initialization: iteration counter $t=1$, residual $\mathbf{r}^0=\mathbf{y}$, $\Lambda^0=\varnothing, \Gamma^0=\varnothing$
2.     Begin iteration
2.1.       $\mathbf{c}^t=\mathbf{A}^{\mathrm{H}}\mathbf{r}^{t-1}$
2.2.       $\Lambda^t=\text{group}(\underset{j}{argmax}\sum_{n=(j-1)v+1}^{jv}(|\mathbf{c}_n^{t-1}|/\|\mathbf{A}_{:n}\|_2))$ with $j=1,2,\ldots,K$
2.3.       $\Gamma^t=\left[\Gamma^{t-1}\quad \Lambda^t\right]$
2.4.       $\mathbf{x}=(\mathbf{A}_{\Gamma^t}^{\mathrm{H}}\mathbf{A}_{\Gamma^t})^{-1}\mathbf{A}_{\Gamma^t}^{\mathrm{H}}\mathbf{y}$
2.5.       $\mathbf{r}^t=\mathbf{y}_i-\mathbf{A}_{\Gamma^t}\mathbf{x}_i$
3.     $t=t+1$, return to step 2.1, if $t<T$ and $\|\mathbf{r}^t\|_2>\zeta\|\mathbf{y}\|_2$
4.     Output $\mathbf{x}$
```

3.2 Proposed GOMP-LS algorithm procedure for high-reliability applications

Considering the link performance, we provide the GOMP-LS algorithm. The solution to each sub-problem solved by GOMP may be different from the data transmitted by sensor nodes because of fading or noise in the wireless communications channel. The sub-problem diversity gain can be achieved because of the fact that the positions of non-zero solutions in each sub-problem are the same. Therefore, the GOMP-LS algorithm is proposed to improve the BER performance for high-reliability applications, as shown in Fig. 3. Similar to conventional GOMP for CS-based random access, on the receiver side, the problem is divided into several sub-problems, and each sub-problem is solved by GOMP. After these processes, the support sets of all the sub-problems are combined using equal gain combining (EGC) in order to get the support set with maximum likelihood. Since $\mathbf{x}$ is sparse, according to CS theory, when the support set of $\mathbf{x}$ is known, the value of non-zero elements in $\mathbf{x}$ can be estimated using LS estimation.

In GOMP-LS, given in Algorithm 2, Φ is a set containing the indexes of active sensor nodes. In Step 7, $\mathbf{x}_i^{'}(j)$ is the jth element in $\mathbf{x}_i^{'}$. If the number of estimated non-zero elements in the frame for a sensor node is larger than S_{thr}, this sensor node is labeled active, and the index of the node will be added into Φ. $\mathbf{A}_{\Phi}^{'}$ is a sub-matrix of $\mathbf{A}^{'}$ containing only the columns of $\mathbf{A}^{'}$ with indexes of the sensor nodes in Φ. Other variants are similar to the expressions in Algorithm 1.



Algorithm 2:

Steps for proposed GOMP-LS Algorithm

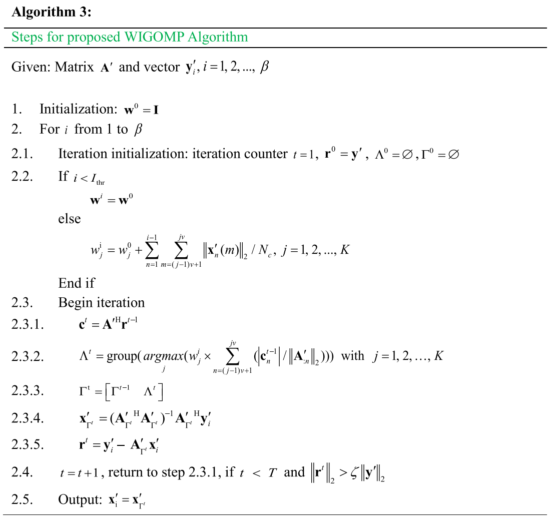
```
Given: Matrix $\mathbf{A}'$ and vector $\mathbf{y}_i'$, $i=1,\ldots,\beta$

1.     Initialization: $\Phi=\varnothing$
2.     For i from 1 to $\beta$
2.1.       Iteration initialization: iteration counter $t=1$, $\mathbf{r}^0=\mathbf{y}'$, $\Lambda^0=\varnothing, \Gamma^0=\varnothing$
2.2.       Begin iteration
2.2.1.         $\mathbf{c}^t=\mathbf{A}'^{\mathrm{H}}\mathbf{r}^{t-1}$
2.2.2.         $\Lambda^t=\text{group}(\underset{j}{argmax}\sum_{n=(j-1)v+1}^{jv}(|\mathbf{c}_n^{t-1}|/\|\mathbf{A}'_{:n}\|_2))$ with $j=1,2,\ldots,K$
2.2.3.         $\Gamma^{\mathrm{t}}=\left[\Gamma^{t-1}\quad \Lambda^t\right]$
2.2.4.         $\mathbf{x}'_{\Gamma^t}=(\mathbf{A}'_{\Gamma^t}{}^{\mathrm{H}}\mathbf{A}'_{\Gamma^t})^{-1}\mathbf{A}'_{\Gamma^t}{}^{\mathrm{H}}\mathbf{y}'_i$
