Self-Tuning RIS Controller for Vehicular Communications: A Hardware Perspective

Shahriar Shahabuddin*, Md. Salman Sakib*, Hassan Malik†, Mahmoud A. Albreem‡
*Oklahoma State University, Stillwater, OK, USA
†University of East Anglia, Norwich, England
‡University of Sharjah, Sharjah, UAE
Email: *[firstname.lastname]@okstate.edu, †hassan.malik@uea.ac.uk, ‡malbreem@sharjah.ac.ae

***Abstract*—Reconfigurable intelligent surfaces (RIS) can reflect an incoming signal toward a target location to enhance the received signal quality. In most studies, it is assumed that the base station (BS) provides configuration data to an RIS controller. However, for a passive RIS with a high number of elements, this data transmission from the BS can introduce significant delays. This challenge is especially pronounced in vehicular communications, where reflection coefficients must be updated even for a few millimeters of movement. To overcome this limitation, we propose a self-tuning RIS controller that independently calculates the reflection coefficients with minimal input from the BS. Our RIS controller leverages the user's location information to compute the reflection coefficients in three steps: channel gain calculation, steering vector calculation, and optimal phase control. A pipelined architecture is developed to support these steps in real-time. The proposed VLSI architecture is implemented on an Artix-7 XC7A100TCSG324-1 FPGA to calculate reflection coefficients for a vehicle moving at a 100 kmh speed.**

***Index Terms*—RIS, RIS Controller, FPGA, VLSI, steering vector, optimal phase**

I. Introduction

Reflective intelligent surfaces (RIS) represent a breakthrough technology widely regarded as a key contender for the sixth-generation (6G) wireless standard [1], [2]. The concept of RIS involves adding a metasurface to create an alternative communication path, similar to that provided by relays. In the uplink, signals from users can be reflected by the RIS toward the base station (BS), while downlink communication from the BS to users occurs in the reverse direction. As 6G advances toward high-frequency communications, such as in the sub-THz or visible light spectrum, these frequencies face increased absorption and penetration loss [3]. When a line-of-sight (LOS) path between the user and BS is obstructed, the alternative route provided by RIS can significantly boost the signal-to-noise ratio (SNR) at the receiver [4]. Thus, RIS is crucial for ensuring effective high-frequency communication in 6G [5], [6]. RIS also holds tremendous potential for space communications, vehicular communications, and wireless sensing applications.

RIS employs varying voltage levels to adjust the reflection properties and phase angle of the reflected beam. In existing literature, it is assumed that a software-programmable device, such as a microcontroller, will function as the back-end to generate these voltages for configuring the RIS. This hardware device is typically referred to as *RIS controller*. Furthermore, most literature assumes that the RIS controller receives information from the base station (BS) to optimize the phase angles of each element, focusing the reflected beam toward the BS. It is important to note that without proper reflection coefficients, the RIS does not improve received power or capacity. However, obtaining these coefficients from the BS may be subject to delays, which could worsen in the case of a moving user [7]. Therefore, we believe that the right place for calculating the reflection coefficients is within the RIS controller itself. We refer to this as a *self-tuning RIS controller*. In this paper, we introduce a digital very-large-scale integration (VLSI) architecture for a self-tuning RIS controller tailored for vehicular communications. Our proposed controller can compute optimized reflection coefficients with minimal feedback data from the BS. There are three primary reasons for opting for a VLSI design over a software implementation in a microcontroller: (1) The calculations required to determine the reflection coefficients for an RIS are intensive, particularly given the large number of elements (potentially in the hundreds) needed to achieve any performance gains in a passive RIS. (2) In a RIS-assisted vehicular communications, the coefficients must be updated frequently, as even slight user movements—on the scale of millimeters—necessitate recalculations, making the computational load considerably higher. (3) The interfaces between the BS and RIS may restrict large parallel transfers, resulting in reflection coefficient information being calculated and transferred serially [8]. Due to these aforementioned reasons, a software implementation is not suitable to process such a massive amount of data in a very short time. Our proposed architecture rapidly calculates optimized reflection coefficients for an RIS with numerous elements, requiring only the location information from the BS. The proposed VLSI architecture is synthesized for the Arty-A7 development platform, which centers around an Artix-7 field-programmable gate array (FPGA). This device was selected for its low cost and size, which is comparable to real-time microcontrollers like the STM32 or Texas Instruments (TI) TIVA series.

