

Article

GPU Accelerated PIC and SIC for OFDM-NOMA

Talgat Manglayev ^{1,*} , Refik Caglar Kizilirmak ² and Nor Asilah Wati Abdul Hamid ³¹ Department of Electrical and Computer Engineering, Nazarbayev University, Astana Z05H0P9, Kazakhstan² Department of Electrical and Electronics Engineering, Nazarbayev University, Astana Z05H0P9, Kazakhstan; refik.kizilirmak@nu.edu.kz³ Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, Selangor 43400, Malaysia; asila@upm.edu.my

* Correspondence: talgat.manglayev@nu.edu.kz

Received: 29 January 2019; Accepted: 22 February 2019; Published: 26 February 2019



Abstract: Non-orthogonal multiple access (NOMA) is a candidate multiple access scheme for the fifth-generation (5G) cellular networks. In NOMA systems, all users operate at the same frequency and time, which poses a challenge in the decoding process at the receiver side. In this work, the two most popular receiver structures, successive interference cancellation (SIC) and parallel interference cancellation (PIC) receivers, for NOMA reverse channel are implemented on a graphics processing unit (GPU) and compared. Orthogonal frequency division multiplexing (OFDM) is considered. The high computational complexity of interference cancellation receivers undermines the potential deployment of NOMA systems. GPU acceleration, however, challenges this weakness, and our numerical results show speedups of about from 75–220-times as compared to a multi-thread implementation on a central processing unit (CPU). SIC and PIC multi-thread execution time on different platforms reveals the potential of GPU in wireless communications. Furthermore, the successful decoding rates of the SIC and PIC are evaluated and compared in terms of bit error rate.

Keywords: non-orthogonal multiple access; successive interference cancellation; parallel interference cancellation; orthogonal frequency division multiplexing; graphics processing unit; CUDA

PACS: J0101

1. Introduction

Non-orthogonal multiple access (NOMA) in the power domain has been proposed as a promising multiple access technique for the upcoming fifth-generation cellular networks. In theory, NOMA fulfills the spectral efficiency requirements of 5G by serving multiple users simultaneously in the same frequency band [1]. For example, in the reverse link of a NOMA system, each user transmits at the same time and in the frequency band so that the base station receives the superimposed version of the transmitted signals by each user. This is in contrast to the multiple access methods used in conventional cellular systems where one user is allocated per time slot or frequency unit [2]. Early works on NOMA demonstrated that NOMA can significantly improve the sum capacity and cell-edge user throughput [3,4].

In NOMA systems, since each user is seen as an interference to others, advanced interference cancellation is required for successful decoding [5]. In the literature, successive interference cancellation (SIC) and parallel interference cancellation (PIC) techniques are considered for NOMA. However, both techniques require massive computation power. For example, in [6], the authors reported that their NOMA testbed can support only nine users due to the limitations in the computation power of their personal computers.

5G networks are expected to boost the data rates while significantly reducing the latency, which requires powerful and efficient baseband processing at the receiver. In this work, the two most popular interference cancellation schemes, PIC and SIC, are implemented and compared in a graphics processing unit (GPU) using CUDA to speed up the most time-consuming process of a NOMA receiver. Numerical results show that PIC, due to its parallel architecture, is more suitable for GPU implementation and outperforms the SIC with a speedup of over 70-times. Moreover, an interference cancellation processing times as low as 7 ms, even for large number of users, is observed with PIC in GPU. The results also reveal the feasibility of interference cancellation techniques for NOMA and their competitiveness in commercial solutions particularly when accelerated by GPU using CUDA.

In addition to their computation performances, the reliability of the transmission with SIC and PIC receivers may differ [7]. In SIC receivers, multi-user interference is removed in a successive manner so that the reliability of the system heavily depends on the correct decoding of the first user. Moreover, in SIC receivers, the received powers from each user should be different so that the receiver can distinguish the users in the power domain and perform successful decoding. In PIC receivers, on the other hand, all the users are decoded at once in parallel, and its performance does not depend on the difference between the received powers as in SIC receivers. However, in order to support a large number of users, PIC-based NOMA systems may require signature signals for each user as in code division multiple access (CDMA) and are more suitable for code domain NOMA [8,9]. In this work, we further present the relevant signal models for both SIC and PIC receivers and discuss their bit-error-rate (BER) performances.

