Analysis of Channel-Aware Multi-User Resource Allocators for Correlated Rayleigh Fading

Robert Walstab, Nick Schwarzenberg, Philipp Schulz, and Gerhard Fettweis
Vodafone Chair Mobile Communications Systems, Technische Universität Dresden, Germany
{robert.walstab, nick.schwarzenberg, philipp.schulz2, gerhard.fettweis}@tu-dresden.de

***Abstract*—When employing wireless connectivity in industrial applications, reliability is indispensable. However, due to the real-time requirements of said scenarios, time diversity is not feasible. Instead, frequency diversity has to be utilized by transmitting on frequencies with different fading characteristics. But as increasing bandwidth requirements for the individual user does not scale well for many users in the presence of a limited system bandwidth, the available resources have to be allocated efficiently. In this work, we compare multi-user resource allocation algorithms utilizing each user's channel state information. The performance of the allocators is evaluated with regard to the achievable outage rate for different scenarios and different sensitivity analyses are performed. We also evaluate the algorithmic complexity of the allocators, both empirically and analytically. For evaluation, we employ analytical, spectrally correlated Rayleigh fading in order to emulate different environments. For a number of existing allocators, we propose modifications to improve the performance in regard to reliability and algorithmic complexity. The simulation results show, that the utilized allocators can be employed for a multitude of scenarios as they are suited for a large variety of application areas. Additionally, our results show that the allocation complexity of existing allocators can be reduced by up to a factor of 10 in the investigated scenario with a modified approach without sacrificing reliability.**

***Index Terms*—Radio resource scheduling, reliability, resource allocation, URLLC.**

I. Introduction

In the course of industrial automation and the Tactile Internet [1] the need for reliable communication is eminent. One feature of the fifth generation of mobile communications that might be up to this task is called ultra-reliable low-latency communications (URLLC) [2], which is defined by the 3GPP [3], [4]. URLLC focuses more on a stable low-latency connection rather than a high throughput. One obvious way to achieve these goals is multi connectivity (MC), where, as the name suggests, multiple communication links are used simultaneously [5]. It can be implemented in different ways either utilizing spectral or spatial diversity. In these ways, the information can be sent multiple times in the same time slot [6]. However, MC does not scale well for a larger number of users when having a limited system bandwidth, as the bandwidth requirements are multiplying [4]. Therefore, a different approach to utilize the limited available bandwidth is needed. In this context, [4] proposed that instead of relying on transmitting over multiple subchannels without respect to the channel's condition, the allocation of subchannels should be performed with the channel state information (CSI) in mind. The concept of the authors' approach takes the characteristics of a Rayleigh fading channel into account. In order to avoid deep fades at a user, the resources should be allocated for each time slot individually so that the limited channel bandwidth is utilized more efficiently. This approach is able to achieve reliable communication for a larger number of users [4].

This work was supported by the German Federal Ministry of Education and Research (BMBF) as part of the project "USWA - Ultra Scalable Wireless Access" under grant 16KISK256.

Multiple publications such as [7]–[9] already investigated the use of adaptive resource allocators but focused on maximizing the system throughput. However, in an industrial application, the reliability of the system is limited by the reliability of the weakest user, i.e., the user with the worst channel conditions. Therefore, rather than maximizing the system throughput, the channel quality of the weakest user has to be maximized [4], which is called max-min allocation [4], [10]. This was done in [2] and [4], where different radio resource allocation (RRA) algorithms were proposed. However, in [2] the authors only considered allocating a single resource per user and in [4] channel data of a channel measurement from only a single factory hall was used for their analyses. In order to generalize the performance of the allocators, different channel conditions and user configurations must be evaluated which is done in this work.

In this work we compare existing and novel subchannel allocators for different environments to determine under which conditions the allocators perform best. We will investigate the reliability of a network utilizing such an allocator and whether the allocator may be suited for a low-latency network for different scenarios and different requirements on the allocators.We also evaluate the computational complexity, both analytically and empirically. Additionally, we extend the allocators to better meet the requirements necessary for URLLC, both in regard to system availability and low latency.