The rest of the paper is organized in the following way: Section II presents the system model of RIS-aided vehicular communication, along with a discussion of the optimized reflection coefficient calculation method. In Section III, we present our simulation results including fixed-point simulations

to find out the appropriate word length. In Sections IV and V, we present the VLSI architecture of the self-tuning RIS and FPGA implementation results. The paper is concluded in section VI.

II. System Model

We consider a point-to-point vehicular communication system supported by an N-element RIS. The RIS consists of passive elements to facilitate communication between a single-antenna moving user device and a single-antenna BS. The position of the user, RIS and the BS can be denoted in a three-dimensional coordinate by (a_u, b_u, c_u), (a_r, b_r, c_r) and (a_b, b_b, c_b), respectively [9]. The RIS and BS are generally stationary, meaning that the only value that continuously changes is the coordinates of the user with mobility. This scenario is depicted in Fig. 1. We assume the user is transmitting a complex baseband signal x to the receiver via a direct LOS path and also via the RIS. We denote a particular element of RIS by n where $n \in \{1, \ldots, N\}$. We also assume the transmitted symbols from the user to RIS suffer from amplitude attenuation $\alpha_{1,n}$ and phase shift of the frequency-flat channel $\xi_{1,n}$. Therefore, the impinged signal on the n-th RIS element can be expressed as

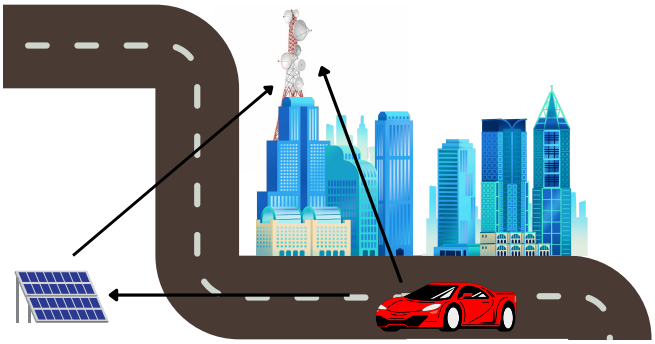

Fig. 1: Vehicular communication using a RIS.

$$z_{\text{in},n} = \alpha_{1,n} e^{-j\xi_{1,n}} x. \tag{1}$$

The RIS element n introduces an amplitude attenuation of β_n and a phase shift of θ_n. The reflected signal of the RIS could be expressed as

$$z_{\text{out},n} = \beta_n e^{j\theta_n} z_{\text{in},n} = \beta_n e^{j\theta_n} \alpha_{1,n} e^{-j\xi_{1,n}} x, \tag{2}$$

where $\theta_n \in [0, 2\pi]$ and $\beta_n \in [0, 1]$ because we are using passive RIS, i.e. the RIS is not equipped with a radio-frequency (RF) chain.

The reflected signal experiences similar channel impairments before reaching the base station. If the amplitude attenuation and the phase shift between the RIS and BS are $\alpha_{2,n}$ and $\xi_{2,n}$, respectively, then the received signal at the BS can be expressed as

$$\begin{aligned} y_n &= \alpha_{1,n} e^{-j\xi_{1,n}} \beta_n e^{j\theta_n} \alpha_{2,n} e^{-j\xi_{2,n}} x \\ &= g_{1,n}^* \beta_n e^{j\theta_n} g_{2,n} x, \end{aligned} \tag{3}$$

where $g_{1,n}^* = \alpha_{1,n} e^{-j\xi_{1,n}}$ and $g_{2,n} = \alpha_{2,n} e^{-j\xi_{2,n}}$ are the channel coefficients between user to RIS and RIS to BS, respectively. The complete received signal including the LOS transmission and all the N RIS elements can be expressed as

$$y = \left(\sum_{i=1}^{N} \beta_n e^{j\theta_n} g_{1,n}^* g_{2,n} + g_d^* \right) x = (\mathbf{g}_1^H \mathbf{\Theta} \mathbf{g}_2 + g_d^*) x, \tag{4}$$

where $g_d^* \in \mathbb{C}$, $\mathbf{g}_2 \in \mathbb{C}^{N \times 1}$ and $\mathbf{g}_1^H \in \mathbb{C}^{1 \times N}$ are the channels between user to the BS, RIS to BS and user to RIS, respectively. Here, $\Theta = \text{diag}(\beta_1 e^{j\theta_1}, \ldots \beta_N e^{j\theta_N})$ represents the reflection coefficients that we aim to optimize in order to fully leverage the benefits of the RIS.