This work is organized as follows. In the next section, we present the system models where the SIC and PIC receivers for OFDM are described. In the Section 3, the CUDA implementation of the receivers is described. Numerical results are presented and discussed in Section 4. We summarize the study in Section 5.

2. System Model

The considered that the NOMA system model in the reverse link consists of a single cell with K users. Each user uses all the N subcarriers of the orthogonal frequency division multiplexing (OFDM) symbol, and they are distinguished at the base station by interference cancellation techniques. Let α_k be the channel attenuation between the k^{th} user and base station. We assume that the K users are distributed in the coverage area of the base station such that $|\alpha_1|^2 > |\alpha_2|^2 > \dots > |\alpha_K|^2$ so that the first user is the closest to the base station and the K^{th} user is the farthest from the base station. The signal received by the base station is the superimposed signal of all the transmitted signals and can be written as:

$$y(t) = \sum_{k=1}^K \sqrt{P} s_k(t) \alpha_k + n(t) \quad (1)$$

where P is the transmitted power, which is equal for each user, $n(t)$ is the additive white Gaussian noise term with variance σ_n^2 , and $s_k(t)$ is the transmitted OFDM symbol by the k^{th} user. The discrete-time domain OFDM symbol transmitted by each user can be written as:

$$s_k[m] = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} S_k[n] e^{j2\pi mk/N} \quad (2)$$

where $S_k[n]$ is a complex symbol at the subcarrier n (either phase shift keying (PSK) or quadrature amplitude modulation (QAM) symbols) and N is the OFDM size. Then, the continuous-time OFDM waveforms transmitted by the k^{th} user $s_k(t)$ can be obtained as $s_k(t) = \sum_{m=0}^{N-1} s_k[m] p(t - mT_s)$ where T_s is the baud rate and $p(t)$ is the pulse-shaping filter.

The received signal by the base station in Equation (2) includes all the signals from each user, and in order to decode each user successfully, the base station implements interference cancellation. The block diagrams of the two considered SIC and PIC receivers for OFDM are given in Figures 1 and 2,

respectively. Both techniques include OFDM demodulation and modulation operations. To be specific, the fast Fourier transform (FFT) operation obtains the complex symbols at each subcarrier, the modulation/demodulation blocks deal with mapping/demapping of bit sequences to complex symbols, and the inverse fast Fourier transform (IFFT) obtains the time domain OFDM signal from frequency domain complex symbols. MATLAB functions used for FFT and IFFT computations use multiple CPU cores, while the rest of the computations including loops are not parallelized.

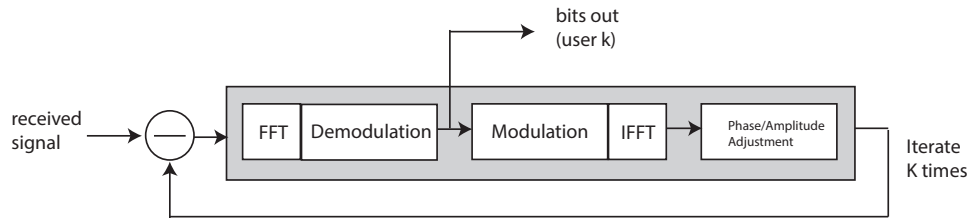


Figure 1. Block diagram of a typical successive interference cancellation (SIC).