2.2.5.         $\mathbf{r}^t=\mathbf{y}'_i-\mathbf{A}'_{\Gamma^t}\mathbf{x}'_i$
2.3.       $t=t+1$, return to step 2.2.1, if $t<T$ and $\|\mathbf{r}^t\|_2>\zeta\|\mathbf{y}'\|_2$
2.4.       $\mathbf{x}'_i=\mathbf{x}'_{\Gamma^t}$
3.     For k from 1 to $K$
           If $\sum_{i=1}^{N_c/v}\sum_{j=(k-1)v+1}^{kv}\|\mathbf{x}'_i(j)\|_0\geq S_{thr}$
               $\Phi=\left[\Phi\quad k\right]$
           End if
4.     Output: $\mathbf{x}'_i=(\mathbf{A}'_{\Phi}{}^{\mathrm{H}}\mathbf{A}'_{\Phi})^{-1}\mathbf{A}'_{\Phi}{}^{\mathrm{H}}\mathbf{y}'_i$, $i=1,2,\ldots,\beta$
```

3.3 Proposed WIGOMP algorithm procedure for low-latency applications

In GOMP-LS, we combine the support set of all the sub-problems to improve link performance, but additional latency is simultaneously introduced. To satisfy the requirements of low-latency applications in the next-generation mobile communications, WIGOMP, another improved GOMP algorithm is proposed, as shown in Fig. 4. In this algorithm, link performance is improved by adjusting weight without introducing additional latency. The solution can be directly output, once the sub-problem is solved, and there is no need to wait until all the sub-problems are solved, as with GOMP-LS. The position of the non-zero elements in the previous sub-problems can be employed to improve estimation accuracy for posterior sub-problems. Therefore, a critical signal can be transmitted in the posterior symbols in a frame, because the incorrect estimation probability of the symbols in the posterior sub-problems is smaller.

The pseudo code of WIGOMP is shown in Algorithm 3, where $\mathbf{w}^i$ is the weight vector in the ith iteration, and w_j^i is the jth element in $\mathbf{w}^i$. All the weights are initialized to 1 in Step 1. Because of the iterations, to restrain the influence of errors occurring previously, in Step 2.2, we keep the weights in the previous $(I_{\text{thr}} - 1)$ sub-problems equal to 1. From the $I_{\text{thr}}{}^{\text{th}}$ sub-problem, the weights in the i^{th} sub-problem are adjusted by the sum of the 2-norm of solutions in the previous $(i - 1)$ sub-problems.



```
Algorithm 3:
Steps for proposed WIGOMP Algorithm
Given: Matrix $\mathbf{A}'$ and vector $\mathbf{y}'_i, i=1,2,...,\beta$

1.      Initialization: $\mathbf{w}^0 = \mathbf{I}$
2.      For $i$ from 1 to $\beta$
2.1.        Iteration initialization: iteration counter $t=1$, $\mathbf{r}^0 = \mathbf{y}'$, $\Lambda^0 = \varnothing, \Gamma^0 = \varnothing$
2.2.        If $i < I_{\text{thr}}$
                $\mathbf{w}^i = \mathbf{w}^0$
            else
                $w_j^{\text{i}} = w_j^0 + \sum_{n=1}^{i-1} \sum_{m=(j-1)v+1}^{jv} \|\mathbf{x}'_n(m)\|_2 / N_c,\ j=1,2,...,K$
            End if
2.3.        Begin iteration
2.3.1.          $\mathbf{c}^t = \mathbf{A}'^{\text{H}}\mathbf{r}^{t-1}$
2.3.2.          $\Lambda^t = \text{group}(\underset{j}{argmax}(w_j^i \times \sum_{n=(j-1)v+1}^{jv} (|\mathbf{c}_n^{t-1}| / \|\mathbf{A}'_{:n}\|_2)))$ with $j=1,2,\ldots,K$
2.3.3.          $\Gamma^{\text{t}} = \left[\Gamma^{t-1} \quad \Lambda^t\right]$
2.3.4.          $\mathbf{x}'_{\Gamma^t} = (\mathbf{A}'_{\Gamma^t}{}^{\text{H}}\mathbf{A}'_{\Gamma^t})^{-1}\mathbf{A}'_{\Gamma^t}{}^{\text{H}}\mathbf{y}'_i$
2.3.5.          $\mathbf{r}^t = \mathbf{y}'_i - \mathbf{A}'_{\Gamma^t}\mathbf{x}'_i$