II. System Model

A. Resource Allocation

An allocation takes place by assigning subchannel of the set $\mathcal{R}_{\text{sys}} = \{1, 2, ..., R_{\text{sys}}\}$ resources of a bandwidth B_{SC}, which has not been allocated yet and allocate it to a user, which has less than R_{user} resources allocated to them. This process is repeated until all users have been assigned R_{user} subchannels, as done in Fig. 1. As a result, all users are served in every timeslot to ensure low latency [4]. How this allocation

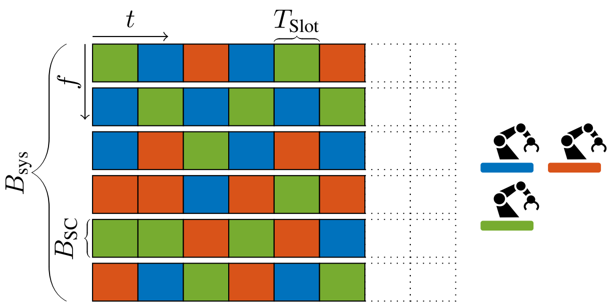

Fig. 1. Example resource allocation for $U = 3$ users where each user gets allocated $R_{\text{user}} = 2$ subchannels of bandwidth B_{SC} for each time slot T_{slot}. Illustration is based on [11].

process is executed is subject to the different allocators, which will be discussed in the next chapter. After all users have their subchannels assigned, the signal-to-noise-ratio (SNR) of every user is defined as the sum of the SNR of all allocated subchannels. If this sum is smaller than the outage threshold of the system the user experiences a user outage [4]:

$$\Pr(\text{user outage}) = \Pr\left(\sum_{r=1}^{R_{\text{user}}} |\Gamma_{\text{alloc},r,u}| < |\Gamma_{\min}|\right). \tag{1}$$

However, as all users are equally important, a user outage results in a system outage. Therefore, the outage of the system is defined as the event that the SNR of at least one user is lower than the outage power threshold [4].

B. Utilized Channel Data

While the authors in [4] investigated the allocators' performance with measured channel data, we utilize generated fading sequences. This has the benefit to be able to alter the channel conditions in order to simulate different environments which enables us to evaluate possible limitations of the allocators for certain channel properties. However, as we try to retain the characteristics of the fading channel observed in [12], the generated channel is aimed to have the same statistical properties as the measured one, at least in the frequency domain. Therefore, the approach proposed in [13] is utilized as it allows to create any number of frequency-correlated Rayleigh fading channels $\mathbf{Z}_i$. It achieves this by multiplying a vector of uncorrelated fading samples $\mathbf{W}_i$ with equal mean power, which are the i-th entries of R_{sys} uncorrelated fading sequences $\mathbf{U}_j$, with a coloring matrix $\mathbf{L}$. The sequences $\mathbf{U}_j$ can be generated using common state-of-the-art approaches, like filtering Gaussian noise with a Doppler filter as done in [13]:

$$\mathbf{W}_i = \left\{u_{i,1}, u_{i,2}, \ldots, u_{i,R_{\text{sys}}}\right\}, \tag{2}$$

$$\mathbf{Z}_i = \frac{\mathbf{L}\mathbf{W}_i}{\sigma_g}, \tag{3}$$

with σ_g being the mean power of the Gaussian random variables of the fading signals. This results in the samples having the desired correlation, while the desired correlation is represented by the covariance matrix $\mathbf{K}$:

$$\mathbf{K} = E\left(\mathbf{Z}\mathbf{Z}^H\right) = E\left(\frac{\mathbf{L}\mathbf{W}\mathbf{W}^H\mathbf{L}}{\sigma_g^2}\right) = \mathbf{L}\mathbf{L}^H. \tag{4}$$

The coloring matrix $\mathbf{L}$ is obtained by performing the eigen decomposition on the covariance matrix $\mathbf{K}$:

$$\mathbf{K} = \mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^H = \mathbf{V}\bar{\boldsymbol{\Lambda}}\bar{\boldsymbol{\Lambda}}^H\mathbf{V}^H = (\mathbf{V}\bar{\boldsymbol{\Lambda}})(\mathbf{V}\bar{\boldsymbol{\Lambda}})^H = \mathbf{L}\mathbf{L}^H, \tag{5}$$

$$\text{with } \boldsymbol{\Lambda} = \bar{\boldsymbol{\Lambda}}\bar{\boldsymbol{\Lambda}}^H, \tag{6}$$