In the SIC receiver (see Figure 1), the information signal of each user is decoded sequentially in an iterative manner [10]. The received signal by the base station is basically the sum of the OFDM signals received from each user. The first signal the receiver decodes (i.e., in the first iteration) belongs to the user that is closest to the base station, and its signal has the highest weight in the received signal, while other users' signals are considered as interference. After the bit sequences are obtained (ideally without any error), the OFDM signal of that particular user is regenerated by following the exact procedure at an OFDM transmitter and adjusting its amplitude and phase as it passes through the channel from that particular user to the base station. The regenerated signal is then subtracted from the received signal, and the process is repeated until all the users are decoded. At iteration k , the time domain OFDM signal for the k^{th} user $\hat{s}_k(t)$ becomes, assuming perfect cancellation at each previous iteration and perfect phase/amplitude adjustment during regeneration [11],

$$\hat{s}_k(t) = \sqrt{P}\alpha_i s_k(t) + \sum_{i=k+1}^K \sqrt{P}\alpha_i s_i(t) + n(t). \quad (3)$$

The signal-to-interference and noise ratio (SINR) of the k^{th} user per subcarrier can be written as:

$$\text{SINR}_k = \frac{P|\alpha_k|^2}{\sigma_n^2 + \sum_{i=k+1}^K P|\alpha_i|^2}. \quad (4)$$

Then, the bit error rate (BER) for the k^{th} user per subcarrier with the SIC can be written for binary transmission as $\text{BER}_k = Q(\sqrt{2\text{SINR}_k})$ where $Q(\cdot)$ is the standard Q-function [12]. In this work, the channel is assumed to be frequency flat, so that the SNR and BER per subcarrier values become the overall values for that particular user k .

In the PIC receiver (see Figure 2), interference cancellation occurs in two stages. In the first stage, the information signal of each user is decoded collectively [13]. For example, in order to decode the k^{th} user's signal, the other users' signals are decoded and regenerated in parallel, then their sum is subtracted from the received signal. The stripped signal includes the signal for the k^{th} user only so that, in the second stage, it can be decoded using OFDM demodulation. Once the set of regenerated OFDM signals is obtained in the first stage, the signals of each user can easily be obtained. For PIC, the time domain OFDM signal for the k^{th} user $\hat{s}_k(t)$ can be written as:

$$\hat{s}_k(t) = \sum_{i=1}^K \sqrt{P}\alpha_i s_i(t) - \sum_{i=1, i \neq k}^K \sqrt{P}\alpha_i \hat{s}_i(t) + n(t). \quad (5)$$

Derivation of closed-form SINR for PIC is mathematically intractable since it depends on the decoding errors in the first stage. The BER for the k^{th} user per subcarrier for a binary transmission can still be expressed as [7,14,15]:

$$BER_k = Q \left\{ \left[\frac{1}{2(P_k/\sigma_n^2)} \left(\frac{1 - (\frac{K-1}{3})^{s+1}}{1 - (\frac{K-1}{3})} \right) + \frac{1}{3^{s+1}} \right. \right. \\ \left. \left. \times \left(\frac{(K-1)^{s-1} - (-1)^{s+1}}{K} \left(\frac{\sum_{i=1, i \neq k}^K P_i}{P_k} + 1 \right) + (-1)^{s+1} \right) \right]^{-1/2} \right\} \quad (6)$$

where s is the number of stages in the PIC receiver (which is two in our case). Note that Equation (6) is obtained based on a similar approach as in multi-stage receivers in code division multiple access (CDMA) with unit processing gain [7]. In CDMA systems, however, each user has signature codes that help to distinguish them. In power domain NOMA systems, on the other hand, these unique codes may not be present, and they can fail to support large number of users since the interference grows stronger as the number of users increases. Code domain NOMA can be considered in order to have comparable reliability for both SIC and PIC; however, comparison of computation performances will be analogous to the ones in the power domain [16,17].