2.4.        $t = t+1$, return to step 2.3.1, if $t < T$ and $\|\mathbf{r}^t\|_2 > \zeta\|\mathbf{y}'\|_2$
2.5.        Output: $\mathbf{x}'_{\text{i}} = \mathbf{x}'_{\Gamma^t}$
```

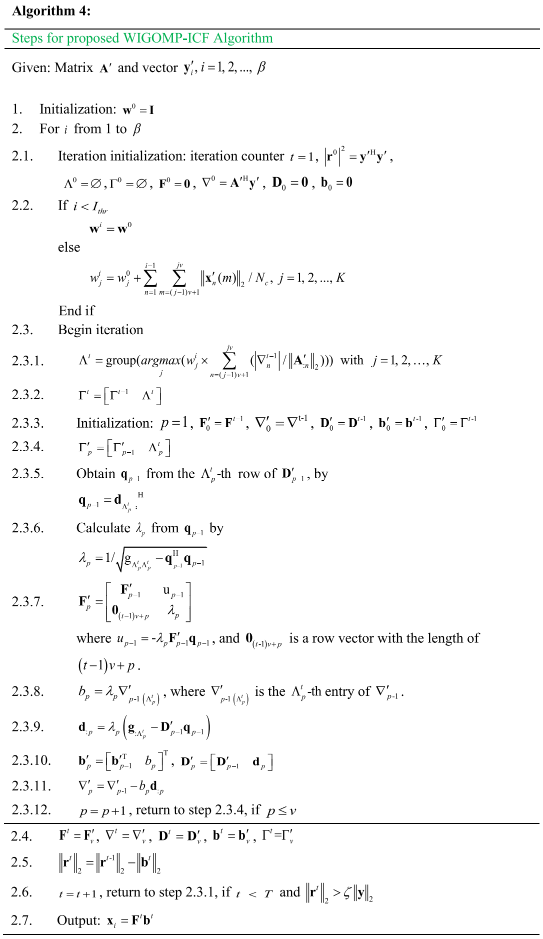
3.4 Proposed low-complexity WIGOMP-ICF algorithm procedure

In OMP and its extending algorithms, including GOMP and WIGOMP, the computational complexity grows exponentially with the number of non-zero elements in the sparse solution. To solve this problem by avoiding matrix inversions, efficient implementations of OMP based on inverse Cholesky factorization are proposed [27]. Therefore, based on WIGOMP, a low-complexity algorithm called WIGOMP-ICF is proposed by employing inverse Cholesky factorization, as shown in Algorithm 4.

To calculate $(\mathbf{A}'_{\Gamma^t}{}^{\text{H}}\mathbf{A}'_{\Gamma^t})^{-1}$ in Step. 2.3.4 of Algorithm 3, we assume $\mathbf{G}_{\Gamma^t} = \mathbf{A}'_{\Gamma^t}{}^{\text{H}}\mathbf{A}'_{\Gamma^t}$. The inverse Cholesky factor $\mathbf{G}_{\Gamma^t}$ is $\mathbf{F}_t$, which satisfies

$$\mathbf{F}_t\mathbf{F}_t{}^{\text{H}} = \mathbf{G}_{\Gamma^t}{}^{-1} = \left(\mathbf{A}'_{\Gamma^t}{}^{\text{H}}\mathbf{A}'_{\Gamma^t}\right)^{-1} \tag{10}$$

In [28], an efficient algorithm is proposed to compute $\mathbf{F}_t$ from $\mathbf{F}_{t\text{ - }1}$ iteratively.



```
Algorithm 4:
Steps for proposed WIGOMP-ICF Algorithm
Given: Matrix $\mathbf{A}'$ and vector $\mathbf{y}'_i, i=1,2,...,\beta$

1.      Initialization: $\mathbf{w}^0 = \mathbf{I}$
2.      For $i$ from 1 to $\beta$
2.1.        Iteration initialization: iteration counter $t=1$, $|\mathbf{r}^0|^2 = \mathbf{y}'^{\text{H}}\mathbf{y}'$,
            $\Lambda^0 = \varnothing, \Gamma^0 = \varnothing$, $\mathbf{F}^0 = \mathbf{0}$, $\nabla^0 = \mathbf{A}'^{\text{H}}\mathbf{y}'$, $\mathbf{D}_0 = \mathbf{0}$, $\mathbf{b}_0 = \mathbf{0}$
2.2.        If $i < I_{thr}$
                $\mathbf{w}^i = \mathbf{w}^0$
            else
                $w_j^i = w_j^0 + \sum_{n=1}^{i-1} \sum_{m=(j-1)v+1}^{jv} \|\mathbf{x}'_n(m)\|_2 / N_c,\ j=1,2,...,K$
            End if
2.3.        Begin iteration
2.3.1.          $\Lambda^t = \text{group}(\underset{j}{argmax}(w_j^i \times \sum_{n=(j-1)v+1}^{jv} (|\nabla_n^{t-1}| / \|\mathbf{A}'_{:n}\|_2)))$ with $j=1,2,\ldots,K$
2.3.2.          $\Gamma^t = \left[\Gamma^{t-1} \quad \Lambda^t\right]$
2.3.3.          Initialization: $p=1$, $\mathbf{F}'_0 = \mathbf{F}^{t-1}$, $\nabla'_0 = \nabla^{\text{t-1}}$, $\mathbf{D}'_0 = \mathbf{D}^{t\text{-}1}$, $\mathbf{b}'_0 = \mathbf{b}^{t\text{-}1}$, $\Gamma'_0 = \Gamma^{t\text{-}1}$
2.3.4.          $\Gamma'_p = \left[\Gamma'_{p-1} \quad \Lambda_p^t\right]$
2.3.5.          Obtain $\mathbf{q}_{p-1}$ from the $\Lambda_p^t$-th row of $\mathbf{D}'_{p-1}$, by
                $\mathbf{q}_{p-1} = \mathbf{d}_{\Lambda_p^t\,:}{}^{\text{H}}$
2.3.6.          Calculate $\lambda_p$ from $\mathbf{q}_{p-1}$ by
                $\lambda_p = 1/\sqrt{g_{\Lambda_p^t\Lambda_p^t} - \mathbf{q}_{p-1}^{\text{H}}\mathbf{q}_{p-1}}$
2.3.7.          $\mathbf{F}'_p = \begin{bmatrix} \mathbf{F}'_{p-1} & \text{u}_{p-1} \\ \mathbf{0}_{(t-1)v+p} & \lambda_p \end{bmatrix}$
                where $u_{p-1} = \text{-}\lambda_p\mathbf{F}'_{p-1}\mathbf{q}_{p-1}$, and $\mathbf{0}_{(t\text{-}1)v+p}$ is a row vector with the length of
                $(t-1)v+p$.
