Received 24 March 2021; accepted 23 April 2021. Date of publication 28 April 2021; date of current version 24 May 2021.

Digital Object Identifier 10.1109/OJCOMS.2021.3075910

Alternative Codes and Phase Rotation Extensions for Alternating Space-Time Coding-Based Physical Layer Security

MICHAEL R. CRIBBS (Graduate Student Member, IEEE), RIC A. ROMERO (Senior Member, IEEE), AND TRI T. HA (Life Fellow, IEEE)

Department of Electrical and Computer Engineering, Naval Postgraduate School, Monterey, CA 93943, USA

CORRESPONDING AUTHOR: M. R. CRIBBS (e-mail: mrcribbs@hotmail.com)

ABSTRACT In this article, several extensions to an alternating space-time (ST) code (STC) physical layer security (PLS) scheme from the literature are proposed. These contributions include alternative orthogonal STCs as well as non-orthogonal and spatially-multiplexed (SM) STCs for improved bandwidth efficiency. Phase rotation (PR) algorithms are provided to build very large sets of unique alternative STCs for use with this scheme. Decoding methods are discussed for alternating non-orthogonal and SM STCs. Monte Carlo simulations are provided to compare bit error rate (BER) performance between different decoding methods. Secrecy system nomenclature is adapted to the alternative STCs and proposed algorithms. Information-theoretic security analysis including message and key equivocation is provided along with expected eavesdropper BER based on an assumed attack methodology. A comparison of security offered by the alternating STC PLS scheme with and without incorporation of proposed PR algorithms is performed. Substantially greater exhaustive key search attack complexity is achieved by using the PR algorithms proposed in this article.

INDEX TERMS Key equivocation, key residue class, message equivocation, phase rotation, physical layer security, space-time coding.

I. INTRODUCTION & MOTIVATION

INFORMATION security is of the utmost importance in the ultra-connected world in which we live today. Whereas communications security is often performed via higher layer cryptography, advances in digital signal processing have made physical layer security (PLS) solutions a desirable area of research in recent years. Many common PLS schemes rely on disparity in quality or characteristics between the main and eavesdropper channels to secure communications or generate symmetric keys at the physical layer [1]–[6]. Alternatively, there are numerous other approaches that rely on combinations of channel characteristics or pre-shared secrets with elements of computational complexity to achieve PLS [7]–[21]. A framework is established in [8] for implementation and analysis of space-time (ST) coding-based PLS schemes that make use of large sets of orthogonal ST codes (STCs) originally discussed in [7].

The contributions of this article are to extend this framework:

- to alternative orthogonal STCs from the literature,
- to non-orthogonal and spatially-multiplexed (SM) STCs offering improved bandwidth efficiency,
- to include phase rotation (PR) algorithms inspired from [11] to build significantly larger STC sets,
- and to provide comparative information-theoretic and bit error rate (BER) analysis to demonstrate increases in security achieved by incorporation of these extensions.

The remainder of this article is outlined as follows: In Section II, alternative orthogonal STCs are presented. In Section III, PR algorithm extensions are discussed with updated STC set cardinalities. In Section IV, the maximal ratio combining (MRC) matrix generation algorithm from [8] is revised. In Section V, adaptations are provided for non-orthogonal and SM STCs. In Section VI, secrecy system

nomenclature is revisited to allow for incorporation of PR algorithms and alternative STCs. In Section VII, information-theoretic security of the adapted alternating STC PLS scheme is analyzed along with expected eavesdropper (*aka* "Eve") BER. Primary channel BER performance using different decoding methods from Section V is also compared. In Section VIII, we discuss additional security provided by incorporation of PR algorithms. In Section IX, we recap the contents of this article. Appendices are also included to provide additional details as needed.

II. ALTERNATIVE ORTHOGONAL SPACE-TIME CODES

In this section, we present two alternative orthogonal STCs that may be used in conjunction with the alternating STC PLS scheme first presented in [8] with a few adaptations.

The first alternative code, referred to as multi-pair orthogonal 3-3-4 (MPO334) from this point forward, is a complex orthogonal 3/4 rate STC represented as

$$\mathbf{G} = \begin{bmatrix} s_1 & s_2 & \frac{s_3}{\sqrt{2}} \\ -s_2^* & s_1^* & \frac{s_3}{\sqrt{2}} \\ \frac{s_3^*}{\sqrt{2}} & \frac{s_3^*}{\sqrt{2}} & \frac{(-s_1-s_1^*+s_2-s_2^*)}{2} \\ \frac{s_3^*}{\sqrt{2}} & \frac{-s_3^*}{\sqrt{2}} & \frac{(s_2+s_2^*+s_1-s_1^*)}{2} \end{bmatrix}, \tag{1}$$

using three transmit (TX) antennas to transmit three data symbols, s_1 through s_3, over four symbol time periods.

Let c represent the number of columns in $\mathbf{G}$, which is the number of TX antennas employed. Likewise, r is used to represent the number of rows in $\mathbf{G}$, which is the number of symbol time periods. The number of symbols per codeword is represented by k, where codeword refers to a single block of encoded symbols as dictated by the chosen STC matrix. The data symbol vector for any given STC refers to $\mathbf{s} = [s_1 \ \ldots \ s_k]^T$, where T represents the transpose operation. An extended data symbol vector, $\mathbf{s}_{ext} = [\mathbf{s}^T \ \mathbf{s}^\dagger]^T$, is also referenced, where $\dagger$ represents the conjugate transpose operation. These definitions remain consistent throughout this article.

A variation of MPO334, referred to as multi-pair orthogonal 4-3-4 (MPO434) from this point forward, is a complex orthogonal 3/4 rate STC represented as

$$\mathbf{G} = \begin{bmatrix} s_1 & s_2 & \frac{s_3}{\sqrt{2}} & \frac{s_3}{\sqrt{2}} \\ -s_2^* & s_1^* & \frac{s_3}{\sqrt{2}} & \frac{-s_3}{\sqrt{2}} \\ \frac{s_3^*}{\sqrt{2}} & \frac{s_3^*}{\sqrt{2}} & \frac{(-s_1-s_1^*+s_2-s_2^*)}{2} & \frac{(-s_2-s_2^*+s_1-s_1^*)}{2} \\ \frac{s_3^*}{\sqrt{2}} & \frac{-s_3^*}{\sqrt{2}} & \frac{(s_2+s_2^*+s_1-s_1^*)}{2} & \frac{(-s_1-s_1^*-s_2+s_2^*)}{2} \end{bmatrix}, \tag{2}$$

using four TX antennas to transmit three data symbols, s_1 through s_3, over four symbol time periods. MPO334 and MPO434 were originally given in [22] and referred to as sporadic 3/4 rate STCs.

A. THREE DIMENSIONAL MATRICES

A slight modification in representation is required to both MPO334 and MPO434 in order to continue using set building algorithms presented in [7] and [8]. These STCs must be translated into three dimensional matrices that are subsequently summed along the third dimension to return to the original forms given in (1) and (2) prior to transmission. By translating each matrix into three dimensions, each element of the resulting matrix contains only a single weighted data symbol variation. This representation becomes compatible for use with set building algorithms in [7] and [8].

As MPO334 and MPO434 contain matrix elements with sums of four symbol variations, the third dimension depth, denoted as d from this point forward, for the resulting matrices is four. Thus, MPO334 is translated into a three dimensional matrix with slices of

$$\mathbf{G}_{3D}(:,\ :,\ 1) = \begin{bmatrix} s_1 & s_2 & \frac{s_3}{\sqrt{2}} \\ -s_2^* & s_1^* & \frac{s_3}{\sqrt{2}} \\ \frac{s_3^*}{\sqrt{2}} & \frac{s_3^*}{\sqrt{2}} & \frac{-s_1}{2} \\ \frac{s_3^*}{\sqrt{2}} & \frac{-s_3^*}{\sqrt{2}} & \frac{s_2}{2} \end{bmatrix}, \tag{3}$$

$$\mathbf{G}_{3D}(:,\ :,\ 2) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{-s_1^*}{2} \\ 0 & 0 & \frac{s_2^*}{2} \end{bmatrix}, \tag{4}$$

$$\mathbf{G}_{3D}(:,\ :,\ 3) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{s_2}{2} \\ 0 & 0 & \frac{s_1}{2} \end{bmatrix}, \text{ and} \tag{5}$$

$$\mathbf{G}_{3D}(:,\ :,\ 4) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{-s_2^*}{2} \\ 0 & 0 & \frac{-s_1^*}{2} \end{bmatrix}. \tag{6}$$

A four dimensional set of unique three dimensional matrices may be built by performing all set building algorithms using $\mathbf{G}_{3D}$ as the base code. All operations must be repeated as necessary for each slice of the three dimensional matrix. As the set building algorithms are designed to build three dimensional sets of two dimensional matrices, set indexing within each algorithm must be modified accordingly.

B. SET BUILDING

For both MPO334 and MPO434, the symbol permutation algorithm in [8] is permitted with allowed symbol permutations vector ($\mathbf{v}_{asp}$) of

$$\mathbf{v}_{asp} = [1 \ \ 2 \ \ 6] \tag{7}$$

to prevent building duplicate codes. Elements within $\mathbf{v}_{asp}$ represent allowed permutation indices. With the full symbol permutations array shown in Fig. 1, it can be seen

Permutation Index	
1	1 2 3
2	1 3 2
3	2 1 3
4	2 3 1
5	3 1 2
6	3 2 1

FIGURE 1. Full symbol permutations array.

that elements given in (7) allow for the original symbol permutation (i.e., [1 2 3]), swapping symbols s_2 and s_3 (i.e., [1 3 2]), or swapping symbols s_1 and s_3 (i.e., [3 2 1]) for these two alternative STCs.

For both of these codes, the symbol negation algorithm in [8] is also permitted with allowed symbol negations vector $(\mathbf{v}_{asn}) = [1]$.

The cardinality of a set of STCs built by employing all set building algorithms with corresponding operation limits is given in [8, eq. (7)] as

$$\left|\mathbf{S}\right| = 2^{r+c+k-1} \cdot r! \cdot c! \cdot |\mathbf{v}_{asp}| \cdot 2^{|\mathbf{v}_{asn}|}, \tag{8}$$

where $|\mathbf{v}|$ represents the length of a vector $\mathbf{v}$. This assumes that all $c!$ column permutations are permitted; however, due to the similarity of particular columns in these alternative STCs, the column permutations algorithm must be limited to prevent duplicate codes from being built. Due to this fact, we replace this term with the length of the allowed column permutations vector ($\mathbf{v}_{acp}$). Thus, the revised set cardinality formula becomes

$$\left|\mathbf{S}\right| = 2^{r+c+k-1} \cdot r! \cdot |\mathbf{v}_{acp}| \cdot |\mathbf{v}_{asp}| \cdot 2^{|\mathbf{v}_{asn}|}, \tag{9}$$

where $|\mathbf{v}_{acp}| = c!$ for the single-pair orthogonal 3-3-4 (SPO334) and multi-pair orthogonal 4-6-8 (MPO468) STCs discussed in [8].

For the MPO334 STC, column permutations that are similar to a previous permutation but with only the first and second columns swapped are not permitted. Using Fig. 1 for reference as $c = k$ for this STC, only permutation indices one, two, and five are permitted, i.e., $\mathbf{v}_{acp} = [1\ 2\ 5]$. Thus, set cardinality for the MPO334 STC using (9) is $|\mathbf{S}| =$ 221,184 with $c = 3$, $k = 3$, $r = 4$, $|\mathbf{v}_{acp}| = 3$, $|\mathbf{v}_{asp}| = 3$, and $|\mathbf{v}_{asn}| = 1$.

Similarly for the MPO434 STC, column permutations that are similar to a previous permutation but with only the first and second columns swapped or third and fourth columns swapped are not permitted. Thus, set cardinality for the MPO434 STC using (9) is $|\mathbf{S}| = 884,736$ with $c = 4$, $k = 3$, $r = 4$, $|\mathbf{v}_{acp}| = 6$, $|\mathbf{v}_{asp}| = 3$, and $|\mathbf{v}_{asn}| = 1$.

For reference, an updated column permutations algorithm employing $\mathbf{v}_{acp}$ is provided in Appendix A.

C. RECEPTION

As the alternative STCs presented in this section are orthogonal, MRC may be performed by a receiver employing as few as one receive (RX) antenna. Due to being three dimensional STC matrices and having non-unit-energy data symbol weights, the algorithm provided in [8] for determining the MRC matrix to be employed must be modified to work with these alternative STCs. Algorithm 8 is revised in Section IV for this purpose and to accommodate PR algorithms presented in Section III.