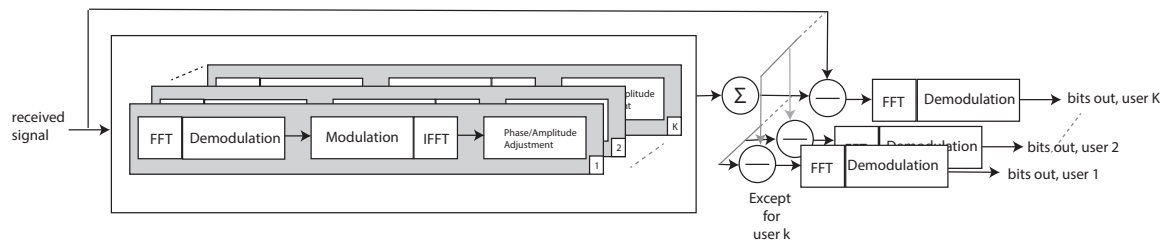


Figure 2. Block diagram of a typical parallel interference cancellation (PIC).

In both techniques, any decoding error that occurs in the intermediate stages propagates to the other stages of the receiver. Furthermore, both receivers rely on accurate power allocation among the users to ensure successful interference cancellation. These will have an impact on the reliability of the receivers (e.g., bit error rate performances); however, they will not change the computational complexity.

In this work, without loss of generality, equal power allocation among the users and power domain NOMA is considered.

3. CUDA Implementation

The SIC and PIC techniques were implemented on both a central processing unit (CPU) and GPU, then their computational speeds were compared. The CUDA codes were compiled on a machine that ran Ubuntu OS with a 12-GB memory NVIDIA TITAN Xp graphics card that had 3840 CUDA cores and a clock rate of 1582 MHz. For computing, the NVCC compiler for the CUDA 9.2 platform and the GCC compiler for C++ object-oriented programming language were used. The CPU codes were compiled with an Intel Core i7 (four cores) 2.3-GHz, DDR3L 1600-MHz 16-GB memory machine that ran Mac OS Mojave. The results with CPU were obtained on MATLAB R2018b.

Figure 3 summarizes the functions and kernels used for the implementation of SIC and PIC on GPU using CUDA. As for the FFT and IFFT tasks, the cuFFTlibrary functions with forward and inverse parameters were called. The functions had $\mathcal{O}(n \log n)$ computation time complexity. One CUDA thread was assigned per subcarrier. A CUDA block of our GPU allowed calling up to 1024 threads per block. For SIC, the chain of operations for each user included FFT, demodulation, modulation, and IFFT with phase/amplitude adjustment (see Figure 1). For the FFT and IFFT computations, functions from the

cuFFT library with $\mathcal{O}(n \log n)$ time complexity were used [18]. The CUDA kernels were developed for demodulation, modulation, and subtraction tasks. Due to the nature of the SIC receiver, the same chain of operations had to be repeated for each user and could not run in parallel. The entire SIC computational time on GPU, however, can still be decreased by processing demodulation, modulation, and subtraction operations per subcarrier, one after the other, but computing OFDM in parallel for each user using the divide and conquer algorithm [19]. Therefore, a grid of N_1 blocks each having N_2 threads was created so that the total number of parallel tasks for each kernel was equal to the number of subcarriers, i.e., $N_1 \times N_2 = N$. These parallel tasks computed OFDM for demodulation, modulation, and subtraction, and this process constitutes one SIC iteration. Then, the process was repeated K times in order to decode the signal of each user.

In the implementation of PIC on GPU, all the functions and kernels per subcarrier of each user can be executed in parallel. In this case, the number of threads per block was set equal to the number of users K . Each thread handles the tasks per subcarrier for each user. The grid of blocks was then created with the number of blocks equal to the FFT size N , which made the total number of parallel tasks $K \times N$.