2.3.8.          $b_p = \lambda_p\nabla'_{p\text{-}1\,(\Lambda_p^t)}$, where $\nabla'_{p\text{-}1\,(\Lambda_p^t)}$ is the $\Lambda_p^t$-th entry of $\nabla'_{p\text{-}1}$.
2.3.9.          $\mathbf{d}_{:p} = \lambda_p\left(\mathbf{g}_{:\Lambda_p^t} - \mathbf{D}'_{p-1}\mathbf{q}_{p-1}\right)$
2.3.10.         $\mathbf{b}'_p = \left[\mathbf{b}'^{\text{T}}_{p-1} \quad b_p\right]^{\text{T}}$, $\mathbf{D}'_p = \left[\mathbf{D}'_{p-1} \quad \mathbf{d}_p\right]$
2.3.11.         $\nabla'_p = \nabla'_{p\text{-}1} - b_p\mathbf{d}_{:p}$
2.3.12.         $p = p+1$, return to step 2.3.4, if $p \le v$
2.4.        $\mathbf{F}^t = \mathbf{F}'_v$, $\nabla^t = \nabla'_v$, $\mathbf{D}^t = \mathbf{D}'_v$, $\mathbf{b}^t = \mathbf{b}'_v$, $\Gamma^t = \Gamma'_v$
2.5.        $\|\mathbf{r}^t\|_2 = \|\mathbf{r}^{t\text{-}1}\|_2 - \|\mathbf{b}^t\|_2$
2.6.        $t = t+1$, return to step 2.3.1, if $t < T$ and $\|\mathbf{r}^t\|_2 > \zeta\|\mathbf{y}\|_2$
2.7.        Output: $\mathbf{x}_i = \mathbf{F}^t\mathbf{b}^t$
```

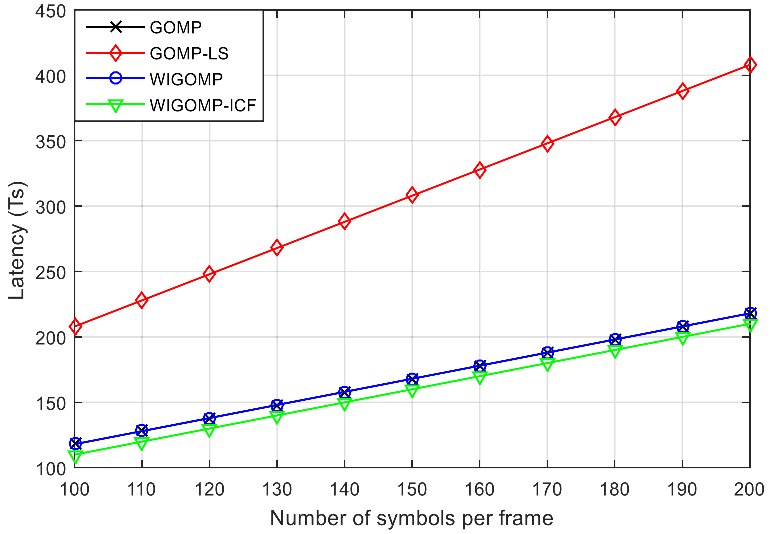


Fig. 7 Number of symbols per frame versus latency

4 Performance evaluation of proposed CS-based random access model

In this section, we describe the simulation results of the performance of the proposed three algorithms. To evaluate the link performance of the proposed algorithms, Monte Carlo simulations with different parameters are performed. We consider scenarios with $K = 128$ sensor nodes. In a frame, there are $N_c = 120$ symbols, with binary phase shift keying (BPSK) modulation and spread via pseudo-noise (PN) sequence with length $M = 32$.