Upon determining the MRC matrix, the sufficient statistic and decoding steps given in [8] are then followed to obtain an estimate of the data symbol vector, $\hat{\mathbf{s}}$. Examples for MRC matrix determination with the MPO334 STC are provided in Section IV. Proof of the sufficient statistic and MRC sequence for the MPO334 STC are provided in Appendix B.

III. PHASE ROTATIONS VICE NEGATIONS

Set building algorithms presented in [7] and revised in [8] included row, column, and symbol negations. These algorithms conducted all permitted combinations of row, column, and symbol negations, respectively, on all STCs contained in the set at start of execution. After each algorithm is performed, the number of STCs contained within the set is equal to the number of STCs in the set at start of execution multiplied by the number of negation combinations conducted. Thus, the full set of STCs was built in this manner.

These three algorithms may be extended to more generic PR algorithms to build much larger sets of STCs. This leads to enhanced security through increased attack complexity discussed further in Section VIII. Whereas this may seem to be a straight forward extension, as negation corresponds to a π radian PR, we discuss a few details in this section that must be considered when implementing this extension. For completeness, revised Algorithms 1, 2, and 7, incorporating PRs, are provided in Appendix A.

A. SELECTION OF PHASE ROTATION ANGLES

For the purpose of simplicity, we propose a PR scheme with selection of equally-spaced PR angles around the complex-plane. To that end, we define a variable $a \geq 1$ such that 2^a equally-spaced PR angles are produced of the form

$$\frac{2\pi x}{2^a} \tag{10}$$

radians, where $x \in \{0, \ldots, 2^a - 1\}$ is the PR angle index. When $a = 1$, this PR scheme is equivalent to negation.

B. SYMBOL-WISE PHASE ROTATION

When implementing a symbol-wise PR algorithm, it must be understood that orthogonal STC matrices consist of conjugated and unconjugated symbol variations. To maintain orthogonality while performing symbol-wise PRs, these dissimilarly-conjugated symbol variations must receive opposing PRs. Thus, if unconjugated variations of a given data symbol are rotated by ϕ radians, conjugated variations of that same data symbol must be rotated by $-\phi$ radians.

As with the symbol negations algorithm, it is important to note which data symbols may be phase-rotated, individually

TABLE 1. Set cardinality of various STCs for various values of *a* when incorporating PR algorithms.

STC	c	k	r	$\lvert\mathbf{v}_{acp}\rvert$	$\lvert\mathbf{v}_{asp}\rvert$	$\lvert\mathbf{v}_{asr}\rvert$	$\lvert\mathbf{S}\rvert$ $(a=1)$	$\lvert\mathbf{S}\rvert$ $(a=2)$	$\lvert\mathbf{S}\rvert$ $(a=3)$	$\lvert\mathbf{S}\rvert$ $(a=9)$
Alamouti	2	2	2	NA	NA	NA	64	512	4,096	$1.07\cdot10^{9}$
SPO334	3	3	4	6	1	0	73,728	4,718,592	301,989,888	$2.08\cdot10^{19}$
MPO468	4	6	8	24	15	2	$7.61\cdot10^{12}$	$6.23\cdot10^{16}$	$5.11\cdot10^{20}$	$1.54\cdot10^{44}$
MPO334	3	3	4	3	3	1	221,184	28,311,552	$3.62\cdot10^{9}$	$1.59\cdot10^{22}$
MPO434	4	3	4	6	3	1	884,736	226,492,416	$5.80\cdot10^{10}$	$1.63\cdot10^{25}$

or concurrently, without resulting in duplicate codes. An allowed symbol PRs vector, denoted as $\mathbf{v}_{asr}$, is used for this purpose within the symbol PR algorithm.

C. SYMBOL CONJUGATIONS WITH PHASE ROTATIONS

A symbol conjugations algorithm is presented in [7] and revised in [8] in which the entire STC element at a given matrix location is conjugated. In order to understand key residues (KRs) when PRs are applied, this algorithm must be revised to conjugate only the symbol itself without affecting any applied PR. The revised algorithm is provided in Appendix A.

D. UPDATED SET CARDINALITY

By incorporating PR algorithms, cardinality of $\mathbf{S}$ when employing SPO334, MPO468, MPO334, or MPO434 as the base code becomes

$$\left|\mathbf{S}\right| = 2^{a(r+c-1)+k}\cdot r!\cdot\left|\mathbf{v}_{acp}\right|\cdot\left|\mathbf{v}_{asp}\right|\cdot 2^{a\left|\mathbf{v}_{asr}\right|}. \quad (11)$$

Alternatively, cardinality of $\mathbf{S}$ when employing the Alamouti base code becomes

$$\left|\mathbf{S}\right| = 2^{a(r+c-1)+k-1}\cdot r!\cdot c! = 8^{a+1}. \quad (12)$$

Cardinality of sets built with these STCs incorporating PR algorithms with various values of a is provided in Table 1.

IV. MAXIMAL RATIO COMBINING MATRIX GENERATION

In [8], Algorithm 8 is provided to generate a MRC matrix for use with the orthogonal STCs and alternating STC PLS scheme discussed in that work. In this section, Algorithm 8 is revised to work with the alternative STCs and PR algorithms discussed herein.

Example 1: The $\mathbf{H}_C$ matrix generated using Algorithm 8 for the MPO334 base code in (1) and channel tap vector, $\mathbf{h} = [h_1 \;\; h_2 \;\; h_3]^T$, is given in transposed form as

$$\mathbf{H}_C = \begin{bmatrix} h_1 & 0 & \frac{-h_3}{2} & \frac{h_3}{2} \\ h_2 & 0 & \frac{h_3}{2} & \frac{h_3}{2} \\ \frac{h_3}{\sqrt{2}} & \frac{h_3}{\sqrt{2}} & 0 & 0 \\ 0 & h_2 & \frac{-h_3}{2} & \frac{-h_3}{2} \\ 0 & -h_1 & \frac{-h_3}{2} & \frac{h_3}{2} \\ 0 & 0 & \frac{(h_1+h_2)}{\sqrt{2}} & \frac{(h_1-h_2)}{\sqrt{2}} \end{bmatrix}^T. \quad (13)$$

Algorithm 8 MRC Matrix Generation

Input: $\mathbf{G}$ {STC Matrix}, $\mathbf{h}$ {Channel tap vector}
Output: $\mathbf{H}_C$ {MRC Matrix}

$\mathbf{H}_C$ = 2-dimensional array of 0's of size r-by-$2k$
for $di = 1$ to d **do**
  for $ri = 1$ to r **do**
    for $ci = 1$ to c **do**
      if $\mathbf{G}(ri, ci, di) \neq 0$ **then**
        $ki =$ symbol index of $\mathbf{G}(ri, ci, di)$
        {e.g., $ki = 1$ for $\mathbf{G}(ri, ci, di) == \frac{-s_1^*}{\sqrt{2}}$}
        $w =$ positive weight applied to $\mathbf{G}(ri, ci, di)$
        {e.g., $w = \frac{1}{\sqrt{2}}$ for $\mathbf{G}(ri, ci, di) == \frac{-s_1^*}{\sqrt{2}}$}
        $\phi =$ PR angle applied to $\mathbf{G}(ri, ci, di)$
        {e.g., $\phi = \pi$ for $\mathbf{G}(ri, ci, di) == \frac{-s_1^*}{\sqrt{2}}$}
        if $\mathbf{G}(ri, ci, di)$ is conjugated **then**
          $g = 1$
        else
          $g = 0$
        end if
        $\mathbf{H}_C(ri, ki + g\cdot k) = \mathbf{H}_C(ri, ki + g\cdot k) + w\cdot e^{j\phi}\cdot\mathbf{h}(ci)$
      end if
    end for
  end for
end for
return $\mathbf{H}_C$

Example 2: A variant of the MPO334 STC is generated using PR Algorithms 1, 2, and 7, given in Appendix A, with $a = 3$. Within Algorithm 1, the first row of the base STC in (1) is rotated with a PR angle index of $x = 1$. Within Algorithm 2, the second column is rotated with a PR angle index of $x = 2$. Within Algorithm 7, symbol s_1 is rotated with a PR angle index of $x = 3$. Thus, the resulting variant of the MPO334 STC is represented as

$$\mathbf{G} = \begin{bmatrix} -s_1 & s_2\cdot e^{j3\pi/4} & \frac{s_3\cdot e^{j\pi/4}}{\sqrt{2}} \\ -s_2^* & s_1^*\cdot e^{j7\pi/4} & \frac{s_3}{\sqrt{2}} \\ \frac{s_3^*}{\sqrt{2}} & \frac{s_3^*\cdot e^{j\pi/2}}{\sqrt{2}} & \frac{(s_1\cdot e^{j7\pi/4}+s_1^*\cdot e^{j\pi/4}+s_2-s_2^*)}{2} \\ \frac{s_3^*}{\sqrt{2}} & \frac{s_3^*\cdot e^{j3\pi/2}}{\sqrt{2}} & \frac{(s_2+s_2^*+s_1\cdot e^{j3\pi/4}+s_1^*\cdot e^{j\pi/4})}{2} \end{bmatrix}. \quad (14)$$

With this STC and channel tap vector, $\mathbf{h} = [h_1 \;\; h_2 \;\; h_3]^T$, the corresponding $\mathbf{H}_C$ matrix generated using Algorithm 8 is given in (15), shown at the bottom of the page.

V. NON-ORTHOGONAL AND SPATIALLY-MULTIPLEXED SPACE-TIME CODES

In this section, we present alternative non-orthogonal and SM STCs that may be used in conjunction with the alternating STC PLS scheme [8]. Various necessary adaptations are discussed.

In a desire to improve bandwidth efficiency, modern research is moving away from transmit diversity towards spatial multiplexing [23]. A middle ground between orthogonal STCs and spatial multiplexing is non-orthogonal STCs which still exhibit some level of transmit diversity. The latest IEEE standard for air interface for broadband wireless access systems includes optional orthogonal, non-orthogonal, and SM STCs [24]. To extend the set building and PLS techniques presented in [7], [8] to these other types of STCs, a suitable decoding methodology must be determined as MRC is only possible for orthogonal STCs.

Receivers of SM or non-orthogonal STCs typically employ sorted QR decomposition (SQRD) along with a successive interference cancellation (SIC) decoder and/or sphere decoding (SD) [25]–[29]. We use these decoding techniques and adapt them as necessary to work with an alternating STC.

To explain this adaptation, we begin with a multiple-input multiple-output (MIMO) system using spatial multiplexing with two TX antennas and two RX antennas. This system employs the alternating STC PLS scheme adapted for use with these other types of STCs [8]. The scheme is supplemented with PR algorithms presented in Appendix A with $a = 6$. Updated Algorithm 8 in Section IV is also used with this system to obtain an initial matrix for use with SQRD and follow-on SD techniques.

The base STC chosen for this system, referred to as SM 2-2-1 (SM221) from this point forward, is a complex rate 2 SM STC (SMSTC) represented as

$$\mathbf{G} = [s_1 \quad s_2], \tag{16}$$

using two TX antennas to transmit two data symbols, s_1 and s_2, over one symbol time period. As this is a SMSTC, there is no transmit diversity being employed.

Without loss of generality, the received sample array, $\mathbf{Z}$, is of size r-by-N_R and represented as

$$\mathbf{Z} = \mathbf{G} \times \mathbf{h}_{BU} + \mathbf{n}, \tag{17}$$

where N_R represents the number of RX antennas, $\mathbf{G}$ is the next alternating STC, $\mathbf{h}_{BU}$ is the c-by-N_R channel tap array between base station (BS) and user equipment (UE), and $\mathbf{n}$ is a r-by-N_R additive white Gaussian noise (AWGN) array. The term array is used here to indicate a component that may be a vector or matrix depending on the STC in use and value of N_R.

For this discussion, the next STC is represented as

$$\mathbf{G} = \begin{bmatrix} s_2 \cdot e^{j\pi/8} & s_1^* \cdot e^{j5\pi/32} \end{bmatrix}, \tag{18}$$

where it can be seen that multiple operations have been applied to the base code in (16).