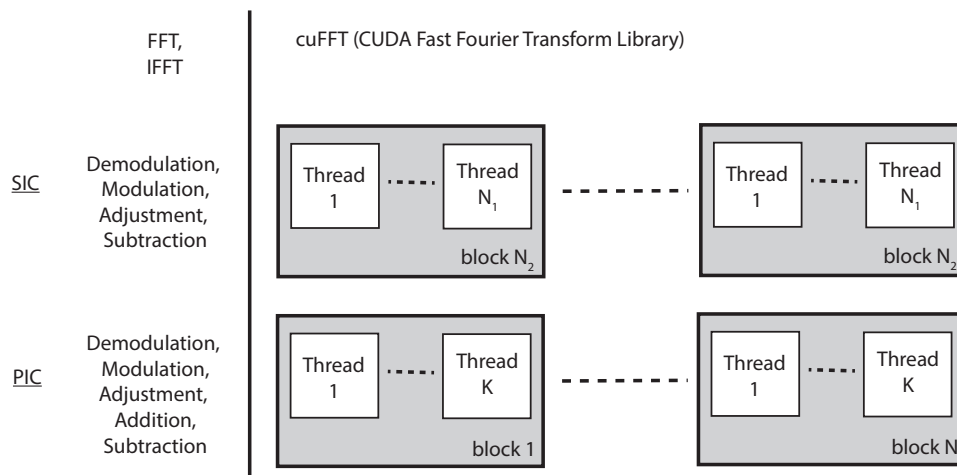


Figure 3. Functions and kernels for SIC and PIC using CUDA.

4. Numerical Results and Discussion

In this section, first we discuss the BER performances of PIC and SIC and then present their implementation on GPU. Single-cell NOMA with different numbers of active users was considered. The K users were distributed in the coverage area of the base station such that $|\alpha_1|^2 > |\alpha_2|^2 > \dots > |\alpha_K|^2$. The user locations were assumed to be fixed. The received power by the base station from the closest user ($P|\alpha_1|^2$) was set at -90 dBm, and the received power differences between the users was 2 dB, i.e., $10\log_{10}(\frac{P|\alpha_i|^2}{P|\alpha_{i+1}|^2}) = 2$ dB. Figure 4 shows the BER of the first user versus the total number of users (K) for both PIC and SIC receivers. The results were obtained for different SNR levels of the first user, $\text{SNR} = P|\alpha_1|^2 / \sigma_n^2$. When SNR was taken as 15 dB, σ_n^2 was set at -105 dBm, and similarly, when SNR was 20 dB, σ_n^2 was -110 dBm. The results in Figure 4 show that for a large number of users in a cell, an increase in the SNR had an insignificant impact on the BER performance of the SIC receiver. This is because the performance was mainly limited by the interference by the other users (see Equation (4)). For the PIC receiver, on the other hand, an increase in the SNR had a significant impact on the BER performance even for a high number of users. Here, it should be noted that the BER performance for the PIC receiver was obtained using Equation (6), which assumed that signature signals were present for each user, which helped in the detection.

Next, we discuss the implementation of the two interference cancellation methods on the GPU platform. Tables 1 and 2 summarize the computation times obtained both on GPU and CPU for the

two interference cancellation techniques. The results with GPU include only the execution times of SIC and PIC; in other words, the time for communication between GPU and CPU spent to copy global variables back and forth to CPU was neglected. MATLAB functions used for FFT and IFFT computations used multiple CPU cores, while the rest of the computations including loops were not parallelized. The size of OFDM was taken as 2048 in Table 1 and 4096 in Table 2, and quadrature phase shift keying (QPSK) with maximum likelihood (ML) decoding was considered [20–22]. A typical NOMA cell was expected to support about 50 users [16]; however, this can be increased by clustering, grouping, or using multiple input multiple output (MIMO) techniques [23,24]. In order to observe the trend with a large number of users, the number of users was varied from 50–350.