4.1 Simulation results and discussion

4.1.1 Link performance evaluation

Figure 5 shows the BER performance of GOMP and GOMP-LS algorithms under various group sizes, v, with activity probability $p_a = 0.02$. First, in both GOMP and GOMP-LS, BER is decreasing while group size v is increasing, which is due to the exploitation of block sparsity. When $E_b/N_0 = 16$, the BER of GOMP decreases from 2.2×10^{-4} to 3×10^{-5}, with group sizes increasing from 2 to 8. Compared with GOMP, GOMP-LS obviously achieves better BER performance. Even the BER of GOMP-LS with $v = 2$ is lower than the BER of GOMP with $v = 8$; i.e., to meet the same BER requirements, we can use GOMP-LS with a smaller group size, instead of GOMP with a larger group size, to decrease the computational complexity. In addition, we note that when the group size is decreased from 8 to 2, the BER increase in GOMP is much larger than in GOMP-LS. Since in GOMP-LS, we consider the relationship between each sub-problem that when the group size is smaller, the number of sub-problems is larger, which introduces more sub-problem diversity gain in GOMP-LS. In addition, we compare the BER performance of the proposed algorithms with GOMP when group size v equals to 8.

Figure 6 shows the performance of GOMP-LS is the best, since it exploits the relationship between sub-problems and gets more diversity gain, which greatly decreases the BER. The performance of WIGOMP is better than GOMP, because the diversity gain is achieved in the posterior sub-problem by introducing the weights of each user according to the solution of the previous sub-problem. The performance of WIGOMP is worse than GOMP-LS, because it can only improve the BER performance of posterior sub-problems, whereas, in GOMP-LS, the performance of all the sub-problems is improved. When $E_b/N_0 = 16$ and $p_a = 0.04$, the BER of

Table 1 Comparison of computational complexity and latency

Algorithm	Computational complexity	Latency
GOMP	$\beta(MKv^2 + 2MKp_av^2 + 2MK^2{p_a}^2v^3 + K^3{p_a}^3v^3 + Kv + K^2{p_a}^2v^2)$	$N_cT_s + T_0 + vT_s$
GOMP-LS	$\beta(MKv^2 + 3MKp_av^2 + 4MK^2{p_a}^2v^3 + 2K^3{p_a}^3v^3 + Kv + K^2{p_a}^2v^2)$	$2N_cT_s + T_1$
WIGOMP	$\beta(MKv^2 + 2MKp_av^2 + 2MK^2{p_a}^2v^3 + K^3{p_a}^3v^3 + Kv + K^2{p_a}^2v^2 + Kp_av^2)$	$N_cT_s + T_2 + vT_s$
WIGOMP-ICF	$\beta(MKv^2 + MKp_av^2 + K^2{p_a}^2v^2 + N_cK + 2\mathrm{Kp_a}v^2 + Kv + Kp_av)$	$N_cT_s + T_3 + vT_s$

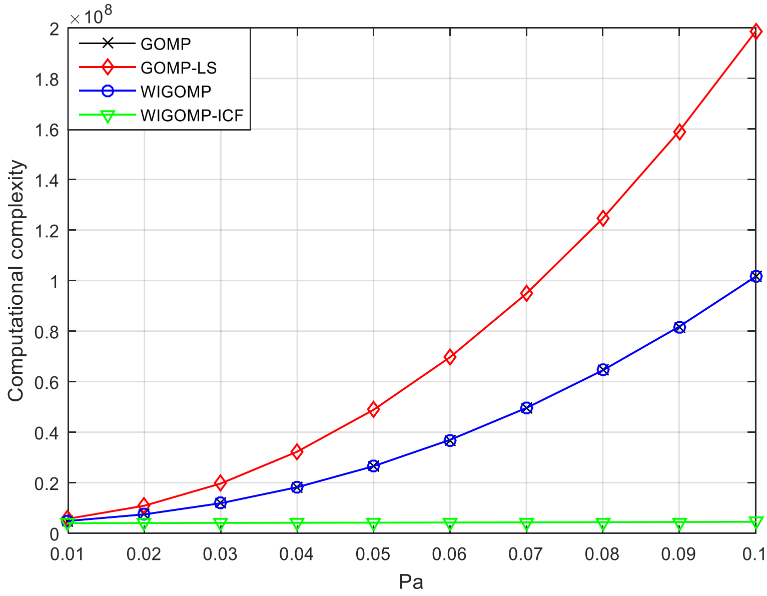


Fig. 8 Activity probability versus computational complexity, where $K = 128$, $M = 32$, $N_c = 120$, $v = 8$, and $\beta = 15$

GOMP, WIGOMP, and GOMP-LS is 1.6×10^{-4}, 9×10^{-5}, and 4×10^{-5}, respectively. Moreover, it is noted that there is more gain for BER performance using GOMP-LS or WIGOMP instead of GOMP when E_b/N_0 becomes high. Under the condition of high E_b/N_0, the number of error data becomes small, so it is easier to correct error bits by EGC or weighted iteration. Because WIGOMP-ICF only decreases the complexity of WIGOMP by avoiding the matrix inverse, the BER performance of WIGOMP-ICF, which is the same as WIGOMP, is not shown here.