A. Optimized RIS Coefficient Calculation

The calculation of optimized RIS reflection coefficients is carried out in three main steps: (1) computing the channel gain for each link, (2) determining the phase shift for each RIS element to correctly *steer* the beam toward the BS, and (3) calculating the reflection coefficients to coherently combine the direct and reflected signals at the BS [10]. We assume that the only information required by the RIS controller is the user's location. Since the RIS controller and the BS are stationary, we consider their location information to be known to the controller. Additionally, the array configuration of the RIS—including the number of elements, the number of rows and columns, and the spacing between the elements—is also known to the controller. For the channel gain of each link, we utilize the known carrier frequency f_c to compute the free space path loss and the distance r between the links to calculate the phase shift as

$$g = \frac{\lambda}{4\pi r} e^{j\frac{2\pi r}{\lambda}}. \tag{5}$$

We use (5) to calculate the channel gains g_d^*, $\mathbf{g}_1$ and $\mathbf{g}_2$. Next, the steering vector needs to be calculated for each element of the RIS as

$$\zeta_{i,k} = e^{j\frac{\lambda}{2}\frac{2\pi}{\lambda}(i \sin\psi_a + k \sin\psi_e)}. \tag{6}$$

Here, we assume a uniform planar array (UPA) where elements are placed $\frac{\lambda}{2}$ spacing apart. In addition, ψ_a and ψ_e represent the azimuth and elevation angle between the RIS and either the user or BS. By multiplying the channel gain with the steering vector, we introduce precise directional phase adjustments, ensuring that the reflected signals are directed toward the location of the BS. This results in effective channel gains $\tilde{\mathbf{g}}_1$ and $\tilde{\mathbf{g}}_2$. The final step involves adjusting the phase shift at the RIS to ensure that the reflected signal from the RIS and the direct line-of-sight (LOS) path constructively interfere at the base station. In other words, our objective is to align the RIS-aided path, including the RIS phase shift Θ, with the phase of the direct signal at the BS. This is achieved by

$$\begin{aligned} \angle g_d^* &= \angle \tilde{g}_{1,n} + \theta_n + \angle \tilde{g}_{2,n} \\ \Rightarrow \theta_n &= \angle g_d^* - \angle \tilde{g}_{1,n} - \angle \tilde{g}_{2,n}. \end{aligned} \tag{7}$$

We use this angle for the reflection coefficient of each element of the RIS as $e^{j\theta_n}$.

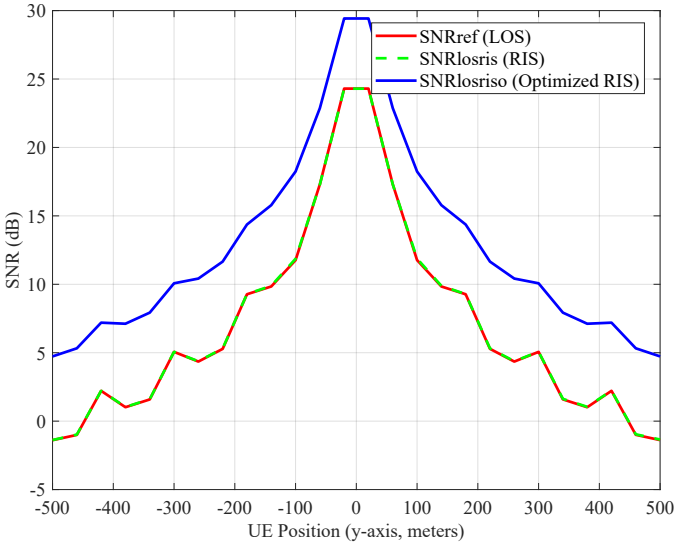

Fig. 2: Received Power vs UE Position for LOS, RIS, and Optimized RIS.

III. Fixed-Point Modeling

In this section, we conducted a fixed-point analysis to determine the minimum word length necessary for effective self-tuning. First, we present the floating-point simulation, which illustrates the relationship between the user's location and the fixed positions of the user, BS and RIS, which are $(a_b = 0, b_b = 0, c_b = 0)$, $(a_r = 50, b_r = 0, c_r = 0)$ in our simulation setup. The RIS consists of $N = 1200$ elements which are arranged in UPA as a 40×30 matrix. if we denote the y-axis position of the user in the vehicle as p, the positions of the user can be represented as $(a_u = 25, b_b = p, c_b = 0)$. For this scenario, we compare the received signal-to-noise ratio (SNR) at the BS across three cases: (1) without RIS, (2) with RIS but without optimized phase angles, and (3) with RIS and optimized phase angles. The user transmits 10000 samples of BPSK signal using $f_c = 39$ GHz carrier frequency. The user moves 1 km along the y-axis, i.e., $p = -500 : 500$. The result of the simulation is shown in Fig. 2. In Fig. 3, we present the fixed point simulation for RIS with an optimized phase.