To begin the decoding process, Algorithm 8 is repeated using channel state information (CSI) for each RX antenna with resulting $\mathbf{H}_C$ matrices concatenated vertically to obtain a final $\mathbf{H}_{CBU}$ matrix of size rN_R-by-$2k$. The channel tap vector input for the ith RX antenna is represented as $\mathbf{h}_{BU}(:, i) = [h_{1i} \;\; \ldots \;\; h_{ci}]^T$. Thus, performing these steps for the STC in (18) for the described MIMO system employing two RX antennas, the final $\mathbf{H}_{CBU}$ matrix is given as

$$\mathbf{H}_{CBU} = \begin{bmatrix} 0 & h_{11} \cdot e^{j\pi/8} & h_{21} \cdot e^{j5\pi/32} & 0 \\ 0 & h_{12} \cdot e^{j\pi/8} & h_{22} \cdot e^{j5\pi/32} & 0 \end{bmatrix}. \tag{19}$$

The remaining approach differs depending on whether or not the STC in use employs transmit diversity. We now present how the remaining portion of this decoding approach differs for these two different types of STCs. Block diagrams are shown in Fig. 2 to illustrate decoding steps for both types of STCs. The dashed arrows in each block diagram represent two optional paths of which only one must be taken.

A. DECODING SPACE-TIME CODES WITHOUT TRANSMIT DIVERSITY

As the STC given in (16) does not employ transmit diversity, we start by discussing the remaining decoding steps required for SMSTCs.

First, we form a restructured $\mathbf{H}_{CBU}$ matrix, denoted as $\mathbf{H}_{RC}$, while noting which symbols are conjugated. We then solve for the symbols, or conjugate symbols, using SQRD and SIC to obtain an estimate of the transmitted symbol vector. SD techniques may then be employed to obtain a near maximum-likelihood (ML) result. Any symbols noted as being conjugated initially must be conjugated again at the end of this process to solve for the original data symbol vector.

The process of forming $\mathbf{H}_{RC}$ while noting which symbols are conjugated starts by identifying the columns within the

$$\mathbf{H}_C = \begin{bmatrix} -h_1 & h_2 \cdot e^{j3\pi/4} & \frac{h_3 \cdot e^{j\pi/4}}{\sqrt{2}} & 0 & 0 & 0 \\ 0 & 0 & \frac{h_3}{\sqrt{2}} & h_2 \cdot e^{j7\pi/4} & -h_1 & 0 \\ \frac{h_3 \cdot e^{j7\pi/4}}{2} & \frac{h_3}{2} & 0 & \frac{h_3 \cdot e^{j\pi/4}}{2} & \frac{-h_3}{2} & \frac{(h_1 + h_2 \cdot e^{j\pi/2})}{\sqrt{2}} \\ \frac{h_3 \cdot e^{j3\pi/4}}{2} & \frac{h_3}{2} & 0 & \frac{h_3 \cdot e^{j\pi/4}}{2} & \frac{h_3}{2} & \frac{(h_1 + h_2 \cdot e^{j3\pi/2})}{\sqrt{2}} \end{bmatrix} \tag{15}$$

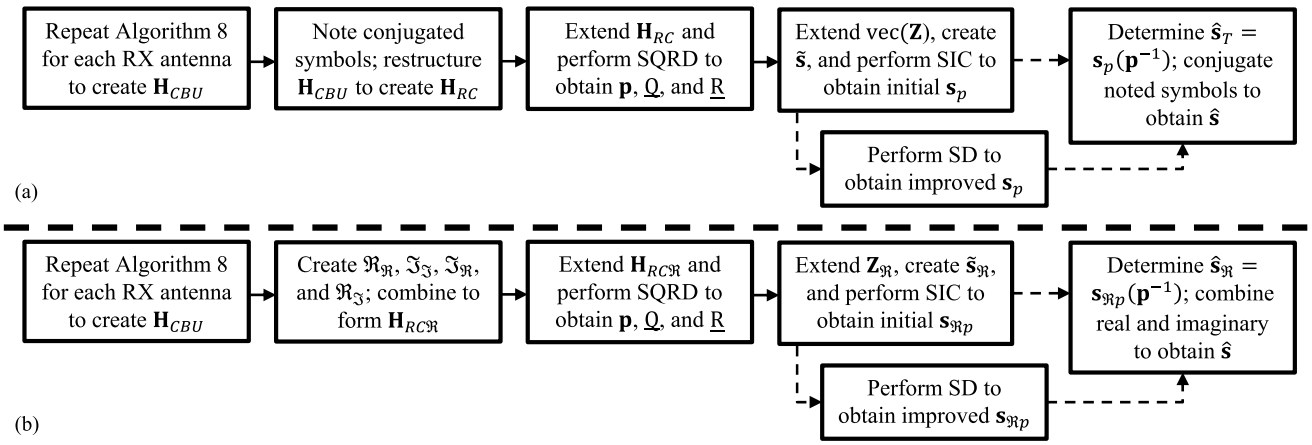

FIGURE 2. Block diagrams illustrating decoding steps for non-orthogonal and SM STCs. (a) STCs without transmit diversity; (b) STCs with transmit diversity.

left-most k columns of $\mathbf{H}_{CBU}$ that are filled with zeros. These column indices represent symbols that are conjugated from the data symbol vector, $\mathbf{s}$, in order to create the transmitted symbol vector, denoted as $\mathbf{s}_T$. From the two left-most columns of (19), it can be seen that only the first column is filled with zeros indicating that s_1 is conjugated and s_2 is not. Upon noting these symbols, form $\mathbf{H}_{RC}$ by summing together the left and right halves of $\mathbf{H}_{CBU}$ as

$$\begin{aligned}\mathbf{H}_{RC} &= \mathbf{H}_{CBU}(:,\ 1:k) + \mathbf{H}_{CBU}(:,\ k+1:2k)\\ &= \begin{bmatrix} h_{21}\cdot e^{j5\pi/32} & h_{11}\cdot e^{j\pi/8} \\ h_{22}\cdot e^{j5\pi/32} & h_{12}\cdot e^{j\pi/8}\end{bmatrix}.\end{aligned} \tag{20}$$

The received sample array shown in (17) may now be rewritten as

$$\begin{aligned}\mathrm{vec}(\mathbf{Z}) &= \mathbf{H}_{RC}\times\mathbf{s}_T + \mathrm{vec}(\mathbf{n})\\ &= \begin{bmatrix} h_{21}\cdot e^{j5\pi/32} & h_{11}\cdot e^{j\pi/8} \\ h_{22}\cdot e^{j5\pi/32} & h_{12}\cdot e^{j\pi/8}\end{bmatrix}\times\begin{bmatrix} s_1^* \\ s_2\end{bmatrix} + \begin{bmatrix} n_1 \\ n_2\end{bmatrix},\end{aligned} \tag{21}$$

where vec( ) represents the vec operator used to "vectorize" or vertically concatenate the columns of the argument to convert the argument into a column vector.

The minimum mean square error (MMSE)-SQRD and SIC processes given in [30] are now completed to obtain an estimate of $\mathbf{s}_T$, denoted as $\hat{\mathbf{s}}_T$. For clarity of discussion, portions of these processes from [30] are repeated here with symbols modified to stay consistent with this work.

The MMSE-SQRD process begins by performing QR decomposition of the extended $\mathbf{H}_{RC}$ matrix,

$$\underline{\mathrm{H}} = \begin{bmatrix}\mathbf{H}_{RC} \\ \sigma_n\mathbf{I}_k\end{bmatrix}, \tag{22}$$

according to [30, Tab. 1], where σ_n represents standard deviation of the AWGN, and $\mathbf{I}_k$ is a k-by-k identity matrix. QR decomposition yields

$$\underline{\mathrm{H}}(:,\ \mathbf{p}) = \underline{\mathrm{Q}}\times\underline{\mathrm{R}}, \tag{23}$$

where $\mathbf{p}$ represents a 1-by-k permutation vector that indicates the sorted column order of $\underline{\mathrm{H}}$ obtained while performing the MMSE-SQRD algorithm allowing for a more optimal detection sequence [30]. The post-sorting algorithm shown in [30, Tab. 2] may be subsequently performed to find the most optimal sequence, but from analysis provided in [30], it is generally not necessary and adds additional complexity. Thus, it is not further discussed here.

The next step is to use the determined $\underline{\mathrm{Q}}$ and $\underline{\mathrm{R}}$ matrices while applying SIC to obtain an initial estimate of $\mathbf{s}_p = \hat{\mathbf{s}}_T(\mathbf{p})$. Thus, $\mathbf{s}_p$ is a permuted version of the transmitted symbol vector estimate utilizing $\mathbf{p}$ for improved sequence of detection. To begin this step, the left-hand side of (21) is extended with a k-by-1 vector of zeros, $\mathbf{0}_k$,

$$\underline{\mathrm{Z}} = \begin{bmatrix}\mathrm{vec}(\mathbf{Z}) \\ \mathbf{0}_k\end{bmatrix}, \tag{24}$$

and multiplied by $\underline{\mathrm{Q}}^{-1} = \underline{\mathrm{Q}}^{\dagger}$ such that

$$\tilde{\mathbf{s}} = \underline{\mathrm{Q}}^{\dagger}\times\underline{\mathrm{Z}}. \tag{25}$$

SIC is then applied to obtain initial estimates of the elements of $\mathbf{s}_p$ in reverse order (i.e., $i = k, \ldots, 1$) due to the upper triangular nature of $\underline{\mathrm{R}}$. This SIC process is shown as

$$\mathbf{s}_p(i) = \underset{\forall x\in\mathbb{M}}{\mathrm{argmin}}\, d\left(\tilde{\mathbf{s}}(i), \underline{\mathrm{R}}_{ii}\cdot x + \sum_{j=i+1}^{k}\underline{\mathrm{R}}_{ij}\cdot\mathbf{s}_p(j)\right), \tag{26}$$

where $d(y, z)$ represents Euclidean distance between y and z, $\underline{\mathrm{R}}_{ij}$ represents the element in the ith row and jth column of $\underline{\mathrm{R}}$, and $\mathbb{M}$ is the set of all symbols in the M-ary modulation scheme [30], [31]. Although somewhat obvious, it should be noted in (26) that summation is not performed for $i = k$ as the summation bounds are invalid.

Upon obtaining this estimate of $\mathbf{s}_p$, it must be decided whether or not to continue with SD techniques such as in [26], [29] to obtain near ML results. For conciseness, we do not discuss these techniques further than to say that by using the initial estimate as the starting point for SD, these techniques may be used to yield a more accurate estimate of $\mathbf{s}_p$.

If BER performance loss at this stage is acceptable, and a reduced-complexity decoding process is desired, the

initial estimate of $\mathbf{s}_p$ may be used during the next decoding step. The amount of performance loss is shown in Section VII-F. Regardless of whether initial or final estimate of $\mathbf{s}_p$ is employed, an estimate of the data symbol vector, $\hat{\mathbf{s}}$, may be obtained from $\mathbf{s}_p$. This is achieved by first finding the inverse permutation, denoted as $\mathbf{p}^{-1}$, of $\mathbf{p}$ such that $\mathbf{p}(\mathbf{p}^{-1}) = [1 \cdots k]$. The inverse permutation of $\mathbf{s}_p$ is then performed to obtain an estimate of the transmitted symbol vector, $\hat{\mathbf{s}}_T = \mathbf{s}_p(\mathbf{p}^{-1})$. The symbols noted as being conjugated initially must now be conjugated again within $\hat{\mathbf{s}}_T$ to obtain $\hat{\mathbf{s}}$. Recall it was noted that s_1 is conjugated within $\mathbf{s}_T$ for the STC given in (18).

This completes the decoding process for SMSTCs. We now discuss the remaining steps involved to decode STCs with transmit diversity.

B. DECODING SPACE-TIME CODES WITH TRANSMIT DIVERSITY

If these same techniques are applied to STCs employing transmit diversity, additional steps are required to maximally combine the multiple copies of each data symbol.

The base STC for this section, referred to as non-orthogonal 4-4-2 (NO442) from this point forward, is a complex rate 2 non-orthogonal STC introduced in [32] represented as

$$\mathbf{G} = \begin{bmatrix} s_1 & s_2 & s_3 & s_4 \\ -s_2^* & s_1^* & -s_4^* & s_3^* \end{bmatrix}, \tag{27}$$

using four TX antennas to transmit four data symbols, s_1 through s_4, over two symbol time periods. Due to the transmit diversity employed by this STC, only two RX antennas are required; thus, $N_R = 2$.

The received sample array, $\mathbf{Z}$, for this system is still represented in (17) but with array dimensions according to this STC. For this discussion, the next STC is represented in (28), shown at the bottom of the page, where it can be seen that multiple operations have been applied to the base code in (27).