4.1.2 Latency and complexity performance evaluation

In Fig. 7, the latencies varying frame lengths of the proposed algorithms are shown. Latency linearly increases with an increase in the number of symbols per frame. The latency of GOMP and GOMP-LS is almost the same, and the latency of WIGOMP-ICF is 7% lower than GOMP. The latency of GOMP-LS is almost twice that of the other algorithms.

In Table 1, we compare the computational complexity and latency of GOMP and the proposed algorithms. The computational complexity is the number of complex

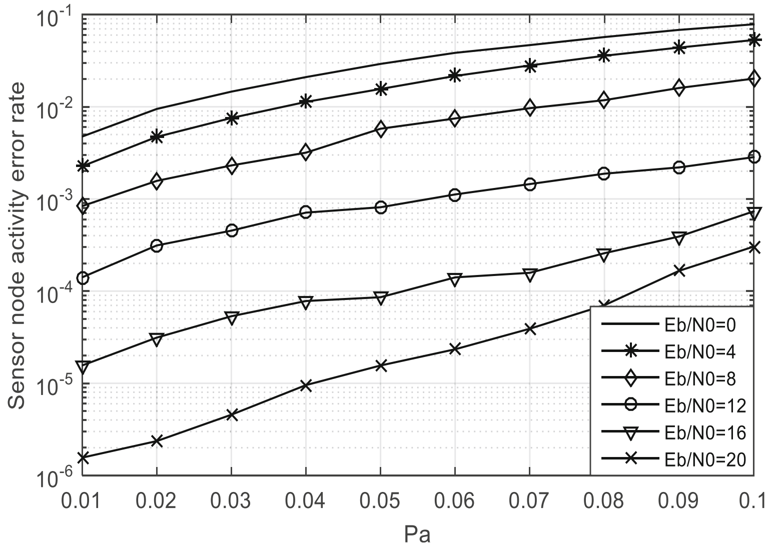


Fig. 9 Activity probability versus sensor node activity error rate with $v = 8$

multiplications, and latency is the delay from input of the first symbol to output of the last symbol. Latency in the algorithms is represented by the symbol duration, T_s, and the processing time of various computational complexities are expressed as follows:

$$\begin{cases} T_0 = T\left(MKv^2 + 2MKp_a v^2 + 2MK^2 {p_a}^2 v^3 + K^3 {p_a}^3 v^3 + Kv + K^2 {p_a}^2 v^2\right) \\ T_1 = T\left(MKv^2 + 3MKp_a v^2 + 4MK^2 {p_a}^2 v^3 + 2K^3 {p_a}^3 v^3 + Kv + K^2 {p_a}^2 v^2\right) \\ T_2 = T\left(MKv^2 + 2MKp_a v^2 + 2MK^2 {p_a}^2 v^3 + K^3 {p_a}^3 v^3 + Kv + K^2 {p_a}^2 v^2 + Kp_a v^2\right) \\ T_3 = T\left(MKv^2 + MKp_a v^2 + K^2 {p_a}^2 v^2 + N_c K + 2Kp_a v^2 + Kv + Kp_a v\right) \end{cases} \tag{11}$$

Figure 8 shows the computational complexity varying the activity probability of the four algorithms in Table 1. The value of the parameters in Table 1 are as follows: number of sensor nodes$K = 128$, spreading factor $M = 32$, number of symbols per frame $N_c = 120$, group size $v = 8$, and the number of sub-problems $\beta = N_c/v = 15$. The computational complexity of GOMP and WIGOMP are almost the same, which is larger than WIGOMP-ICF and smaller than GOMP-LS. With the increasing activity probability, the computational complexity of WIGOMP-ICF is almost invariable, whereas the computational complexity of other algorithms increases exponentially, especially GOMP-LS. When $p_a = 0.04$, the computational complexity of GOMP is around five times that of WIGOMP-ICF, and the computational complexity of GOMP-LS is around eight times of WIGOMP-ICF's. To compare the latency of the proposed algorithms with GOMP, we assume the processing time for various computational complexities are $T_0 = 10T_s$, $T_1 = 16T_s$, $T_2 = 10T_s$, and $\mathrm{T}_3 = 2T_s$.

4.1.3 Activity error rate performance evaluation

Figure 9 shows the sensor node activity error rate with a varying activity probability for group size $v = 8$. In CS-based random access, both the activity of the sensor nodes and the transmitted data are detected on the receiver side. The sensor node activity error rate decreases while E_b/N_0 is increasing. Sensor node activity error includes a case where an active node is detected as inactive, and vice versa. Moreover, the high-activity probability results in more detection errors. In terms of CS theory, the high-activity probability corresponds to the overall high sparsity, and the sparsity is destroyed gradually with the activity probability increasing, which leads to performance degradation in CS algorithms. At $E_b/N_0 = 12$, the sensor node activity error rate increases from 1.5×10^{-4} to 2×10^{-3} when activity probability increases from 0.01 to 0.1.