where $\boldsymbol{\Lambda}$ denotes the matrix of eigenvalues and $\mathbf{V}$ is the matrix of eigenvectors of $\mathbf{K}$. However, it has to be noted that $\mathbf{K}$ does not represent the covariance matrix of the fading envelope but the covariance matrix of the Gaussian random variables with a Rayleigh fading envelope. Nonetheless, [14] derived the one-to-one mapping of the elements $\rho_{i,j}$ of the covariance matrix $\mathbf{K}$ of the Gaussian random variable and the elements $\bar{\rho}_{i,j}$ of the covariance matrix $\bar{\mathbf{K}}$ of the envelope [13], [14]:

$$\bar{\rho}_{i,j} = \frac{(1 + |\rho_{i,j}|)\, E_i\left(\frac{2\sqrt{|\rho_{i,j}|}}{1+|\rho_{i,j}|}\right) - \frac{\pi}{2}}{2 - \frac{\pi}{2}}, \tag{7}$$

where $E_i(\cdot)$ is the complete elliptic integral of the second order. While it is not possible to solve (7) for $\rho_{i,j}$ in closed form [15], an approximation has been provided by [14]:

$$\rho_{i,j} = \sqrt{\bar{\rho}_{i,j}}. \tag{8}$$

The main reason for generating synthetic Rayleigh fading is the ability to generate channel data with desired characteristics such as specified correlation of the subchannels. This can be achieved by modifying the aforementioned correlation matrix. More specifically, the entries $\bar{\rho}_{i,j}$ of $\bar{\mathbf{K}}$ are getting multiplied by a scaling factor α. However, as scaling the correlation coefficients would not have any effect (scaling the coefficients does not change their relation to each other), the elements $\bar{\rho}_{i,i}$ will not be multiplied and are therefore set to one:

$$\rho_{i,j} = \begin{cases} 1 & i = j, \\ \sqrt{\alpha \cdot \bar{\rho}_{i,j}} & i \neq j. \end{cases} \tag{9}$$

The parameters used for the following simulations were derived from the measured channel data of [12] and are summarized in Tab. I.

TABLE I
SYSTEM PARAMETERS

Center Frequency	f_c	3.75 GHz
System Bandwidth	B_{sys}	78.1 MHz
Subchannel Bandwidth	B_{SC}	0.195 MHz
Total Subchannels	R_{sys}	400
Slot Length	T_{slot}	1 ms
Vehicle Speed	v	$1\,\text{m}\,\text{s}^{-1}$
Vehicle Spacing	s	1 m
Maximum Doppler Frequency	f_m	12.67 Hz

III. Allocators

A. Reference

To be able to classify the performance w.r.t. the achievable outage rate of the proposed allocators, it is beneficial to have an upper bound of the achievable outage rate. This task is fulfilled by the allocator *BestImpossible*. This allocator assigns all users their preferred subchannels, i.e. the first R_{user} subchannels with the highest SNR. As this means that a single subchannel can be allocated to multiple users, this method is not realizable, hence its name. Nevertheless, this theoretical allocator will be used in future investigations as all users getting their channels with the highest SNR implies that there is no better allocation.

We also introduce two conventional allocators, which serve as a baseline for evaluating the performance of the CSI aware ones. The first one being *StaticConsecutive*. When applying this allocation technique, each user gets assigned its resources once and all assigned resources are consecutive in the frequency domain. This corresponds with the traditional method of not benefiting from frequency diversity as much as the allocators introduced later. As this technique is prone to higher outage rates for larger coherence bandwidths (having a large coherence bandwidth means that when one resource falls below the threshold there is a high chance for this being true for the adjacent channels) *StaticInterleaved* is employed to mitigate this problem. Like *StaticConsecutive* the subchannels are allocated statically. But in contrast, the resources for each user are spread across the whole available spectrum maximizing the distance between the resources of the same user and therefore benefiting from frequency diversity by decorrelating the resources in frequency.