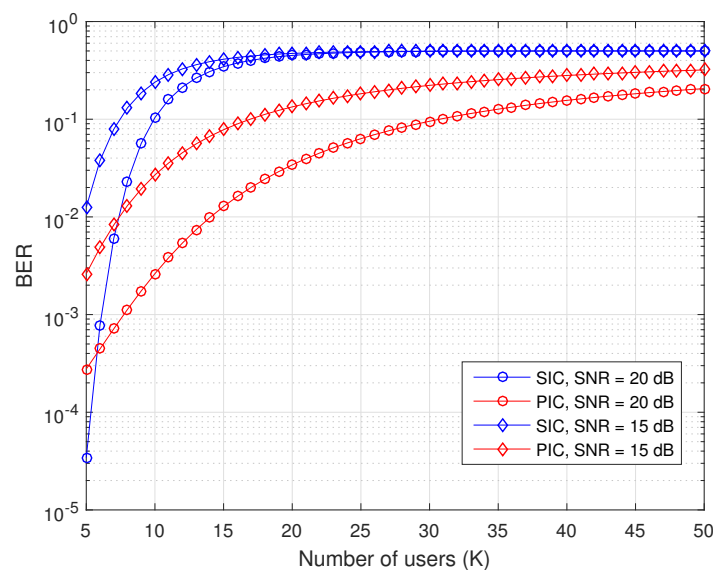


Figure 4. Bit error rate (BER) versus the number of users (K) for PIC and SIC.

The time spent on CPU sharply varied for the SIC and PIC schemes in both OFDM FFT sizes. PIC performed summation (see Figure 2) and also iterated FFT computation and demodulation tasks apart from the same iterated tasks as SIC. Consequently, it took about twice more time to execute PIC than SIC in both tables with CPU. It was also observed that FFT size had a marginal effect on SIC execution time on the GPU platform. It took about from 69 ms for 50 UEs and reaching 515 ms for 350 users. On the one hand, the SIC scheme on GPU was executed slower than on CPU, but on the other hand, it was faster than PIC on CPU. SIC ran slower on GPU, because the process was iterative and depended on the frequency (clock rate) of each called core, rather than on the number of cores. As was mentioned earlier, our CPU had four cores with 2.3 GHz each, and the frequency of our GPU cores was 1.5 GHz (1582 MHz). Moreover, PIC on a CPU with an FFT size of 4096 was the slowest time of all the results. This was due to serial approach of CPU running the scheme and FFT computation. Fifty users may be decoded in about 88 ms. The time spent to decode 350 users impractically reached about 616 ms.

As seen in Tables 1 and 2, the SIC execution time on CPU started from only about 28 ms and 47 ms for 50 users. The time gradually increased to 157 ms and 330 ms for 350 users in a cell respectively for FFT sizes of 2048 and 4096; whereas, PIC on GPU was the fastest. The execution time of PIC on GPU was almost 90- and 138-times faster than on the CPU for different OFDM FFT sizes and less sensitive to the number of users. It took only 2.54 ms for 50 users and 6.88 ms for 350 users with an FFT size of 4096 and approximately 2 ms for any number of users for an FFT size of 2048. Furthermore, it was observed that PIC was nearly 75-times faster than SIC on GPU for a large number of users with a 4096

FFT size and 220-times faster for a 2048 FFT size. This was because SIC had mutual data dependency within users during execution iterations and had to be executed serially on both CPU and GPU.

Table 1. Comparison of the computational times of SIC and PIC in ms with an FFT size of 2048.

Number of Users	CPU		GPU	
	SIC	PIC	SIC	PIC
50	27.70	44.01	68.69	1.97
100	45.49	84.90	138.39	2.08
150	68.22	130.17	208.28	2.19
200	91.05	173.69	281.20	2.21
250	113.73	223.56	353.08	2.23
300	135.94	269.35	426.68	2.26
350	157.98	315.15	500.45	2.28

Table 2. Comparison of the computational times of SIC and PIC in ms with an FFT size of 4096.

Number of Users	CPU		GPU	
	SIC	PIC	SIC	PIC
50	46.65	87.83	69.22	2.54
100	93.77	172.86	141.49	3.22
150	142.67	263.28	213.03	4.13
200	188.77	349.27	285.72	4.81
250	236.25	442.43	360.51	5.52
300	279.16	530.04	439.31	6.00
350	330.65	616.26	514.95	6.88

5. Conclusions

In this work, a CUDA platform was proposed to implement computationally-challenging interference cancellation schemes in the reverse channel of OFDM-NOMA networks. Both SIC and PIC schemes were implemented, and their algorithms along with OFDM were illustrated. Finally, their computation times were compared. Numerical results showed significant speedups of the PIC scheme with GPU implementation as compared to CPU. Furthermore, for a large number of users, PIC was found to be approximately 75-times and 220-times faster than SIC on GPU for FFT sizes of 4096 and 2048, respectively.