It is important to note that the RIS does not provide any improvement when the phase angles are not optimized [11]. Hence, the LOS and RIS curves in Fig. 2 have identical performance.However, we observe a significant increase in received power with the optimized RIS, ranging from 2 to 8 dB. The greater the distance of the user from the BS, the more pronounced the benefits of using the RIS become. A fixed-point simulation is presented in Fig. 3. After conducting numerous fixed-point simulations, we identified two key areas where word length significantly affected the performance of the algorithm. The first one is $\mathbf{h}_r^*$, shown in Fig. 3. Note that, even word length with $W = 20$ bits with 19 bits of fraction, the performance lost nearly 1 dB when the user was more than 350 m away from the BS. In our simulation scenario, the values of $\mathbf{g}$ are all ones and does not impact the performance. This is because the BS and RIS are located at the same point along the y-axis. However, for different scenarios where the BS and RIS are not at the same point along the y-axis, the value of $\mathbf{g}$ can also impact the gain of RIS. The second major area where word lengths severely impacted performance is in the look-up tables (LUTs) used for the sine function. The LUTs require at least $W = 16$ bits to not introduce any quantization loss.

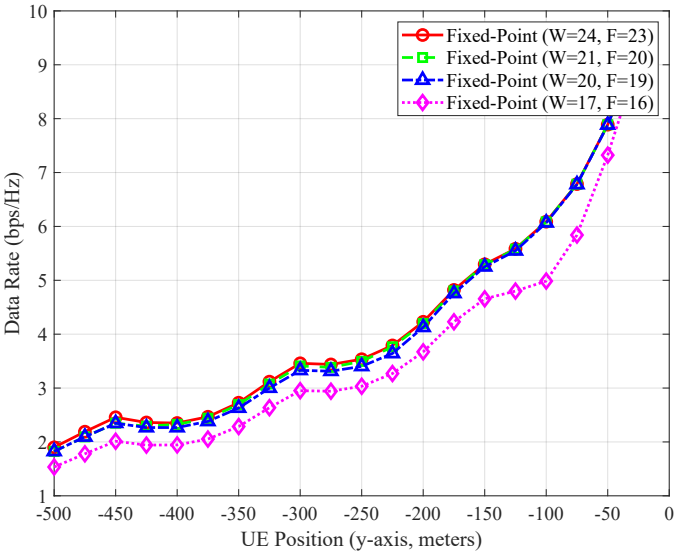

Fig. 3: Achievable Data Rate vs UE Position for Optimized RIS for different fixed-point format.

IV. VLSI Architecture

Similar to Section II-A, we divide the VLSI structure of the RIS coefficient calculation into three steps. Each step in these processes heavily relies on complex exponential calculations. Consequently, the proposed digital VLSI architecture requires a digital implementation for various trigonometric functions. To ensure speed and simplicity, we primarily use look-up tables (LUT) for trigonometric functions over methods such as CORDIC and Taylor series expansion in this work. The inputs to the proposed architecture include the relative distances, as well as the azimuth and elevation angles of the user, RIS, and BS. The output of the VLSI architecture is the optimized N RIS coefficients which are computed in a serial fashion according to the algorithm in Section II-A. The inputs are first utilized in a gain calculation block which consists of two small LUTs of 2^4 addresses for sin and cos functions. Each address in these LUTs has a width of 8 bits. For each LUT in this design, the size and number of address locations are carefully chosen based on fixed-point simulation results. The $\frac{\lambda}{4\pi}$ and $\frac{2\pi}{\lambda}$ values are used as precomputed parameters. The gain calculation block is replicated three times for g_d^*, $g_{1,n}^*$ and $g_{2,n}$ respectively. These three values will be the same for the next N values coming out of the steering vector calculation engine. Since the circuitry required for gain calculation is idle most of the time, this represents a significant waste of resources. It should be investigated whether gain calculations can be performed in the BS software and transferred to the RIS with an acceptable delay.