5 Conclusions

In this paper, we employ CS-based random access scheme with unique identification sequences for MTC by considering the sparsity of devices activity, which efficiently reduces the signaling overhead and guarantees the link performance. Three improved GOMP algorithms are proposed for various applications in the next generation of mobile communication systems. Simulation results show that the BER of GOMP-LS is the lowest. Under the conditions of $E_b/N_0 = 16$ dB and $p_a = 0.04$, the BER of GOMP is 1.6×10^{-4}, while the BER of GOMP-LS decreases to 4×10^{-5}. However, the latency of GOMP-LS is almost as twice as that of GOMP. Therefore, GOMP-LS is suitable for high-reliability applications. To achieve relatively high reliability without increasing latency, WIGOMP is proposed. While keeping the same computational complexity and latency as GOMP, under the same conditions mentioned above, the BER of WIGOMP decreases to 9×10^{-5}, which is better than GOMP, but worse than GOMP-LS. Moreover, WIGOMP-ICF, which has the same BER performance as WIGOMP, is proposed, combining WIGOMP with matrix decomposition, and the latency of this algorithm is 7% lower than GOMP. Hence, WIGOMP is more suitable for low latency applications. In addition, we analyze the sensor node activity performance of the proposed CS-based random access scheme which results in better activity error rate.

Abbreviation

AP: Access point; BER: Bit error rate; BPSK: Binary phase shift keying; CP: Chaining pursuit; CS: Compressive sensing; EGC: Equal gain combining; GOMP: Group orthogonal matching pursuit; GOMP-ICF: Group orthogonal matching pursuit with inverse Cholesky factorization; GOMP-LS: Group orthogonal matching pursuit least square; IoT: Internet of things; LTE: Long - term evolution; MP: Matching pursuit; MTC: Machine type communication; MUD: Multi-user detection; OMP: Orthogonal matching pursuit; PN: Pseudo -noise; QoS: Quality of service; WIGOMP: Weighted iteration group orthogonal matching pursuit

Acknowledgements

This research was supported by the MSIT (Ministry of Science, ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2019-2014-1-00729) supervised by the IITP (Institute of Information & communications Technology Planning & Evaluation).

Authors' contributions

YH proposed the three improved GOMP algorithms considering the tradeoff between link performance and latency in order to support diverse applications requirements. She also employed the promising CS-based random access scheme to reduce the signal overhead and latency. Moreover, she wrote some method aspects of the manuscript and also performed the simulations. WC drew the random access scenario of MTC and also made all the block diagrams of algorithms. Moreover, she wrote some sections of the manuscript and also corrected all the English mistakes in the overall manuscript. IA solved the reviewers' comments in the Response Letter and improved the readability and fluency of the manuscript. Also, he modified the abstract, introduction, conclusions parts, and corrected the sequence of the sections in the manuscript as well. In addition, he corrected technical issues related to the manuscript and proposed schemes as well. LS by discussion helped in proposing the CS-based random access scheme which has the capability of reducing signal overhead and latency when providing reliable communications in the MTC system. Moreover, she wrote some technical equations which help in the analysis of the proposed scheme. KC is the technical leader of this manuscript. He suggested all the technical issues for the proposed GOMP algorithms and also for simulation aspects. In addition, he corrected all the simulation methodology of this manuscript and also corrected all the mistakes in the simulation environment as well as in the structure of overall manuscript. All authors read and approved the final manuscript.

Funding
This research was supported by the MSIT (Ministry of Science, ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2019-2014-1-00729) supervised by the IITP (Institute for Information and Communications Technology Promotion).

Availability of data and materials
Not applicable

Competing interests
The authors declare that they have no competing interests.

Received: 17 September 2016 Accepted: 4 July 2019
Published online: 30 July 2019

References

1. M. Hasan, E. Hossain, D. Niyato, Random access for machine-to-machine communication in LTE-advanced networks: issues and approaches. *IEEE Communications Magazine*. **51**(6), 86–93 (2013)
2. KD Lee, S Kim, B Yi, Throughput comparison of random access methods for M2M service over LTE networks, in Proc. of IEEE Globecom Workshops, (Houston, TX, Dec. 2011), pp. 373–377.
3. D. Niyato, P. Wang, D.I. Kim, Performance modeling and analysis of heterogeneous machine type communications. *IEEE Transactions on Wireless Communications*. **13**(1), 2836–2849 (2014)
4. S.Y. Lien, K.C. Chen, Y. Lin, Toward ubiquitous massive accesses in 3GPP machine-to-machine communications. *IEEE Communications Magazine*. **49**(4), 66–74 (2011)
5. K Au, L Zhang, P Zhu, Uplink contention based SCMA for 5G radio access, in Proc. of IEEE Globecom Workshops, (Austin, TX, Dec. 2014), pp. 900-905.