B. Optimal max-min allocator

While the previously mentioned allocators do not require CSI and therefore do not account for the varying channel conditions, a subchannel allocation algorithm dependent on knowledge of the channel state to allocate the available resources adaptively is proposed in [2]. The authors' concept is to maximize the SNR of the weakest user at any given time slot as the goal of the allocation is not to maximize the throughput of the system but rather to ensure reliability by minimizing the risk of a system outage. This is achieved by sorting all $R_{\text{user}} \cdot U$ channel gains. Next, the ω largest entries of this sorted list are looked at to determine if a valid allocation exists. A valid allocation is achieved if every user gets allocated R_{user} unique subchannels and none of the subchannels got allocated to more than one user. Next, the Dulmage-Mendelsohn (DM) decomposition is performed which returns a subgraph of the users and the available resources that form a perfect matching [2]. If this subgraph contains all users, the allocation is valid. In this case, the resource at the ω-th position of the sorted gain list is removed from $\mathcal{R}_{\text{sys}}$. Otherwise, the window size ω, which is initialized as $\omega = U$, gets increased by one and the process will be repeated.

In [2] the user at this position was also removed, as this allocator was designed to only assign a single resource to each user. However, this is not sufficient for this work, so it has to be extended. Therefore, before removing a user from the user list $\mathcal{U}$ it has to be checked if this user has been assigned the required number of subchannels. After that, ω is again set to the number of users having not assigned the sufficient number of resources and the process is repeated until $\mathcal{U}$ is empty. In the following, this allcocator will be referred to as *Traßl*. Because this process is time and resource heavy (see Sec. IV-B) a different approach is proposed by the authors in [4].

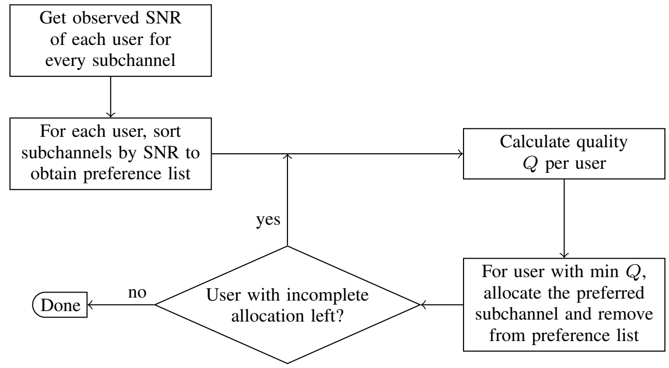

Fig. 2. Algorithm of preference based allocators based on [4].

C. Preference Based Allocators

With the aim of achieving comparable reliability while only needing a fraction of computing resources the authors of [4] suggested an additional approach.

As the SNR of the weakest user is dependent on which subchannels were allocated to it, an iterative approach is suggested. For each user u their SNRs $\Gamma(r, u, t)$ for all subchannels $r \in \mathcal{R}_{sys}$ are sorted descendingly. This resulting list will be called the user's preference list $\mathcal{P}_{u,t,i}$ with i being the current iteration. It is now used to quantify the quality metric Q for each user that is then used to decide which user gets to choose its next preferred subchannel first. After a subchannel is chosen, it will be removed from the preference list $\mathcal{P}_{u,t,i+1}$ of all users (the subchannel is no longer available). If any user does not have its needed number of allocated channels this process is repeated until all users are served. This procedure is illustrated in Fig. 2. A key part of the performance of the preference based allocators is played by the quality metric Q, as it determines the order of the users choosing their preferred subchannel. In [4], three possible functions for Q are proposed. The simplest approach of the three is called *PrefSingle* and the corresponding Q_{single} is obtained by just taking the first entry in the preference list $\mathcal{P}_{u,t,i}$ into account:

$$Q_{\text{single}} = \Gamma_{\mathcal{P}}(1). \tag{10}$$

However, this approach might not be suited when allocating multiple subchannels to a user: it does not benefit from frequency diversity, as the single largest Γ does not hold information on how well the user is performing in the whole bandwidth. Therefore, it is extended by *PrefWindow* which instead of only the first preferred subchannel does take the first L resources into account. The parameter L will be referred to as the lookahead distance of the allocator.

$$Q_{\text{Window}} = \sum_{n=1}^{L} \Gamma_{\mathcal{P}}(n). \tag{11}$$

To punish the quality metric Q of a user, who has only few good performing resources available, even more, the last preference list based allocator called *PrefGradient* proposes calculating Q by multiplying the first and the L-th SNR of $\mathcal{P}_{u,t,i}$, as this corresponds to the Gradient of the user preference list [4].

$$Q_{\text{Gradient}} = \Gamma_{\mathcal{P}}(1) \cdot \Gamma_{\mathcal{P}}(L). \tag{12}$$