5G base stations are expected to face computationally-heavy tasks for baseband processing. In addition to interference cancellation, some other techniques such as MIMO at improved data rates are going to put a tremendous baseband processing load on 5G base stations. The results here demonstrated that PIC on GPU took only 3 ms to decode 50 users and left room for other computationally-heavy processes to satisfy the strict latency requirement of 5G networks.

Author Contributions: T.M. is the first author who proposed the main idea, analyzed and simulated the system, and presented the writing—original draft preparation, writing—review and editing, visualization. R.C.K. is the second who has experience in wireless communication research. He has made a supervision, review, and given the first author some useful comments and funding acquisition for this research. N.A.W.A.H. is the third who analyzed the data and contributed the analysis tools. All authors read and approved the final manuscript.

Funding: This research was partially supported by the Universiti Putra Malaysia under the Grant Putra, GP/2017/9569600.

Acknowledgments: We would like to thank School of Engineering, Nazarbayev University for providing us the GPU equipment for computation and also we would like to thank our colleagues from Faculty of Computer Science and Information Technology, Universiti Putra Malaysia as the NVIDIA GPU Education Center, who provided insight and expertise that greatly assisted the research.

Conflicts of Interest: The authors declare no conflict of interest. The founding sponsors had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, and in the decision to publish the results.