As in the previous section, Algorithm 8 is repeated using CSI for each RX antenna with resulting $\mathbf{H}_C$ matrices concatenated vertically to obtain a final $\mathbf{H}_{CBU}$ matrix of size rN_R-by-$2k$. As before, the channel tap vector input for the ith RX antenna is represented as $\mathbf{h}_{BU}(:, i) = [h_{1i} \ \ldots \ h_{ci}]^T$. Thus, performing these steps for the STC in (28) for the described MIMO system employing two RX antennas, the final $\mathbf{H}_{CBU}$ matrix is given in (29), shown at the bottom of the page, in transposed form.

As in the previous section, the next step is to form the restructured $\mathbf{H}_{CBU}$ matrix; however, to properly combine the multiple copies of each transmitted symbol for the STC given in (28), this restructured matrix must also be real-valued. This allows for real and imaginary components of each data symbol to be solved for individually. This process first generates

$$\mathfrak{R}_{\mathfrak{R}} = \sum_{i=0}^{1} \mathfrak{R}\Big(\mathbf{H}_{CBU}(:,\ ik+1 : ik+k)\Big), \tag{30}$$

$$\mathfrak{I}_{\mathfrak{I}} = \sum_{i=0}^{1} \mathfrak{I}\Big(\mathbf{H}_{CBU}(:,\ ik+1 : ik+k)\Big) \cdot (-1)^{i+1}, \tag{31}$$

$$\mathfrak{I}_{\mathfrak{R}} = \sum_{i=0}^{1} \mathfrak{I}\Big(\mathbf{H}_{CBU}(:,\ ik+1 : ik+k)\Big), \text{ and} \tag{32}$$

$$\mathfrak{R}_{\mathfrak{I}} = \sum_{i=0}^{1} \mathfrak{R}\Big(\mathbf{H}_{CBU}(:,\ ik+1 : ik+k)\Big) \cdot (-1)^{i}, \tag{33}$$

where $\mathfrak{R}(\,)$ represents the real component(s) of the argument and $\mathfrak{I}(\,)$ represents the imaginary component(s) of the argument. These four real-valued matrices are then combined to form the restructured, real-valued $\mathbf{H}_{RC\mathfrak{R}}$ matrix of size $2rN_R$-by-$2k$ represented as

$$\mathbf{H}_{RC\mathfrak{R}} = \begin{bmatrix} \mathfrak{R}_{\mathfrak{R}} & \mathfrak{I}_{\mathfrak{I}} \\ \mathfrak{I}_{\mathfrak{R}} & \mathfrak{R}_{\mathfrak{I}} \end{bmatrix}. \tag{34}$$

The received sample array from (17) may now be split into real-valued components and rewritten as

$$\mathbf{Z}_{\mathfrak{R}} = \mathbf{H}_{RC\mathfrak{R}} \times \mathbf{s}_{\mathfrak{R}} + \mathbf{n}_{\mathfrak{R}}, \text{ i.e.,} \tag{35}$$
$$\begin{bmatrix} \mathfrak{R}(\text{vec}(\mathbf{Z})) \\ \mathfrak{I}(\text{vec}(\mathbf{Z})) \end{bmatrix} = \begin{bmatrix} \mathfrak{R}_{\mathfrak{R}} & \mathfrak{I}_{\mathfrak{I}} \\ \mathfrak{I}_{\mathfrak{R}} & \mathfrak{R}_{\mathfrak{I}} \end{bmatrix} \times \begin{bmatrix} \mathfrak{R}(\mathbf{s}) \\ \mathfrak{I}(\mathbf{s}) \end{bmatrix} + \begin{bmatrix} \mathfrak{R}(\text{vec}(\mathbf{n})) \\ \mathfrak{I}(\text{vec}(\mathbf{n})) \end{bmatrix},$$

where $\mathbf{s}_{\mathfrak{R}}$ is the real-valued data symbol vector such that $\mathbf{s} = \mathbf{s}_{\mathfrak{R}}(1{:}k) + j\mathbf{s}_{\mathfrak{R}}(k+1 : 2k)$ [31], [33]. The MMSE-SQRD and SIC processes given in [30] are now completed to obtain an estimate of $\mathbf{s}_{\mathfrak{R}}$, denoted as $\hat{\mathbf{s}}_{\mathfrak{R}}$.

$$\mathbf{G} = \begin{bmatrix} s_4^* \cdot e^{j21\pi/16} & s_3^* \cdot e^{j15\pi/16} & s_2 \cdot e^{j49\pi/32} & s_1^* \cdot e^{j57\pi/32} \\ s_2^* \cdot e^{j3\pi/2} & s_1 \cdot e^{j39\pi/32} & s_4 \cdot e^{j23\pi/32} & s_3 \cdot e^{j17\pi/16} \end{bmatrix} \tag{28}$$

$$\mathbf{H}_{CBU} = \begin{bmatrix} 0 & h_{21} \cdot e^{j39\pi/32} & 0 & h_{22} \cdot e^{j39\pi/32} \\ h_{31} \cdot e^{j49\pi/32} & 0 & h_{32} \cdot e^{j49\pi/32} & 0 \\ 0 & h_{41} \cdot e^{j17\pi/16} & 0 & h_{42} \cdot e^{j17\pi/16} \\ 0 & h_{31} \cdot e^{j23\pi/32} & 0 & h_{32} \cdot e^{j23\pi/32} \\ h_{41} \cdot e^{j57\pi/32} & 0 & h_{42} \cdot e^{j57\pi/32} & 0 \\ 0 & h_{11} \cdot e^{j3\pi/2} & 0 & h_{12} \cdot e^{j3\pi/2} \\ h_{21} \cdot e^{j15\pi/16} & 0 & h_{22} \cdot e^{j15\pi/16} & 0 \\ h_{11} \cdot e^{j21\pi/16} & 0 & h_{12} \cdot e^{j21\pi/16} & 0 \end{bmatrix}^T \tag{29}$$

The MMSE-SQRD process begins by performing QR decomposition of the extended $\mathbf{H}_{RC\Re}$ matrix,

$$\underline{\mathrm{H}} = \begin{bmatrix} \mathbf{H}_{RC\Re} \\ \frac{\sigma_n}{\sqrt{2}}\mathbf{I}_{2k} \end{bmatrix}, \tag{36}$$

according to [30, Tab. 1], where $\mathbf{I}_{2k}$ is a $2k$-by-$2k$ identity matrix. QR decomposition yields

$$\underline{\mathrm{H}}(:,\ \mathbf{p}) = \mathrm{Q} \times \underline{\mathrm{R}}, \tag{37}$$

where $\mathbf{p}$ represents a 1-by-$2k$ permutation vector in this instance [30].

The next step is to use the determined $\underline{\mathrm{Q}}$ and $\underline{\mathrm{R}}$ matrices while applying SIC to obtain an initial estimate of $\mathbf{s}_{\Re p} = \hat{\mathbf{s}}_{\Re}(\mathbf{p})$. Thus, $\mathbf{s}_{\Re p}$ is a permuted version of the real-valued data symbol vector estimate utilizing $\mathbf{p}$ for improved sequence of detection. To begin this step, the left-hand side of (35) is extended with a $2k$-by-1 vector of zeros, $\mathbf{0}_{2k}$,

$$\underline{\mathrm{Z}}_{\Re} = \begin{bmatrix} \mathbf{Z}_{\Re} \\ \mathbf{0}_{2k} \end{bmatrix}, \tag{38}$$

and multiplied by $\mathrm{Q}^{-1} = \mathrm{Q}^{T}$ such that

$$\tilde{\mathbf{s}}_{\Re} = \mathrm{Q}^{T} \times \underline{\mathrm{Z}}_{\Re}. \tag{39}$$

SIC is then applied to obtain initial estimates of the elements of $\mathbf{s}_{\Re p}$ in reverse order (i.e., $i = 2k, \ldots, 1$) due to the upper triangular nature of $\underline{\mathrm{R}}$. This SIC process is shown as

$$\mathbf{s}_{\Re p}(i) = \underset{\forall x \in \mathbb{M}_{\Re}}{\operatorname{argmin}}\, d\left(\tilde{\mathbf{s}}_{\Re}(i), \underline{\mathrm{R}}_{ii} \cdot x + \sum_{j=i+1}^{2k} \underline{\mathrm{R}}_{ij} \cdot \mathbf{s}_{\Re p}(j)\right), \tag{40}$$

where $\mathbb{M}_{\Re}$ is the set of all unique real values for symbols in the M-ary modulation scheme [30], [31]. This assumes that the set of all unique imaginary values for these symbols is the same as $\mathbb{M}_{\Re}$, which is true for all common digital modulation schemes including those specified in [34] for the 5th Generation New Radio (5G NR). As with (26), it should be noted in (40) that summation is not performed for $i = 2k$ as the summation bounds are invalid.

As in the previous section, it must be decided whether or not to continue with SD techniques to obtain a more accurate estimate of $\mathbf{s}_{\Re p}$. Similar to before, regardless of whether initial or final estimate is employed, an estimate of the data symbol vector, $\hat{\mathbf{s}}$, may be obtained from $\mathbf{s}_{\Re p}$. This is achieved by first performing the inverse permutation of $\mathbf{s}_{\Re p}$ to obtain an estimate of the real-valued data symbol vector, $\hat{\mathbf{s}}_{\Re} = \mathbf{s}_{\Re p}(\mathbf{p}^{-1})$. The final estimate of the data symbol vector is obtained by combining the real and imaginary components as $\hat{\mathbf{s}} = \hat{\mathbf{s}}_{\Re}(1 : k) + j\hat{\mathbf{s}}_{\Re}(k+1 : 2k)$.

This completes the decoding process for non-orthogonal STCs employing transmit diversity.

C. SET BUILDING

We now look at the restrictions and cardinality obtained when building sets with these types of STCs.

For all SMSTCs, data symbols are inserted into every STC matrix element without repetition; thus, k is equal to the number of matrix elements. Due to this structure, a set built using any SMSTC as the base code may be produced using the three symbol-wise algorithms alone. For the symbol PRs algorithm, $\mathbf{v}_{asr} = [1\ \ldots\ k]$. For the symbol permutations algorithm, $\mathbf{v}_{asp} = [1\ \cdots\ k!]$, and for the symbol conjugations algorithm, the operation limit is 2^k. Thus, the cardinality of any set built using a SMSTC with k data symbols as the base code is given as

$$\left|\mathbf{S}\right| = 2^{k(a+1)} \cdot k!. \tag{41}$$

Non-orthogonal STCs must be evaluated individually to determine allowed operation limits for each set building algorithm. When the NO442 STC given in (27) is employed as the base code for set building, the symbol PRs algorithm is permitted with $\mathbf{v}_{asr} = [1]$, and the symbol permutation algorithm is permitted with $\mathbf{v}_{asp} = [1\ 3\ 6]$. Additionally, similar to the Alamouti STC as noted in [7], the operation limit for symbol conjugations is 2^{k-1} to prevent building duplicate codes. Thus, set cardinality when employing the NO442 STC is

$$\begin{aligned} \left|\mathbf{S}\right| &= 2^{a(r+c-1)+k-1} \cdot r! \cdot c! \cdot \left|\mathbf{v}_{asp}\right| \cdot 2^{a|\mathbf{v}_{asr}|} \\ &= 2^{5a+3} \cdot 2! \cdot 4! \cdot 3 \cdot 2^{a} \\ &= 1,152 \cdot 64^{a}, \end{aligned} \tag{42}$$

which coincides with that of the SPO334 STC given in (11) and evaluated in Table 1.

VI. NOMENCLATURE

In [8], definitions were provided for a cryptogram, KR, key residue class (KRC), and message and cryptogram residue classes as they pertain to ST coding-based PLS. In this section, we revisit these definitions and adapt them as necessary for incorporation of PR algorithms and alternative STCs presented in this article.

A. CRYPTOGRAM DEFINITION

The cryptogram, $\mathbf{E}$, is the transmitted STC matrix, $\mathbf{G}$, with embedded data symbol vector, $\mathbf{s}$, containing k specific data symbols of the chosen digital modulation scheme. This definition continues to hold, but by incorporating PR algorithms, cryptogram elements are no longer guaranteed to be $\in \mathbb{M}$ of the chosen M-ary modulation scheme. This is also true for the MPO334 and MPO434 STCs even without PR algorithms due to weights applied to matrix elements.