6. A. Laya, L. Alonso, J.A. Zarate, Is the random access channel of LTE and LTE-A suitable for M2M communications? A survey of alternatives. *IEEE Communications Surveys & Tutorials*. **16**(1), 4–16 (2014)
7. G.P. Fettweis, The tactile internet: Applications and challenges. *IEEE Vehicular Technology Magazine*. **9**(1), 64–70 (2014)
8. C. Bockelmann, H.F. Schepker, Compressive sensing based multi-user detection for machine-to-machine communication. *Transaction on Emerging Telecommunications Technologies*. **24**(44), 384–400 (2013)
9. F Monsees, M Woltering, C Bockelmann, A Dekorsy, Compressive sensing multi-user detection for multicarrier systems in sporadic machine type communication, in Proc. of IEEE 81st Vehicular Technology Conference (VTC Spring), (Glasgow, UK, May 2015), pp. 1-5.
10. D.L. Donoho, Compressed sensing. *IEEE Transactions on Information Theory*. **52**(4), 1289–1306 (2006)
11. E.J. Candes, M.B. Wakin, An introduction to compressive sampling. *IEEE Signal Processing Magazine*. **25**(2), 21–30 (2008)
12. AT Abebe, CG Kang, Compressive sensing-based random access with multiple-sequence spreading for MTC, in Proc. of IEEE Globecom Workshops, (San Diego, CA, Dec. 2015), pp. 1-6.
13. H.F. Schepker, C. Bockelmann, A. Dekorsy, Efficient detectors for joint compressed sensing detection and channel decoding. *IEEE Transactions on Communications*. **63**(6), 2249–2260 (2015)
14. HF Schepker, C Bockelmann, Coping with CDMA asynchronicity in compressive sensing multi-user detection, in Proc. of IEEE 77th Vehicular Technology Conference, (Dresden, German, Jun. 2013), pp. 1-5.
15. HF Schepker, A Dekorsy, Compressive sensing multi-user detection with block-wise orthogonal least squares, in Proc. of IEEE 75th Vehicular Technology Conference, (Yokohama, Japan, May 2012), pp. 1-5.
16. HF Schepker, Sparse multi-user detection for CDMA transmission using greedy algorithms, in Proc. of 8th International Symposium on Wireless Communication Systems, (Aachen, German, Nov. 2011), pp. 291-295.
17. S.G. Mallat, Z. Zhang, Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*. **41**(12), 3397–3415 (1993)
18. J.A. Tropp, A.C. Gilbert, Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Transactions on Information Theory*. **53**(12), 4655–4666 (2007)
19. X Wang, Z Zhao, N Zhao, H Zhang, On the application of compressed sensing in communication networks, in Proc. of 5th International ICST Conference on Communications and Networking in China (CHINACOM), (Beijing, China, Aug. 2010), pp. 1-7.
20. MA Khajehnejad, J Yoo, A Anandkumar, B Hassibi, Summary based structures with improved sublinear recovery for compressed sensing, in Proc. of IEEE International Symposium on Information Theory Proceedings (ISIT), (St. Petersburg, Russia, Aug. 2011), pp. 1427-1431.
21. Y Beyene, C Boyd, K Ruttik, C Bockelmann, R Jäntti, Compressive sensing for MTC in new LTE uplink multi-user random access channel, in Proc. of IEEE International Conference on Green Innovation For African Renaissance (AFRICON), (Addis Ababa, Abyssinia, Sept. 2015), pp. 1-5.
22. J Liu, HY Cheng, CC Liao, AYA Wu, Scalable compressive sensing-based multi-user detection scheme for Internet-of-Things applications, in Proc. of IEEE Workshop on Signal Processing Systems (SiPS), (Hangzhou, China, Oct. 2015), pp. 1-6.
23. H. Zhu, G.B. Giannakis, Exploiting sparse user activity in multiuser detection. *IEEE Transactions on Communications*. **59**(2), 454–465 (2011)
24. A.T. Abebe, C.G. Kang, Iterative order recursive least square estimation for exploiting frame-wise sparsity in compressive sensing-based MTC. *IEEE Communications Letters*. **20**(5), 1018–1021 (2016)
25. Y Ji, C Bockelmann, A Dekorsy, Compressed sensing based multi-user detection with modified sphere detection in machine-to-machine communications, in *Proc. of 10th International ITG Conference on Systems, Communications and Coding*, (Hamburg, Germany, Feb. 2015), pp. 1-6.
26. A. Majumdar, R.K. Ward, Fast group sparse classification. *Canadian Journal of Electrical and Computer Engineering*. **34**(4), 136–144 (2009)
27. H Zhu, G Yang, W Chen, Efficient Implementations of Orthogonal Matching Pursuit Based on Inverse Cholesky Factorization, in *Proc. of IEEE 78th Vehicular Technology Conference (VTC Fall)*, (Las Vegas, NV, Sept. 2013), pp. 1-5.
28. H. Zhu, W. Chen, B. Li, F. Gao, An improved square-root algorithm for V-BLAST based on efficient inverse Cholesky factorization. *IEEE Transactions on Wireless Communications*. **10**(1), 43–48 (2011)

6 Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.