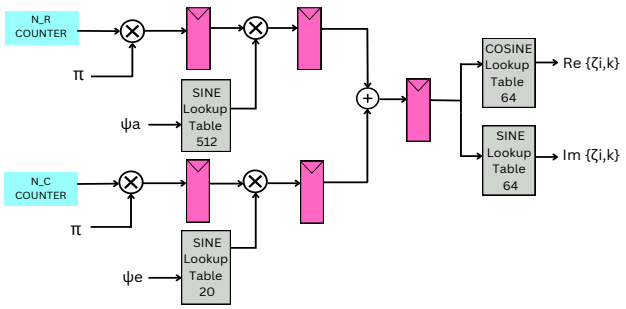

Fig. 4: Block diagram of the steer vector computation engine

Equation 6 presents the next step of our VLSI design which can be expanded using Euler's formula as

$$\zeta_{i,k} = \cos(i\pi \sin\psi_a + k\pi \sin\psi_e) + j \sin(i\pi \sin\psi_a + k\pi \sin\psi_e)$$

In an ideal situation, a single large LUT that covers all the required input ranges would be adopted for (6). However, our fixed-point analysis reveals that a single LUT becomes more costly than using several optimized LUTs operating in parallel. For example, the single LUT needs to support an input range of $[-17, 0]$ with a step-size of 2^{-7}, i.e., an LUT of 2048 addresses. However, by utilizing optimized LUTs for each variable, we need four LUTs with 512, 20, 64, and 64 addresses, respectively, which are significantly less expensive together than the large LUT. It is important to note that techniques such as the symmetric bipartite table method (SBTM) or the symmetric table addition method (STAM) could be used to reduce the size of the large LUT [12], [13]. However, this would increase the latency of the steering vector engine and complicate the control logic. A block diagram of the steering vector computation engine consisting of the four LUTs mentioned above is shown in Fig. 4. Registers are inserted after each multiplication to shorten the critical path. The outputs from this engine are multiplied with $g^*_{1,n}$ and $g_{2,n}$ using complex multipliers to achieve effective channel gain. Therefore, we need two steering vector computation engines in our design. The final step is to calculate the angles from the complex values of the effective channel gains and determine the reflection coefficient through subtraction. We use $\arctan$ functions to calculate the angles of complex channel gains. Similar to $\sin$ and $\cos$, we apply LUT based $\arctan$ calculations. Three such LUTs are needed for the three effective channel gain inputs. The outputs of these LUTs are subtracted according to (7). Two more $\sin$ LUTs are used to obtain the final result $e^{j\theta_n}$. From the fixed-point simulation, it is enough to have 8 bits for each address of these two final $\sin$ LUTs.

V. FPGA Implementation

The VLSI architecture was developed using SystemVerilog for an Artix-7 XC7A100TCSG324 FPGA device with a speed grade of -1. We use a 50 MHz clock for synthesis and implementation with a flattening setting of *none*. Note that, for every $\frac{\lambda}{2}$, movement the channel coefficients need to be updated. As we assume a $f_c = 39$ GHz and the travel distance of the user as 1 km in our system model, the channel and the RIS reflection coefficients need to be updated for every movement of 3.84 mm. If we have to calculate $N = 1200$ reflection coefficients for every 3.84 mm movement, a total of 8.64 million coefficients must be calculated every second for a vehicle traveling at 100 km/h. Therefore, a clock frequency of 9 MHz or higher is sufficient, as each clock cycle can output one reflection coefficient. Our target platform, the Arty A7, includes four peripheral module interface (PMOD) connectors, each supporting 4 bits, allowing us to output a complex vector consisting of 8-bit real and 8-bit imaginary numbers.

TABLE I: Resource utilization of the proposed VLSI architecture in an Artix-7 XC7A100TCSG324-1 Device

Component Type	Numbers Used	Percentage Used
Slice LUTs	3885	6.13%
Slice Registers	518	1%
Block RAM	4	1.48%
DSP48E1	17	7.08%
Bonded IOB	86	40.95%

The resource utilization of our proposed design is shown in Table I. Due to the low number of input-output block (IOB) in the device, the utilization is pretty high, more than 40%. The DSP48E1 slices are associated with the use of complex multipliers in the design. Block Random Access Memory (BRAM) is utilized for the large LUT blocks in our architecture. The rest of the design is supported by slice LUTs and registers. To the best of our knowledge, there is currently no hardware implementation of the RIS controller available in the literature, preventing us from making a comparison with the state-of-the-art.