Example 3: A variant of the Alamouti STC built using PR algorithms and $a = 3$ is represented as

$$\mathbf{G} = \begin{bmatrix} s_1 \cdot e^{j3\pi/4} & -s_2^{*} \\ s_2 \cdot e^{j\pi/4} & s_1^{*} \cdot e^{j3\pi/2} \end{bmatrix}. \tag{43}$$

Employing 16QAM with $\mathbf{s} = [+1+1j,\ -3-1j]^{T}$, the cryptogram is represented as

$$\mathbf{E} = \begin{bmatrix} -\sqrt{2} & +3-1j \\ -\sqrt{2}-2\sqrt{2}j & -1-1j \end{bmatrix}. \tag{44}$$

B. KEY RESIDUE DEFINITION

The KR is the pattern or set of symbol pair relationships and their matrix locations within a given STC. In order to extend this definition to alternative STCs, including those built using PR algorithms, a broader understanding of symbol pair relationships must be incorporated. In [8], these refer to conjugate and negative-conjugate relationships that exist between symbol variations within a symbol pair, but use of PRs rather than negations expands possible relationships far beyond these two stated options. Additionally, some symbol variations within these pairs for the MPO334 and MPO434 STCs do not exhibit conjugate relationships. Furthermore, some non-orthogonal and all SM STCs contain symbols that are not paired at all. Thus, we remove the word "pair" within the revised definition.

We herein define the KR as the pattern or set of symbol relationships and their matrix locations within a given STC. For the KR, the particular symbol, e.g., s_1 versus s_2, in each relationship is irrelevant. The pattern is defined only by the relationship and location of each symbol element.

We define symbol relationships first as those which exist between the two variants in any present symbol pair. When both variants are similarly-conjugated, the relationship is determined by dividing the second variant by the first. When the variants are dissimilarly-conjugated, the relationship is determined by dividing the second variant by the conjugate of the entire first variant, including any PR that exists. Note that this is in contrast to the symbol conjugation algorithm modification discussed in Section III-C where the conjugation operation does not get applied to any PRs.

Example 4: If the first variant is $s_1 \cdot e^{j\pi/4}$, and the second variant is $s_1^* \cdot e^{j\pi/2}/2$, the variants are dissimilarly-conjugated. Thus, the relationship is determined by performing

$$\frac{s_1^* \cdot e^{j\pi/2}}{2} \Big/ \left(s_1^* \cdot e^{j7\pi/4}\right) = \frac{e^{j3\pi/4}}{2}. \tag{45}$$

The symbol relationship between these two variants is referred to as conjugate $e^{j3\pi/4}/2$.

If all cryptogram elements are guaranteed to be $\in \mathbb{M}$ of the chosen M-ary modulation scheme, symbol relationships could be limited to only those which exist between symbols; however, as this guarantee does not always hold, symbol relationships must be further defined to include the PRs, with applicable weighting factors, applied to data symbols within the reference column(s) of the STC matrix. These PRs are recorded as minimal equivalent, non-negative PRs obtained by subtracting the maximum number of $\pi/2$ multiples without obtaining a negative result; thus, all recorded PRs are $\in [0, \pi/2)$ radians.

For all STCs discussed in this work, the reference column(s) are the leftmost column(s) of the STC matrix that do not contain summations of terms but collectively contain a variation of every data symbol within the data symbol vector. For all SMSTCs, every column is a reference column in order to contain a variation of every data symbol.

TABLE 2. Tabular form KR for the Alamouti variant code shown in (43).

SGI	Relationship	Location of Variant 1	Location of Variant 2
1.1.1	Conj. $e^{j\pi/4}$	(1, 1)	(2, 2)
1.2.1	Conj. $e^{j5\pi/4}$	(2, 1)	(1, 2)
2.1.1	$e^{j\pi/4}$	(1, 1)	(1, 1)
2.2.1	$e^{j\pi/4}$	(2, 1)	(2, 1)

1) TABULAR FORM KEY RESIDUE

The KR may be represented in tabular form by grouping these relationships together. Symbol groups that produce a zero inner product are formed first. These groups consist of one or more symbol relationships for one or more data symbols. Symbol relationships for the reference column PRs are grouped together last and listed as corresponding complex exponentials with applicable weighting factors. As these relationships consist of only a single symbol variant, the symbol location is listed for both variant location 1 and 2. The tabular form KR is produced by listing the relationship and variant locations of each symbol relationship, of each data symbol, for all symbol groups within the STC.

Within the tabular form KR, a symbol group index (SGI) is used to abstractly identify the count of symbol relationships for each data symbol within each symbol group. For instance, SGI 3.2.1 refers to the first symbol relationship for the second data symbol in the third symbol group. Whereas the symbol relationship count is unnecessary for many STCs, it is important for the MPO334 and MPO434 STCs. For reference, the tabular form KR for a variant of the MPO334 STC is provided in Appendix C.

Example 5: The tabular form KR for the Alamouti variant code shown in (43) is provided in Table 2.

2) MATRIX FORM KEY RESIDUE

A KR may also be represented in matrix form. Given a STC, the matrix form KR may be determined by performing symbol permutations, symbol PRs, and symbol conjugations. Each operation is performed as necessary in order to place the unconjugated and minimally-rotated symbols in the data symbol vector, $\mathbf{s}$, in order from top-to-bottom, left-to-right within the STC reference column(s). A minimally-rotated symbol is one with a minimal equivalent, non-negative PR $\in [0, \pi/2)$ radians as previously described; thus, symbol PRs performed to achieve this result must be integer multiples of $\pi/2$ radians. Recall from Section III-C that the modified symbol conjugation operation does not impact applied PRs.

For consistency with the MPO334 and MPO434 STCs, the symbol in the highest row of the reference column that does not have a weight of $1/\sqrt{2}$ is considered s_1, the next like term is considered s_2, and the terms weighted by $1/\sqrt{2}$ are considered $s_3/\sqrt{2}$. For reference, the matrix form KR for a variant of the MPO334 STC is provided in Appendix C.

Example 6: The matrix form KR corresponding to the Alamouti variant code shown in (43) is determined by phase

TABLE 3. Number of KRCs for various STCs when incorporating PR algorithms with various values of *a*.

STC	# of KRCs ($a = 1$)	# of KRCs ($a = 2$)	# of KRCs ($a = 3$)	# of KRCs ($a = 9$)
Alamouti	2	4	32	$8,388,608$
SPO334	192	$1,536$	$98,304$	$6.76 \cdot 10^{15}$
MPO468	$2,580,480$	$3.30 \cdot 10^{8}$	$2.71 \cdot 10^{12}$	$8.18 \cdot 10^{35}$
MPO334	576	$9,216$	$1,179,648$	$5.19 \cdot 10^{18}$
MPO434	$2,304$	$73,728$	$1.89 \cdot 10^{7}$	$5.31 \cdot 10^{21}$
SM221	1	1	4	$16,384$
NO442	12	48	$3,072$	$2.11 \cdot 10^{14}$

rotating symbol s_1 and is represented as

$$\mathbf{G}_{KR} = \begin{bmatrix} s_1 \cdot e^{j\pi/4} & -s_2^* \\ s_2 \cdot e^{j\pi/4} & s_1^* \end{bmatrix}. \tag{46}$$

C. KEY RESIDUE CLASS DEFINITION

The KRC is a subset of the full code set consisting of all "keys", or STCs, sharing the same KR. All KRCs are of equal size and collectively partition the full code set for any given STC. The KRC size without PR algorithms is given in [8] as

$$4^k \cdot k!. \tag{47}$$

The KRC size for any given STC depends on the number of data symbols per codeword and the number of allowed symbol PRs that may be performed without changing the associated KR. From the definition given in the previous subsection, all $k!$ symbol permutations and 2^k symbol conjugations may be performed without affecting the KR. The number of symbol PRs permitted depends on the value of a. When $a = 1$, only 0 and π radian PRs are allowed for each symbol; however, when $a \geq 2$, all symbols may be phase-rotated by 0, $\pi/2$, π, or $3\pi/2$ radians without changing the associated KR.

Thus, incorporation of PR algorithms results in an extended KRC size of

$$2^{k(\min(a,\,2)+1)} \cdot k! \tag{48}$$

for any value of a, where $\min(a, 2)$ represents the standard minimum operation between a and 2.

For all STCs discussed in this work, it can be seen that the number of KRCs increases as the value of a increases by dividing the corresponding full set cardinality given in (11), (12), (41), or (42) as applicable by the KRC size in (48). Table 3 provides the number of KRCs for various STCs when incorporating PR algorithms with various values of a.

D. MESSAGE AND CRYPTOGRAM RESIDUE CLASS DEFINITION

A message residue class is the set of data symbol vectors that might have produced a given cryptogram. Similarly, a cryptogram residue class is the set of cryptograms that may be produced from a given data symbol vector by employing only those STCs from a single KRC. Whereas no adaptation is required for these definitions, incorporation of PR algorithms affects the size of these residue classes.

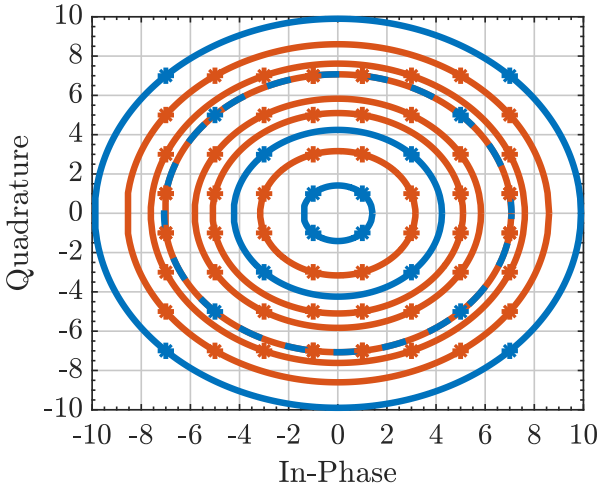

FIGURE 3. Illustration of USSs within the 64QAM symbol alphabet with $a \geq 2$.

Without PR algorithms, the size of these classes is given in [8, eq. (32)] as

$$\phi_i = \frac{4^k \cdot k!}{f}. \tag{49}$$

The repetition metric, f, is calculated as

$$f = \prod_{l=1}^{u} n_{s_l}!, \tag{50}$$

where u is the number of unique symbol sets (USSs) represented in each data symbol vector of the message residue class, and $\mathbf{n}_s = [n_{s_1} \ \cdots \ n_{s_u}]^T$ is a u-by-1 column vector containing the number of times a symbol appears in each data symbol vector from each USS represented. Without PR algorithms or when $a = 1$, a USS is a set of four symbols within the chosen modulation scheme equal to the original, negative, conjugate, and negative-conjugate of one another. It is this set of equalities that results in the 4^k term in (49) [8].

Employing PR algorithms with $a \geq 2$ introduces additional equalities requiring the USS definition to be revised. Thus, when $a \geq 2$, a USS is herein defined as a set of symbols within the chosen modulation scheme equal to the original or conjugate of one another phase-rotated by an integer multiple of $\pi/2$ radians. This affects the 4^k term in (49) and the u variable and elements of $\mathbf{n}_s$ within the f metric.

As a point of reference, the USSs for the 64QAM scheme with $a \geq 2$ are shown in Fig. 3 with colors representing the number of symbols within each set. Constant energy rings are added as an aid to the reader. The two USSs on the

red and blue dashed energy ring are separated due to this revised USS definition.

Thus, by employing PR algorithms with $a \geq 2$, the 4^k term in (49) must be replaced by the product of the number of symbols within the USS of each symbol in the data symbol vector. Performing this replacement along with substituting (50) into (49) provides

$$\phi_i = \prod_{l=1}^{u} \frac{|USS_l|^{n_{s_l}}}{n_{s_l}!} \cdot k!, \tag{51}$$

where $|USS_l|$ represents the number of symbols contained within USS_l, and u and $\mathbf{n}_s$ are as previously defined in (50).

For convenience, the terms in (51) that change for different residue classes are grouped together in a new metric, Q_f, defined as

$$Q_f = \prod_{l=1}^{u} \frac{|USS_l|^{n_{s_l}}}{n_{s_l}!} \tag{52}$$

such that

$$\phi_i = Q_f \cdot k!. \tag{53}$$

To understand how the repetition metric, f, is affected by incorporation of PR algorithms and its relation to Q_f, we recall the properties of message and cryptogram residue classes within a pure cipher system [8], [35]. Property 3 states that the number of data symbol vectors within a message residue class, i.e., ϕ_i, is equal to the number of cryptograms in the associated cryptogram residue class and is a divisor of the number of STCs in the chosen KRC. Property 4 states that each data symbol vector in a message residue class can be embedded within exactly f different STCs of the chosen KRC to produce each cryptogram in the associated cryptogram residue class [8]. From these properties, the repetition metric can be understood to be

$$f = \frac{|KRC|}{\phi_i}, \tag{54}$$

where $|KRC|$ represents the KRC size. Thus, substituting (48) for $|KRC|$ and (53) for ϕ_i yields

$$\begin{aligned} f &= \frac{2^{k(\min(a,\,2)+1)} \cdot k!}{Q_f \cdot k!} = \frac{2^{k(\min(a,\,2)+1)}}{Q_f} \\ &= 2^{k(\min(a,\,2)+1)} \cdot \prod_{l=1}^{u} \frac{n_{s_l}!}{|USS_l|^{n_{s_l}}}. \end{aligned} \tag{55}$$

When $a = 1$, the previous USS definition is valid such that each USS contains four symbols. For this case, it can be seen how (55) is equal to (50).