In addition to the metrics proposed in [4], a new quality metric Q will be introduced in this work. It is based on *PrefWindow* with an additional weighting of the summands, such that

$$Q_{\text{Weighted}} = \sum_{n=1}^{L} w_n \Gamma_{\mathcal{P}}(n), \tag{13}$$

where w_n denotes the weights of the n-th SNR. We propose to obtain them from the correlation coefficients of the subchannels by using the correlation coefficients between the first and the n-th subchannel as the weight w_n to incorporate the CSI into the calculation of the qualitz metric Q:

$$w_n = \rho(\mathbf{P}_1, \mathbf{P}_n), \tag{14}$$

$$= \frac{1}{N-1} \sum_{i=1}^{N} \left(\frac{\mathbf{P}_{1,i} - \mu_{\mathbf{P}_1}}{\sigma_{\mathbf{P}_1}} \right) \left(\frac{\mathbf{P}_{n,i} - \mu_{\mathbf{P}_n}}{\sigma_{\mathbf{P}_n}} \right), \tag{15}$$

where N is the number of samples per subchannel obtained by the channel sounding which has to be performed for this allocator.

Among proposing an allocator aimed at offering lower outage rates we also introduce modifications of the preference based allocators published in [4] to reduce the computational effort. One way to achieve this is to limit the iterations of the preference based allocators (see Fig. 2). While [2] suggests to only allocate a single resource per iteration, a possible adaption would be to let the user with the smallest metric Q chose all resources at once. These methods will be called *Bulk*.

IV. Allocator Performance

The authors of [4] concluded that grouping adjacent subchannels together to a single resource does not affect the allocation performance as long as the resulting subchannel width is smaller than the coherence bandwidth at correlation threshold 0.9. Additionally, it decreases the allocation complexity. Therefore, in the following investigations the number of available resources R_{sys} is reduced to 200 by merging two adjacent subchannels.

TABLE II
Simulation Parameters

System Load	η	100 %
Total Subchannels	R_{sys}	200
Number of Users	U	20
Subchannels per User	R_{user}	10
Bandwidth per User	B_{user}	3.91 MHz

A. Reliability

a) Lookahead Distance: As some of the allocators described in Sec. III-C only take a subset of the CSIs into account, the size L of this subset was investigated. Different simulations showed, that L should be a (fractional rational) multiple of the number of allocated subchannels per user R_{user}, as the optimal window size correlated with the number of subchannels per user. But this relation opens up another question if the window of a user's quality metric should change dynamically with the number of subchannels that a user still needs in order to complete its allocation, i. e.

$$L_u = l \cdot (R_{\text{user}} - M_{\text{alloc},u}), \tag{16}$$

where $M_{\text{alloc},u}$ denotes the number of subchannels already allocated to a user u. The alternative would be that the window size remains static during the allocation process:

$$L = l \cdot R_{\text{user}}. \tag{17}$$

However, for both methods it will occur that L is larger than the number of available resources as the list of available subchannels shrinks. In these cases the window size is set to the number of available resources.

Generally speaking, all allocators have an optimal l which is in the order of one to three and utilizing a smaller window size distance leads to significantly worse outcomes than using a larger distance. As a conclusion, all future uses of the preference based allocators calculating their quality metric utilizing a window will be executed with window size which is a multiple of $l = 1.5$ of the number R_{user} of subchannels that will be allocated.

b) Coherence Bandwidth: The coherence bandwidth is the main characteristic of a scenario. Consequently, this section will compare how sensitive the different allocators are to changes of this property. To be able to execute this comparison, multiple allocations have been performed utilizing generated channel data. The different coherence bandwidths, ranging from 0.196 MHz to 3.609 MHz at a correlation threshold of 0.9, were achieved by the scaling of the measured channel's covariance matrix as done in (9). In Fig. 3, it can be observed that a higher coherence bandwidth leads to lower outage power thresholds needed to reach an outage rate of 10^{-3}, with *Static-Consecutive* being the most sensitive. The conventional allocator *StaticInterleaved* achieves higher outage power thresholds and even outperforms *PrefSingle* as it benefits from spectral diversity. However, *PrefSingle* performs generally worse than the remaining different preference based allocators, namely *PrefGradient*, *PrefWindow* and *PrefWeighted*. These allocators