VI. Conclusion

In this paper, we delved into the design and implementation of a digital VLSI architecture for a self-tuning RIS controller, focusing on its potential to enhance vehicular communications. Our simulations and fixed-point modeling demonstrated that the self-tuning RIS significantly improves the SNR for users positioned farther away from the base station. We also presented a VLSI architecture and FPGA implementation of the RIS that supports vehicular communication of 100 kmh. Looking ahead, future studies should explore the integration of advanced machine learning algorithms to further optimize the RIS's performance, ensuring it can adapt to the ever-evolving demands of next-generation networks.

References

[1] M. A. ElMossallamy, H. Zhang, L. Song, K. G. Seddik, Z. Han, and G. Y. Li, "Reconfigurable Intelligent Surfaces for Wireless Communications: Principles, Challenges, and Opportunities," *IEEE Transactions on Cognitive Communications and Networking*, vol. 6, no. 3, pp. 990–1002, 2020.

[2] Q. Wu, S. Zhang, B. Zheng, C. You, and R. Zhang, "Intelligent Reflecting Surface-Aided Wireless Communications: A Tutorial," *IEEE transactions on communications*, vol. 69, no. 5, pp. 3313–3351, 2021.

[3] E. Basar, M. Di Renzo, J. De Rosny, M. Debbah, M.-S. Alouini, and R. Zhang, "Wireless Communications Through Reconfigurable Intelligent Surfaces," *IEEE Access*, vol. 7, pp. 116 753–116 773, 2019.

[4] L. Yang, Y. Yang, M. O. Hasna, and M.-S. Alouini, "Coverage, Probability of SNR Gain, and DOR Analysis of RIS-aided Communication Systems," *IEEE Wireless Communications Letters*, vol. 9, no. 8, pp. 1268–1272, 2020.

[5] E. Shi, J. Zhang, H. Du, B. Ai, C. Yuen, D. Niyato, K. B. Letaief, and X. Shen, "RIS-Aided Cell-Free Massive MIMO Systems for 6G: Fundamentals, System Design, and Applications," *Proceedings of the IEEE*, vol. 112, no. 4, pp. 331–364, 2024.

[6] S. Basharat, S. A. Hassan, H. Pervaiz, A. Mahmood, Z. Ding, and M. Gidlund, "Reconfigurable Intelligent Surfaces: Potentials, Applications, and Challenges for 6G Wireless Networks," *IEEE Wireless Communications*, vol. 28, no. 6, pp. 184–191, 2021.

[7] A. I. Abdulshakoor, N. A. Ali, and H. S. Hassanein, "A Comprehensive Analysis of Packet Delay in Reconfigurable Intelligent Surface-Assisted Communication Networks," in *ICC 2024-IEEE International Conference on Communications*. IEEE, 2024, pp. 1691–1696.

[8] M. Deng, M. Ahmed, A. Wahid, A. A. Soofi, W. U. Khan, F. Xu, M. Asif, and Z. Han, "Reconfigurable Intelligent Surfaces Enabled Vehicular Communications: A Comprehensive Survey of Recent Advances and Future Challenges," *IEEE Transactions on Intelligent Vehicles*, 2024.

[9] W. Shi, H. Jiang, B. Xiong, X. Chen, H. Zhang, Z. Chen, and Q. Wu, "RIS-empowered V2V Communications: Three-dimensional Beam Domain Channel Modeling and Analysis," *IEEE Transactions on Wireless Communications*, 2024.

[10] N. S. Perović, L.-N. Tran, M. Di Renzo, and M. F. Flanagan, "Optimization of RIS-aided MIMO Systems Via the Cutoff Rate," *IEEE Wireless Communications Letters*, vol. 10, no. 8, pp. 1692–1696, 2021.

[11] S. Zeng, H. Zhang, B. Di, Z. Han, and L. Song, "Reconfigurable intelligent surface (ris) assisted wireless coverage extension: Ris orientation and location optimization," *IEEE Communications Letters*, vol. 25, no. 1, pp. 269–273, 2020.

[12] M. J. Schulte and J. E. Stine, "Symmetric Bipartite Tables for Accurate Function Approximation," in *Proceedings 13th IEEE Sympsoium on Computer Arithmetic*. IEEE, 1997, pp. 175–183.

[13] J. E. Stine and M. J. Schulte, "The Symmetric Table Addition Method for Accurate Function Approximation," *Journal of VLSI signal processing systems for signal, image and video technology*, vol. 21, pp. 167–177, 1999.