VII. ANALYSIS AND RESULTS

We now use established nomenclature to analyze how information-theoretic security of the alternating STC PLS scheme presented in [8] is changed by adding PR algorithms from this work. Theoretical expected BER for a passive eavesdropper is also discussed and plotted along with Monte Carlo simulations for confirmation. Finally, we present primary channel BER results of Monte Carlo simulations performed using the decoding techniques from Section V.

First, we review and revise the communications link, eavesdropper assumptions, and attack methodology from [8]. To prevent loss of generality, variables c, r, and N_R are used to allow for any STC to be employed.

A. COMMUNICATIONS LINK AND EAVESDROPPER ASSUMPTIONS

A potential MIMO communications link employing the alternating STC PLS scheme between BS and UE begins by conversion of a binary source to data symbols using a common digital modulation scheme. These symbols are then inserted into the next pseudorandomly built STC and transmitted across the wireless interface using c TX antennas over r symbol time periods. The r-by-N_R receive sample array, $\mathbf{Z}$, received by UE using N_R RX antennas is given in (17). UE is assumed to have perfect CSI and knowledge of the STC used with each codeword including all applied PRs. UE performs the applicable decoding method depending on the STC in use to equalize and decode the data symbols and demodulates the symbols to binary data using the known digital modulation scheme.

For this communications link, it is assumed that Eve:

- has a priori knowledge of the chosen original base code (OBC), value of a indicating allowed number of PRs, and digital modulation scheme employed,
- may have a priori knowledge of some plaintext data,
- employs N_{RE} RX antennas,
- has perfect CSI of the channel between BS and Eve denoted by the c-by-N_{RE} channel tap array $\mathbf{h}_{BE}$,
- has perfect timing synchronization,
- passively records all received data samples for an entire communications session,
- and has the ability to generate the matrix form KR, $\mathbf{G}_{KR}$, associated with all KRCs for the chosen OBC.

B. EAVESDROPPER ATTACK METHODOLOGY

With these non-trivial assumptions, Eve may gain information about the embedded data symbol vector and STC employed for each transmitted cryptogram from the recorded sample array, $\mathbf{Z}$, by performing the following steps [8]:

Step 1) For each KRC of the OBC, use $\mathbf{G}_{KR}$ along with channel tap array, $\mathbf{h}_{BE}$, to determine the corresponding matrix, $\mathbf{H}_{CBE}$, necessary for the applicable decoding method to be employed.

Step 2) Perform decoding method of the received sample array using the $\mathbf{H}_{CBE}$ matrix associated with each KRC to obtain results, $\hat{\mathbf{s}}_i$ for $i \in \{1, \ldots, l\}$ where l is the number of KRCs.

Step 3) Determine the KRC to which the STC used for transmission of the cryptogram belongs. This is performed by finding the index value, i, of the

result, $\hat{\mathbf{s}}_i$, with the minimum Euclidean distance to any one of the potential data symbol vectors, $\mathbf{s} \in \mathbb{M}^k$. This can be represented as

$$i = \underset{\forall i \in \{1,\ldots,l\}}{\operatorname{argmin}} \left(\min_{\forall \mathbf{s} \in \mathbb{M}^k} d\big(\mathbf{s}, \hat{\mathbf{s}}_i\big) \right). \tag{56}$$

Step 4) Obtain a coded symbol vector, $\mathbf{s}_c$, by selecting the data symbol vector, $\mathbf{s}$, with the minimum Euclidean distance to the determined result, $\hat{\mathbf{s}}_i$, from Step 3 as represented by

$$\mathbf{s}_c = \underset{\forall \mathbf{s} \in \mathbb{M}^k}{\operatorname{argmin}}\, d\big(\mathbf{s}, \hat{\mathbf{s}}_i\big). \tag{57}$$

From this point forward, the worst case is analyzed by assuming that Eve has been able to successfully obtain a coded symbol vector within the correct message residue class for each codeword of the entire recorded communications session. Likewise, it is assumed that Eve has successfully determined the correct KRC for each codeword. Although this requires a substantial amount of work, this case is possible with the assumptions provided in Section VII-A validated with sufficient signal-to-noise ratio (SNR), substantial amount of time and computing resources.

C. MESSAGE EQUIVOCATION

Once the coded symbol vector, $\mathbf{s}_c$, within the correct message residue class has been obtained, the amount of uncertainty remaining about the transmitted data symbol vector within each codeword is referred to as the equivocation of the message [35]. This is represented as

$$H(\mathbf{s}|\mathbf{s}_c) = \log_2\big(Q_f \cdot k!\big), \tag{58}$$

where $Q_f \cdot k!$ is the size of the message residue class to which the coded symbol vector belongs [8]. Given a particular coded symbol vector, Q_f may be calculated using (52) and inserted into (58) to determine message equivocation.

To calculate the mean and upper bounds (UBs) on message equivocation, the probability mass function of Q_f is required for all modulations evaluated. These functions are empirically derived by first calculating Q_f, according to (52), for all M^k possible data symbol vectors for a given M-order modulation scheme and value of k. Assuming equally probable data symbol vectors, the probability of each Q_f value is calculated by counting the number of vectors resulting in each value and dividing that count by the total number of vectors, M^k.

For all modulations evaluated, all USSs contain at least four symbols. Thus, the lower bound (LB) on message equivocation occurs when $Q_f = 4^k/k!$. This is the case when a coded symbol vector contains k symbols from a single USS with $|USS_l| = 4$. This is always true for quadrature phase-shift keying (QPSK). Mean, UB, and LB values of message equivocation for various STCs with k symbols per codeword employing various modulations can be seen in Fig. 4.

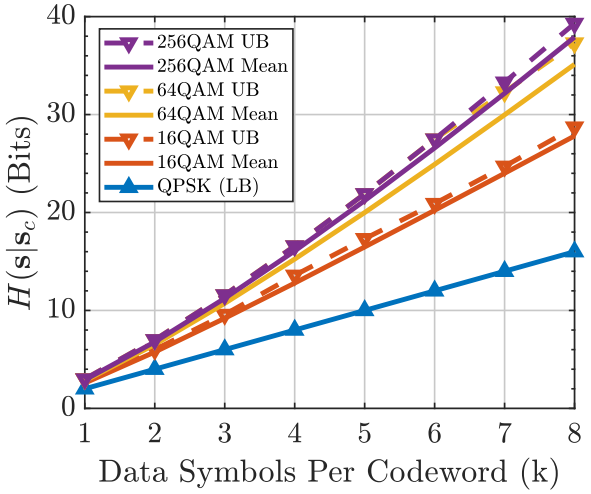

FIGURE 4. Message equivocation given STCs with various number of data symbols per codeword and modulation schemes.

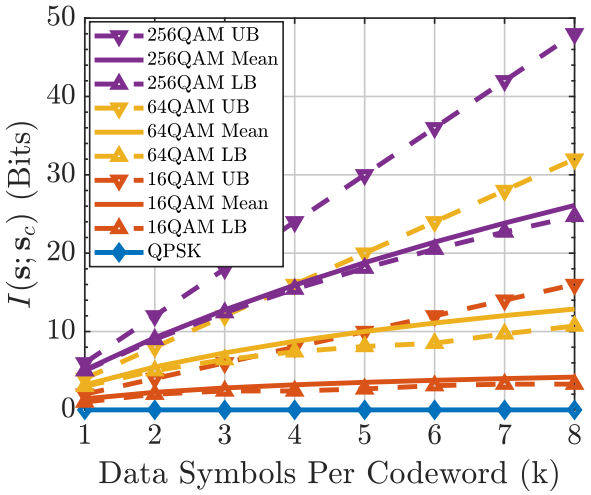

FIGURE 5. Mutual information between s and $\mathbf{s}_c$ given STCs with various number of data symbols per codeword and modulation schemes.

The amount of information Eve gains about the message by obtaining $\mathbf{s}_c$ is the mutual information between $\mathbf{s}$ and $\mathbf{s}_c$, represented as

$$\begin{aligned} I(\mathbf{s};\mathbf{s}_c) &= H(\mathbf{s}) - H(\mathbf{s}|\mathbf{s}_c) \\ &= k \cdot \log_2(M) - \log_2\big(Q_f \cdot k!\big), \end{aligned} \tag{59}$$

where $H(\mathbf{s})$ is the entropy in $\mathbf{s}$ before obtaining $\mathbf{s}_c$ which is equal to the number of bits per codeword [8], [35]. When QPSK is used, $M = 4$, $Q_f = 4^k/k!$, $I(\mathbf{s};\mathbf{s}_c) = 0$, and Eve gains no information about the message by obtaining the coded symbol vector. Mean, UB, and LB values of this mutual information for various STCs with k symbols per codeword employing various modulations can be seen in Fig. 5.

D. KEY EQUIVOCATION

Assuming the correct KRC is determined in Step 3 of Section VII-B, the KR associated with that KRC, i.e., $\mathbf{G}_{KR}$,

TABLE 4. Entropy in G, key equivocation, and mutual information between G and $\mathbf{G}_{KR}$ in bits for various STCs with $a = 9$.

STC	$H(\mathbf{G})$	$H(\mathbf{G}\|\mathbf{G}_{KR})$	$I(\mathbf{G};\mathbf{G}_{KR})$
Alamouti	30.0	7.0	23.0
SPO334	64.2	11.6	52.6
MPO468	146.8	27.5	119.3
MPO334	73.8	11.6	62.2
MPO434	83.8	11.6	72.2
SM221	21.0	7.0	14.0
NO442	64.2	16.6	47.6

is readily available. The amount of uncertainty remaining about the STC used for transmission of each cryptogram is called the equivocation of the key [35]. In this case, key equivocation depends only on the KRC size as given in (48). Thus, key equivocation is represented as

$$H(\mathbf{G}|\mathbf{G}_{KR}) = \log_2\left(2^{k(\min(a,\,2)+1)} \cdot k!\right). \tag{60}$$

The amount of information Eve gains about the key by obtaining the KR is the mutual information between $\mathbf{G}$ and $\mathbf{G}_{KR}$, represented as

$$I(\mathbf{G};\mathbf{G}_{KR}) = H(\mathbf{G}) - H(\mathbf{G}|\mathbf{G}_{KR}), \tag{61}$$

where

$$H(\mathbf{G}) = \log_2(|\mathbf{S}|) \tag{62}$$

is the entropy in $\mathbf{G}$ before obtaining the KR in bits. $H(\mathbf{G})$ depends on the set cardinality which depends on the STC chosen and the value of a. For various STCs discussed in this work, the values of $H(\mathbf{G})$, $H(\mathbf{G}|\mathbf{G}_{KR})$, and $I(\mathbf{G};\mathbf{G}_{KR})$ are provided in Table 4 for $a = 9$.

E. EXPECTED EAVESDROPPER BIT ERROR RATE

Although message equivocation gives one perspective of uncertainty, it does not offer a complete understanding of expected BER for an eavesdropper when decoding the coded symbol vector, $\mathbf{s}_c$. From [7], *symbol assignment* and *variation decision* operations may be performed to decode $\mathbf{s}_c$. In [8, eq. (65)], the expected eavesdropper BER when decoding $\mathbf{s}_c$ by this method is given as

$$\text{BER}_{\text{exp}} = \frac{E[l|k] \cdot E[B_{ca}] + (k - E[l|k]) \cdot E[B_{ia}]}{k \cdot b}, \tag{63}$$

where $E[\,]$ is the expectation operator, l is the number of correctly assigned symbols per codeword during *symbol assignment*, k is the number of symbols per codeword, B_{ca} is the number of bit errors per correctly assigned symbol, B_{ia} is the number of bit errors per incorrectly assigned symbol, $b = \log_2(M)$ is the number of bits per symbol, and M is the order of the digital modulation scheme employed.