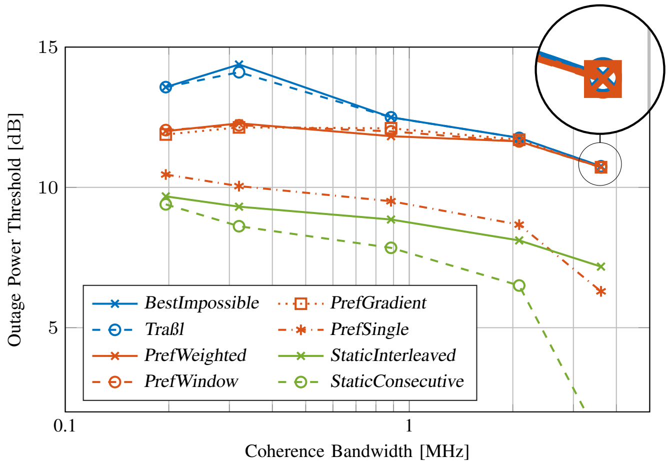

Fig. 3. Outage power threshold of different allocators which causes outages 10^{-3} plotted over the coherence bandwidth of the evaluated synthetic channel data with 20 users (higher is better).

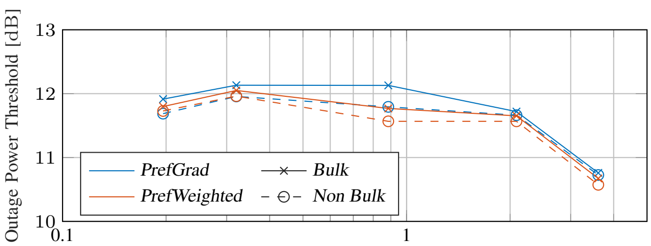

Fig. 4. Outage power threshold of different allocators which causes outages of 10^{-3} plotted over the coherence bandwidth of the evaluated synthetic channel data with 20 users utilizing the Bulk allocation and the conventional method.

almost reach the performance of the theoretical upper bound of *BestImpossible* for higher coherence bandwidths but performs slightly worse at lower ones.

When comparing the reliability of the *Bulk* allocators to their *non-Bulk* counterparts in Fig. 4 it can be seen that the performance only reduces by a fraction of a dB. Looking at *PrefWeighted* it is evident that assigning only one resource per iteration does improve the performance for low outage rates for all coherence bandwidths, while doing the same with the allocator *PrefGradient* improves the performances of all outage rates for low coherence bandwidths. For higher coherence bandwidths there is no disadvantage when assigning all subchannels at once.

c) Number of Users: We also investigated the influence of the number users as more users mean a higher competition between the users for the available resources. Therefore, it may be expected that more users lead to a worse performance of the system w.r.t. the outage rate.

However, all allocators suffer from a higher user count similarly and a negative correlation of the user count and the performance of the allocators is present. Therefore, none of the allocators is specifically well suited for higher a number of users.

B. Computational Effort

While the previous sections covered the reliability of the allocators, URLLC also requires a low latency connection as the name suggests. One aspect of achieving this desired low latency is to reduce the computing time of the allocators. This also ensures the CSI being still up date when the allocation process is completed.

1) Complexity: The optimal allocator by Traßl utilizes the DM decomposition which has to be performed in every iteration of the algorithm by Traßl. Of course, the complexity of the DM decomposition depends on its implementation but according to [16] it can be realized with $\mathcal{O}(E^2)$ where E is the number of edges of the graph. In the case of the *Traßl* algorithm the number of edges is the sum of the subchannels and users in the window ω. In contrast to the preference based methods, the complexity of finding and deleting the resource can be neglected as the DM decomposition has a higher one. In the worst case, the resources of the weakest user always perform worse than every resource of every other user:

$$\sum_{u=1}^{U}\sum_{r=0}^{R_{\text{user}}} \mathcal{O}\left(\left((U-u)(R_{\text{sys}}-u\cdot r)\right)^2\right) = \mathcal{O}\left(R_{\text{user}}^3 U^5\right). \quad (18)$$

When utilizing the preference based allocators by [4], for every user and iteration a search operation of $\mathcal{O}(n)$ and a deletion procedure of $\mathcal{O}(n)$ has to be performed as the chosen subchannel has to be removed from the preference lists of each user. Thus, each iteration of the preference based algorithm has a complexity of $\mathcal{O}(n)+\mathcal{O}(n)=\mathcal{O}(n)$, where n is the length of the preference list.