References

1. Dai, L.; Wang, B.; Yuan, Y.; Han, S.; Chih-lin, I.; Wang, Z. Non-orthogonal multiple access for 5G: Solutions, challenges, opportunities, and future research trends. *IEEE Commun. Mag.* **2015**, *53*, 74–81. [[CrossRef](#)]
2. Tse, D.; Viswanath, P. *Fundamentals of Wireless Communication*; Cambridge University Press: New York, NY, USA, 2005; pp. 120–156, ISBN 978-0521845274.
3. Manglayev, T.; Kizilirmak, R.C.; Kho, Y.H. Optimum power allocation for non-orthogonal multiple access (NOMA). In Proceedings of the 2016 IEEE 10th International Conference on Application of Information and Communication Technologies (AICT), Baku, Azerbaijan, 12–14 October 2016; pp. 1–4.
4. Saito, Y.; Kishiyama, Y.; Benjebbour, A.; Nakamura, T.; Li, A.; Higuchi, K. Non-Orthogonal Multiple Access (NOMA) for Cellular Future Radio Access. In Proceedings of the 77th Vehicular Technology Conference (VTC Spring), Dresden, Germany, 2–5 June 2013; pp. 1–5.
5. Yunzheng, T.; Long, L.; Shang, L.; Zhi, Z.H. A Survey: Several Technologies of Non-Orthogonal Transmission for 5G. *China Commun.* **2015**, *12*, 1–15. [[CrossRef](#)]
6. Tran, T.; Voznak, M. Multi-Points Cooperative Relay in NOMA System with N-1 DF Relaying Nodes in HD/FD Mode for N User Equipments with Energy Harvesting. *Electronics* **2019**, *8*, 167. [[CrossRef](#)]
7. Anwar, A.; Seet, B.-C.; Li, X.J. *Recent Advances in Information, Communications and Signal Processing*; Non-orthogonal Multiple Access: Recent Developments and Future Trends; River Publishers: Roma, Italy, 2018; pp. 1–34, ISBN 978-87-93609-42-6.
8. Riazul Islam, S.M.; Avazov, N.; Dobre, O.A.; Kwak, K. Power-domain non-orthogonal multiple access (NOMA) in 5G systems: Potentials and challenges. *IEEE Commun. Surv. Tutor.* **2017**, *19*, 721–742. [[CrossRef](#)]
9. Ori, S.; Zaidel, B.M.; Shitz, S. Low-density code-domain NOMA: Better be regular. In Proceedings of the 2017 IEEE International Symposium on Information Theory (ISIT), Aachen, Germany, 15 August 2017; pp. 2628–2632.
10. Song, G.; Wang, X. Comparison of Interference Cancellation Schemes for Non-Orthogonal Multiple Access System. In Proceedings of the 83rd Vehicular Technology Conference (VTC Spring), Nanjing, China, 15–18 May 2016; pp. 1–5.
11. Kizilirmak, R.C. *Towards 5G Wireless Networks-A Physical Layer Perspective*; Non-Orthogonal Multiple Access (NOMA) for 5G Networks; IntechOpen: Rijeka, Croatia, 2016; ISBN 978-953-51-2833-5.
12. Goldsmith, A. *Wireless Communications*; Cambridge University Press: New York, NY, USA, 2005; ISBN 978-0521837163.
13. Anwar, A.; Seet, B.; Li, X.J. PIC-based receiver structure for 5G downlink NOMA. In Proceedings of the 10th International Conference on Information, Communications and Signal Processing (ICICS), Singapore, 2–4 December 2015; pp. 1–5.
14. Kaul, A.; Woerner, B.D. Analytic limits on performance of adaptive multistage interference cancellation for CDMA. *Electron. Lett.* **1994**, *30*, 2093–2095. [[CrossRef](#)]
15. Buehrer, R.M.; Correal-Mendoza, N.S.; Woerner, B.D. A simulation comparison of multiuser receivers for cellular CDMA. *IEEE Trans. Veh. Technol.* **2000**, *49*, 1065–1085. [[CrossRef](#)]
16. Shin, W.; Vaezi, M.; Lee, B.; Love, D.J.; Lee, J.; Poor, H.V. Non-orthogonal multiple access in multi-cell networks: Theory, performance, and practical challenges. *IEEE Commun. Mag.* **2017**, *55*, 176–183. [[CrossRef](#)]
17. Sundararajan, M.; Govindaswamy, U. Multi-carrier Spread Spectrum Modulation Schemes and Efficient FFT Algorithms for Cognitive Radio Systems. *Electronics* **2014**, *3*, 419–443. [[CrossRef](#)]
18. NVIDIA Corporation. CUDA Toolkit 4.2 CUFFT Library. Programming Guide. Available online: https://developer.download.nvidia.com/compute/DevZone/docs/html/CUDALibraries/doc/CUFFT_Library.pdf (accessed on 14 February 2019).
19. Acar, U.A.; Blelloch, G.E. Algorithm Design: Parallel and Sequential. 2017, pp. 46–48. Available online: <http://www.parallel-algorithms-book.com> (accessed on 13 September 2018).

20. Rappaport, T.S. 5G Millimeter Wave Wireless: Trials, Testimonies, and Target Rollouts. In Proceedings of the IEEE Workshop on Millimeter-Wave Network Systems (mmSys) IEEE Infocom, Honolulu, HI, USA, 16 April 2018.
21. Myint, S.H.; Yu, K.; Sato, T. Modelling and Analysis of Error Process in 5G Wireless Communication Using Two-state Markov Chain. *IEEE Access* **2019**. [[CrossRef](#)]
22. Roh, W.; Seol, J.; Park, J.; Lee, B.; Lee, J.; Kim, Y.; Cho, J.; Cheun, K.; Aryanfar, F. Millimeter-Wave Beamforming as an Enabling Technology for 5G Cellular Communications: Theoretical Feasibility and Prototype Results. *IEEE Commun. Mag.* **2014**, *52*, 106–113. [[CrossRef](#)]
23. Manglayev, T.; Kizilirmak, R.C.; Kho, Y.H.; Hamid, N.A.W.A. GPU Accelerated Successive Interference Cancellation for NOMA Uplink with User Clustering. *Wirel. Pers. Commun.* **2018**, *103*, 2391–2400. [[CrossRef](#)]
24. Li, Y.; Baduge, G.A.A. Underlay Spectrum-Sharing Massive MIMO NOMA. *IEEE Commun. Lett.* **2019**, *23*, 1–4. [[CrossRef](#)]



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).