Assuming codeword symbols are independent of one another, $E[B_{ia}] = \frac{b}{2}$. Proof is given in [8] that $E[l|k] = 1$ regardless of the value of k. Without PR algorithms or when $a = 1$, $E[B_{ca}] = 1$ based on the average of the four possible variation decisions [7].

TABLE 5. Expected number of bit errors per correctly assigned symbol for various modulations.

Modulation	$E[B_{ca}]$
QPSK	1
16QAM	1.5
64QAM	2
256QAM	2.5

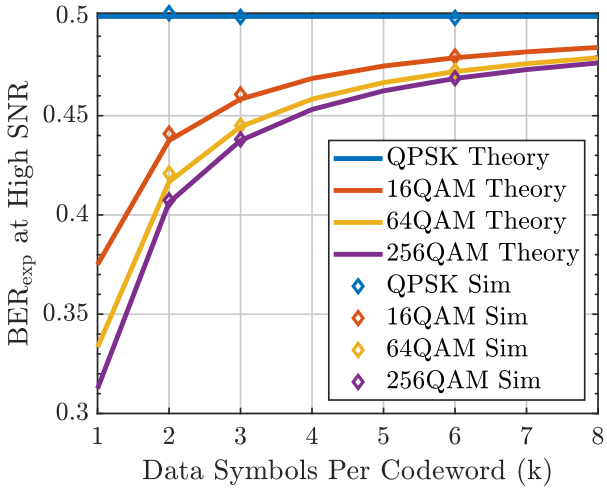

FIGURE 6. Theoretical plot of (64) with various number of data symbols per codeword and modulation schemes including high SNR Monte Carlo simulations for the Alamouti, SPO334, and MPO468 STCs.

When PR algorithms are used with $a \geq 2$, the number of possible variation decisions for a given symbol from $\mathbf{s}_c$ becomes dependent on the number of symbols within the particular USS to which the original symbol belongs. Assuming modulation symbols are equally probable, $E[B_{ca}]$ is empirically determined by calculating the expected number of bit errors when demodulating each symbol within each USS by every other symbol within the same USS and averaging results across all symbols within the chosen modulation scheme. These empirical results for various 5G NR modulations employing bit mappings specified in [34] are provided in Table 5.

Inserting stated values for $E[l|k]$ and $E[B_{ia}]$ into (63) yields

$$\text{BER}_{\text{exp}} = \frac{1 \cdot E[B_{ca}] + (k-1) \cdot \frac{b}{2}}{k \cdot b}. \tag{64}$$

This represents the average expected eavesdropper BER for all message residue classes once the eavesdropper has obtained a coded symbol vector within the correct message residue class. A plot of (64) for various STCs with k symbols per codeword employing various modulation schemes with corresponding values of $E[B_{ca}]$ from Table 5 can be seen in Fig. 6.

Additionally, 100,000 Monte Carlo simulations are performed for the Alamouti, SPO334, and MPO468 STCs employing QPSK, 16QAM, 64QAM, and 256QAM with energy per bit to noise power spectral density ratio ($\text{E}_\text{b}/\text{N}_0$)

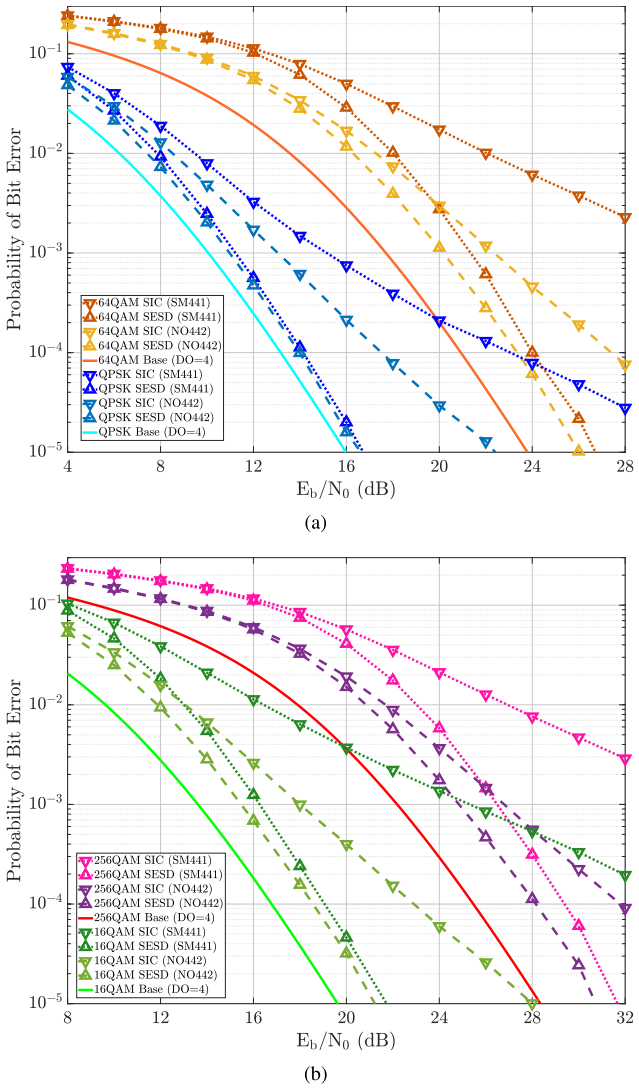

FIGURE 7. Comparison of decoding techniques for SM441 and NO442 STCs using various modulations and DO of four. (a) QPSK and 64QAM; (b) 16QAM and 256QAM.

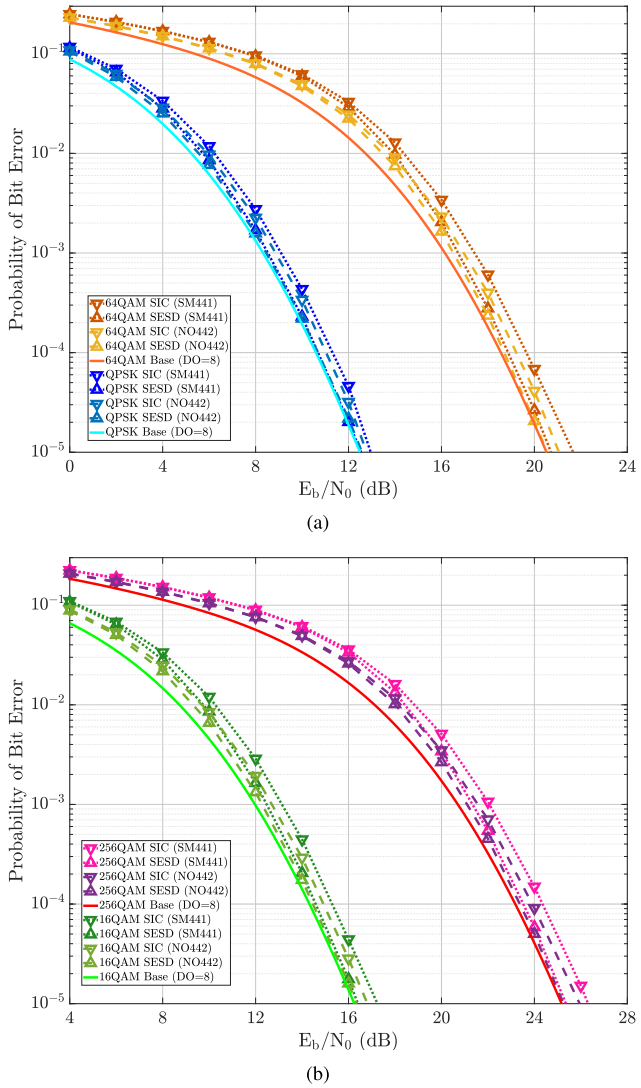

FIGURE 8. Comparison of decoding techniques for SM441 and NO442 STCs using various modulations and DO of eight. (a) QPSK and 64QAM; (b) 16QAM and 256QAM.

values of 5, 10, 15, and 20 dB, respectively. For each simulation, the worst case assumption that Eve has obtained a coded symbol vector within the correct message residue class is enforced. The results of these simulations are included in Fig. 6.

F. PRIMARY CHANNEL BIT ERROR RATE RESULTS

Monte Carlo simulations are performed over $1 \cdot 10^6$ iterations of two MIMO systems in a Rayleigh fading channel with AWGN. The first system employs a SMSTC referred to as SM441 using four TX antennas to transmit four data symbols, s_1 through s_4, over one symbol time period. This system applies the alternating STC PLS scheme with $a = 4$, four RX antennas, and various modulations. The second system employs the NO442 STC as shown in (27) using four TX antennas. This system applies the alternating STC PLS scheme with $a = 4$, two RX antennas, and various modulations.

Decoding techniques presented in Sections V-A and V-B are compared. A baseline performance curve is plotted for an orthogonal STC with diversity order (DO) of four corresponding to theoretical expected performance of the Alamouti STC received using two RX antennas [36]. This curve represents the baseline as both MIMO systems under test exhibit a combined DO of four; thus, expected ML performance for both systems should exhibit the same slope as that of the baseline curve but with horizontal separation due to non-orthogonality of the employed STCs [31]. The decoding techniques tested include the combined MMSE-SQRD and SIC algorithms, denoted as SIC for short, along with the MATLAB communications toolbox Schnorr-Euchner (SE) sphere decoder (SESD) [26], [29], [37]. Results for various modulations are shown in Fig. 7.

It can be seen from Fig. 7 that BER performance degradation is more significant between the two compared decoding techniques when employing SM441 rather than NO442. This indicates that the significant BER performance boost is likely worth the additional decoding complexity of performing follow-on SD techniques when using a SMSTC; however,

TABLE 6. Increases in security of the alternating STC PLS scheme by incorporation of PR algorithms with $a = 9$.

STC	Modulation	Message Equivocation (Bits)	Key Equivocation (Bits)	Expected Eavesdropper BER	Attack Complexity (Bits)
Alamouti	QPSK	0.00	2	0.000	24
	16QAM	1.08		0.063	
	64QAM	1.61		0.083	
SPO334	QPSK	0.00	3	0.000	48
	16QAM	1.52		0.042	
	64QAM	2.39		0.056	
MPO468	QPSK	0.00	6	0.000	104
	16QAM	2.62		0.021	
	64QAM	4.65		0.028	

this may not be true for non-orthogonal STCs with transmit diversity depending on the application's required error rate.

The same Monte Carlo simulations are performed again with twice the number of RX antennas for each system to better understand decoding performance differences with additional antenna resources. Thus, for the first system using the SM441 STC, eight RX antennas are now employed. For the second system using the NO442 STC, four RX antennas are now employed. Baseline performance curves are plotted once again but with DO of eight. Results of these simulations with additional RX antennas are shown in Fig. 8.

It can be seen by comparing Figs. 7 and 8 that there is much less degradation in BER performance between the two compared decoding techniques when employing additional RX antennas with either MIMO system under test. This indicates that SD techniques become less beneficial as additional RX antennas are employed.

VIII. SECURITY COMPARISON

We now present a comparison of security offered by the alternating STC PLS scheme with negation algorithms versus incorporation of PR algorithms with $a = 9$.

The least complex attack is for Eve to perform a brute force search for the temporary base code (tBC), temporary order of operations vector ($\mathbf{v}_{too}$), and pseudorandom number generator (PRNG) state at the beginning of the data transmission phase [8]. This attack is referred to as the exhaustive key search [38]. A 64-bit state space is assumed for the PRNG, and the number of possible values for $\mathbf{v}_{too}$ is 5! or 7! depending on the allowed operations for the chosen STC. The number of possible values for tBC depends on the set cardinality of the chosen STC. Thus, the largest increase in security is provided by the significantly larger set cardinalities achieved by incorporating PR algorithms.

Increases in security in terms of message and key equivocation and expected eavesdropper BER can also be seen by comparing Figs. 4 and 6 and Table 4 with similar figures and tables in [8]. Table 6 provides a summary of increased security obtained by incorporation of PR algorithms. Message and key equivocation increases are given in bits of uncertainty per codeword. Increases in expected eavesdropper BER assume that Eve applies the attack methodology discussed in Section VII-B and coded symbol vector decoding steps discussed in Section VII-E. The exhaustive key search attack complexity increase in bits is calculated as the difference between the base two logarithm of set cardinalities for $a = 9$ and $a = 1$ as given in Table 1 for the various STCs.