At the beginning of the process, n consists of $R_{\text{sys}} = R_{\text{user}} \cdot U$ resources which will be reduced by one for each iteration as a subchannel is removed. The worst case for the algorithm would occur if every user gets allocated a resource before another user gets its second. This would mean that the first user can only be removed from the process after $(R_{\text{user}}-1)U+1$ iterations, because every user has to get R_{user} resources assigned until they will be removed. Subsequently, after each iteration with a complexity $\mathcal{O}(n)$ another user is removed from the process, resulting in the overall complexity:

$$\sum_{i=0}^{(R_{\text{user}}-1)U} U\mathcal{O}(UR_{\text{user}}-i) + \sum_{u=0}^{U-1}(U-u)\mathcal{O}(U-u) \quad (19)$$

$$= \mathcal{O}\left(R_{\text{user}}^2 U^3\right). \quad (20)$$

In the improved Bulk version of the preference based algorithm every user gets all of their preferred subchannels when they are chosen due to their Q. This reduces the number of iterations by a factor of R_{user} and leads to:

$$\sum_{u=0}^{U-1}(U-u)\mathcal{O}\left(UR_{\text{user}}-uR_{\text{user}}\right) = \mathcal{O}\left(R_{\text{user}}U^3\right). \quad (21)$$

As expected, the complexity is reduced by a factor of R_{user}.

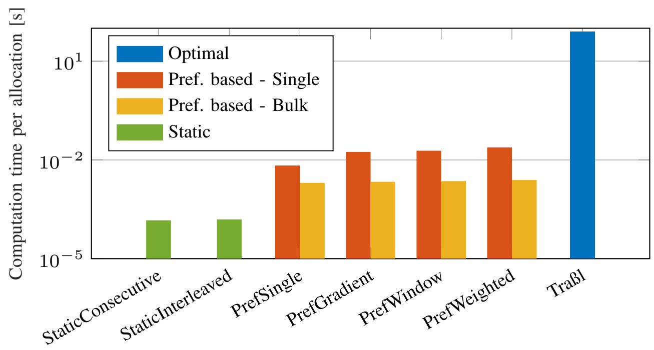

Fig. 5. Compute times per allocation for the allocators done with the simulation parameters of Tab. II.

2) Empirical Results: Besides investigating the complexity of the adaptive allocators, we also compare the computational effort empirically using the high-level programming language MATLAB. Therefore, the results are only suited for comparing the allocators and not describing their real-life performance. In Fig. 5 the computing time of the allocators for an identical scenario of $R_{\text{sys}} = 200$ subchannels, $U = 20$ users and a coherence bandwidth $B_{\text{coh},0.9} = 3.609\,\text{MHz}$ is illustrated. It can be seen that the channel unaware allocators are by far the fastest performing ones, while the preference list based algorithms require up to 10^2 as much computing time without taking the time for acquiring the CSI into account which would introduce additional overhead. Nevertheless, this section will only compare the time needed to perform the actual allocation. Even more time than for the preference based allocators is needed to perform an allocation with the *Traßl* allocator, of which the computing time per allocation is almost another 10^4 times longer. Looking at the preference based allocators it can be seen that the fastest allocator of these four is *PrefSingle*, as this allocator's quality metric Q is by far the simplest to compute. On the other end of the computing time spectrum is *PrefWeighted* which can also be explained by the computing of Q: while *PrefSingle*'s Q was the simplest, the Q of *PrefWeighted* is the most complex of the four preference based allocators.

In section IV-A the performance deficit of allocating all resources at once was investigated with the result, that the performance losses were small to negligible. Now, in Fig. 5 the significant reduction of these allocators is illustrated. The reduction ranges from a factor of almost 3.5 (*PrefSingle*) to a maximum reduction of a factor of 10 (*PrefWeighted*). Consequently, the preference based allocators can be improved by employing the *Bulk* method without significantly compromising on the reliability.

V. Conclusion and Outlook

In this work, we compared the performance of different CSI aware RRA algorithms for different scenarios. For this, we utilized synthetic, spectrally correlated channel fading in order to model different channel conditions by varying the coherence bandwidth of the channel. We showed that the employed allocators are suited for all scenarios regarding the coherence bandwidth and the user count. This work also confirms the conclusions of [2] and [4], in which was stated that the outage rate of a system can be reduced significantly when utilizing a channel aware allocator.

Additionally, we showed that the optimal max-min algorithm proposed in [2] might not be the ideal choice for employment in URLLC due to its high complexity. While the preference based algorithms of [4] were also more complex than conventional MC methods, they were generally faster than *Traßl* from [2]. Additionally, we improved their allocation complexity by introducing the *Bulk* variants. Therefore, we proposed a new set of allocators with lower complexity and negligibly worse performance in terms of reliability.

Future studies could extend the allocators by allowing different numbers of resources for each user to utilize the available spectrum more efficiently and benefit from situations where one user is already satisfied with less subchannels. This would free up spectrum which then can be used by a different user which would otherwise experience an outage.

References

[1] G. P. Fettweis, "The tactile internet: Applications and challenges," *IEEE Vehicular Technology Magazine*, vol. 9, no. 1, pp. 64–70, 2014.
[2] A. Traßl *et al.*, "On the Outage Probability of Channel Prediction Enabled Max-Min Radio Resource Allocation," *IEEE Wireless Communications and Networking Conference (WCNC)*, p. 6, Apr. 2022.
[3] Z. Li *et al.*, "5G URLLC: Design Challenges and System Concepts," in *2018 15th International Symposium on Wireless Communication Systems (ISWCS)*, Aug. 2018, pp. 1–6.
[4] N. Schwarzenberg *et al.*, "Channel-aware multi-user resource allocation for ultra-reliable low-latency communications," in *2023 IEEE 34th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, Toronto, Canada, Sep. 2023.
[5] M.-T. Suer *et al.*, "Multi-Connectivity as an Enabler for Reliable Low Latency Communications—An Overview," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 1, pp. 156–169, 2020.
[6] A. Wolf *et al.*, "How Reliable and Capable is Multi-Connectivity?" *IEEE Transactions on Communications*, vol. 67, no. 2, Feb. 2019.
[7] G. Mainland, D. C. Parkes, and M. Welsh, "Decentralized, Adaptive Resource Allocation for Sensor Networks," in *Proceedings of the 2nd Symposium on Networked Systems Design and Implementation*, 2005.
[8] G. Araniti *et al.*, "Adaptive Resource Allocation to Multicast Services in LTE Systems," *IEEE Transactions on Broadcasting*, vol. 59, no. 4, pp. 658–664, Dec. 2013.
[9] C. Li and X. Wang, "Adaptive subchannel allocation in multiuser MC-CDMA systems," in *IEEE Global Telecommunications Conference, 2004. GLOBECOM '04.*, vol. 4, Nov. 2004, pp. 2503–2507 Vol.4.
[10] L. Massoulie and J. Roberts, "Bandwidth sharing: objectives and algorithms," *IEEE/ACM Transactions on Networking*, vol. 10, no. 3, 2002.
[11] A. Traßl *et al.*, "Outage prediction for ultra-reliable low-latency communications in fast fading channels," *EURASIP Journal on Wireless Communications and Networking*, vol. 2021, no. 1, p. 92, Dec. 2021.
[12] F. Burmeister *et al.*, "Measuring Time-Varying Industrial Radio Channels for D2D Communications on AGVs," in *2021 IEEE Wireless Communications and Networking Conference (WCNC)*, Mar. 2021.
[13] L. C. Tran *et al.*, "A Generalized Algorithm for the Generation of Correlated Rayleigh Fading Envelopes in Wireless Channels," *EURASIP Journal on Wireless Communications and Networking*, vol. 2005, no. 5, Dec. 2005.
[14] W. C. Jakes, *Microwave Mobile Communications*, 2015.
[15] B. Natarajan, C. Nassar, and V. Chandrasekhar, "Generation of correlated Rayleigh fading envelopes for spread spectrum applications," *IEEE Communications Letters*, vol. 4, no. 1, pp. 9–11, Jan. 2000.
[16] J. Chen and I. A. Kanj, "Constrained minimum vertex cover in bipartite graphs: complexity and parameterized algorithms," vol. 67, no. 4, 2003, pp. 833–847, parameterized Computation and Complexity 2003.