IX. CONCLUSION

Several extensions were proposed to the alternating STC PLS scheme, initially presented in [8], including alternative orthogonal, non-orthogonal, and SM STCs as well as row, column, and symbol PR algorithms. A revised MRC matrix generation algorithm was given for use with these extensions. Methods were discussed to decode alternating non-orthogonal and SM STCs. Monte Carlo simulations were provided to compare BER performance between two different decoding methods. It was shown that SD techniques are beneficial in certain circumstances but are not vastly superior to SQRD and SIC methods in every case. Secrecy system nomenclature applied to ST coding-based PLS schemes was adapted for use with alternative STCs and algorithms presented herein. Information-theoretic security analysis was provided along with expected eavesdropper BER based on an assumed attack methodology. A comparison of security offered by the alternating STC PLS scheme with and without use of PR algorithms was performed. Substantially greater exhaustive key search attack complexity is achieved by using PR algorithms.

APPENDIX A
UPDATED SET BUILDING ALGORITHMS

The row, column, and symbol negation algorithms given in [8] are revised here to produce corresponding PR Algorithms 1, 2, and 7. Values of c, k, r, and d for the chosen STC are known to all algorithms in this article. If a two dimensional STC is employed, then $d = 1$. All text within curly brackets are comments to aid the reader. The $de2bi(i, n, p)$ MATLAB function is used to create a length n, little-endian (*aka* right most significant digit) row vector of the decimal value i represented in base p [37]. The $diag(\mathbf{v})$ MATLAB function is used to create a square

Algorithm 1 Row Phase Rotations

Inputs: **S** {Set of STCs},
PC_i {# of populated codes in input **S**},
$\mathbf{A}_{syms}$ {Symbols record array}

Outputs: **S**, PC_o {# of populated codes in output **S**}, $\mathbf{A}_{syms}$

$PC_o = PC_i \cdot 2^{a \cdot r}$ {Operation limit = $2^{a \cdot r}$}
for $it = 1$ to $2^{a \cdot r} - 1$ **do**
  $\mathbf{v}_{row} = de2bi(it, r, 2^a)$ {Row PR vector}
  In $\mathbf{v}_{row}$, replace each element, x, with $e^{j2\pi x/2^a}$
  $\mathbf{T}_{row} = diag(\mathbf{v}_{row})$
  for $m = 1$ to PC_i **do**
    $\mathbf{A}_{syms}(PC_i \cdot it + m, :) = \mathbf{A}_{syms}(m, :)$
    {Update symbols record array}
    for $di = 1$ to d **do**
      $\mathbf{S}(:, :, di, PC_i \cdot it + m) = \mathbf{T}_{row} \times \mathbf{S}(:, :, di, m)$
    end for
  end for
end for
return **S**, PC_o, $\mathbf{A}_{syms}$

Algorithm 2 Column Phase Rotations

Inputs: **S**, PC_i, $\mathbf{A}_{syms}$
Outputs: **S**, PC_o, $\mathbf{A}_{syms}$

$PC_o = PC_i \cdot 2^{a \cdot (c-1)}$ {Operation limit = $2^{a \cdot (c-1)}$}
for $it = 1$ to $2^{a \cdot (c-1)} - 1$ **do**
  $\mathbf{v}_{col} = de2bi(it, c, 2^a)$ {Column PR vector}
  In $\mathbf{v}_{col}$, replace each element, x, with $e^{j2\pi x/2^a}$
  $\mathbf{T}_{col} = diag(\mathbf{v}_{col})$
  for $m = 1$ to PC_i **do**
    $\mathbf{A}_{syms}(PC_i \cdot it + m, :) = \mathbf{A}_{syms}(m, :)$
    {Update symbols record array}
    for $di = 1$ to d **do**
      $\mathbf{S}(:, :, di, PC_i \cdot it + m) = \mathbf{S}(:, :, di, m) \times \mathbf{T}_{col}$
    end for
  end for
end for
return **S**, PC_o, $\mathbf{A}_{syms}$

diagonal matrix with elements of vector **v** on the main diagonal [37]. Lastly, $\times$ is used to indicate matrix multiplication whereas $\circ$ indicates element-wise multiplication, and $==$ indicates logically equal whereas $=$ indicates assignment.

The column permutations and symbol conjugations algorithms given in [7] and revised in [8] are revised once again and included here as Algorithms 4 and 5, respectively, to work with alternative STCs and PR algorithms presented in this article.

APPENDIX B
PROOF OF MRC SEQUENCE FOR MPO334

In this appendix, we prove the MPO334 STC along with updated Algorithm 8 given in Section IV to be compatible with the two-step MRC sequence given in [8]. Starting with the MPO334 STC represented as

$$\mathbf{G} = \begin{bmatrix} s_1 & s_2 & \frac{s_3}{\sqrt{2}} \\ -s_2^* & s_1^* & \frac{s_3}{\sqrt{2}} \\ \frac{s_3^*}{\sqrt{2}} & \frac{s_3^*}{\sqrt{2}} & \frac{(-s_1 - s_1^* + s_2 - s_2^*)}{2} \\ \frac{s_3^*}{\sqrt{2}} & \frac{-s_3^*}{\sqrt{2}} & \frac{(s_2 + s_2^* + s_1 - s_1^*)}{2} \end{bmatrix}, \tag{65}$$

$$\begin{aligned} \mathbf{H}_{int} &= \mathbf{H}_{CBU}^{\dagger} \times \mathbf{H}_{CBU} \\ &= \begin{bmatrix} h_1^* & 0 & \frac{-h_3^*}{2} & \frac{h_3^*}{2} \\ h_2^* & 0 & \frac{h_3^*}{2} & \frac{h_3^*}{2} \\ \frac{h_3^*}{\sqrt{2}} & \frac{h_3^*}{\sqrt{2}} & 0 & 0 \\ 0 & h_2^* & \frac{-h_3^*}{2} & \frac{-h_3^*}{2} \\ 0 & -h_1^* & \frac{-h_3^*}{2} & \frac{h_3^*}{2} \\ 0 & 0 & \frac{(h_1^* + h_2^*)}{\sqrt{2}} & \frac{(h_1^* - h_2^*)}{\sqrt{2}} \end{bmatrix} \times \begin{bmatrix} h_1 & h_2 & \frac{h_3}{\sqrt{2}} & 0 & 0 & 0 \\ 0 & 0 & \frac{h_3}{\sqrt{2}} & h_2 & -h_1 & 0 \\ \frac{-h_3}{2} & \frac{h_3}{2} & 0 & \frac{-h_3}{2} & \frac{-h_3}{2} & \frac{(h_1 + h_2)}{\sqrt{2}} \\ \frac{h_3}{2} & \frac{h_3}{2} & 0 & \frac{-h_3}{2} & \frac{h_3}{2} & \frac{(h_1 - h_2)}{\sqrt{2}} \end{bmatrix} \\ &= \begin{bmatrix} h_1 h_1^* + \frac{h_3 h_3^*}{2} & h_1^* h_2 & \frac{h_1^* h_3}{\sqrt{2}} & 0 & \frac{h_3 h_3^*}{2} & -\frac{h_2 h_3^*}{\sqrt{2}} \\ h_1 h_2^* & h_2 h_2^* + \frac{h_3 h_3^*}{2} & \frac{h_2^* h_3}{\sqrt{2}} & -\frac{h_3 h_3^*}{2} & 0 & \frac{h_1 h_3^*}{\sqrt{2}} \\ \frac{h_1 h_3^*}{\sqrt{2}} & \frac{h_2 h_3^*}{\sqrt{2}} & h_3 h_3^* & \frac{h_2 h_3^*}{\sqrt{2}} & -\frac{h_1 h_3^*}{\sqrt{2}} & 0 \\ 0 & -\frac{h_3 h_3^*}{2} & \frac{h_2^* h_3}{\sqrt{2}} & h_2 h_2^* + \frac{h_3 h_3^*}{2} & -h_1 h_2^* & -\frac{h_1 h_3^*}{\sqrt{2}} \\ \frac{h_3 h_3^*}{2} & 0 & -\frac{h_1^* h_3}{\sqrt{2}} & -h_1^* h_2 & h_1 h_1^* + \frac{h_3 h_3^*}{2} & -\frac{h_2 h_3^*}{\sqrt{2}} \\ -\frac{h_2^* h_3}{\sqrt{2}} & \frac{h_1^* h_3}{\sqrt{2}} & 0 & -\frac{h_1^* h_3}{\sqrt{2}} & -\frac{h_2^* h_3}{\sqrt{2}} & h_1 h_1^* + h_2 h_2^* \end{bmatrix} \end{aligned} \tag{67}$$

Algorithm 4 Column Permutations

Inputs: $\mathbf{S}$, PC_i, $\mathbf{A}_{syms}$, $\mathbf{v}_{acp}$ {Allowed column permutations vector}
Outputs: $\mathbf{S}$, PC_o, $\mathbf{A}_{syms}$

$PC_o = PC_i \cdot |\mathbf{v}_{acp}|$ {Operation limit = $|\mathbf{v}_{acp}|$}
$\mathbf{C}_{perms} = perms(c)$ {Column permutations array}
for $it = 1$ to $|\mathbf{v}_{acp}| - 1$ **do**
  for $m = 1$ to PC_i **do**
    $\mathbf{A}_{syms}(PC_i \cdot it + m, :) = \mathbf{A}_{syms}(m, :)$
    $\mathbf{S}(:, :, :, PC_i \cdot it + m) = \mathbf{S}(:, \mathbf{C}_{perms}(\mathbf{v}_{acp}(it + 1), :), :, m)$
  end for
end for
return $\mathbf{S}$, PC_o, $\mathbf{A}_{syms}$

and channel tap vector, $\mathbf{h}_{BU} = [h_1 \;\; h_2 \;\; h_3]^T$, Algorithm 8 is performed to construct

$$\mathbf{H}_{CBU} = \begin{bmatrix} h_1 & h_2 & \frac{h_3}{\sqrt{2}} & 0 & 0 & 0 \\ 0 & 0 & \frac{h_3}{\sqrt{2}} & h_2 & -h_1 & 0 \\ \frac{-h_3}{2} & \frac{h_3}{2} & 0 & \frac{-h_3}{2} & \frac{-h_3}{2} & \frac{(h_1+h_2)}{\sqrt{2}} \\ \frac{h_3}{2} & \frac{h_3}{2} & 0 & \frac{-h_3}{2} & \frac{h_3}{2} & \frac{(h_1-h_2)}{\sqrt{2}} \end{bmatrix}. \quad (66)$$

To confirm the sufficient statistic, [8, eq. (24)] is calculated to obtain (67), shown at the bottom of the previous page.

Equation [8, eq. (23)] is then used iteratively for $x, y \in \{1, \ldots, k\}$ to confirm [8, eq. (22)].

With $k = 3$, $x = 1$, and $y = 1$,

$$\begin{aligned} \hat{\mathbf{H}}(1,1) &= \mathbf{H}_{int}(1,1) + \mathbf{H}^*_{int}(4,4) + \mathbf{H}_{int}(1,4) + \mathbf{H}^*_{int}(4,1) \\ &= h_1 h_1^* + \frac{h_3 h_3^*}{2} + h_2 h_2^* + \frac{h_3 h_3^*}{2} + 0 + 0 \\ &= \|\mathbf{h}_{BU}\|^2. \end{aligned} \quad (68)$$

With $k = 3$, $x = 2$, and $y = 1$,

$$\begin{aligned} \hat{\mathbf{H}}(2,1) &= \mathbf{H}_{int}(2,1) + \mathbf{H}^*_{int}(5,4) + \mathbf{H}_{int}(2,4) + \mathbf{H}^*_{int}(5,1) \\ &= h_1 h_2^* - h_1 h_2^* - \frac{h_3 h_3^*}{2} + \frac{h_3 h_3^*}{2} = 0. \end{aligned} \quad (69)$$

With $k = 3$, $x = 3$, and $y = 1$,

$$\begin{aligned} \hat{\mathbf{H}}(3,1) &= \mathbf{H}_{int}(3,1) + \mathbf{H}^*_{int}(6,4) + \mathbf{H}_{int}(3,4) + \mathbf{H}^*_{int}(6,1) \\ &= \frac{h_1 h_3^*}{\sqrt{2}} - \frac{h_1 h_3^*}{\sqrt{2}} + \frac{h_2 h_3^*}{\sqrt{2}} - \frac{h_2 h_3^*}{\sqrt{2}} = 0. \end{aligned} \quad (70)$$

With $k = 3$, $x = 1$, and $y = 2$,

$$\begin{aligned} \hat{\mathbf{H}}(1,2) &= \mathbf{H}_{int}(1,2) + \mathbf{H}^*_{int}(4,5) \\ &\quad + \mathbf{H}_{int}(1,5) + \mathbf{H}^*_{int}(4,2) \\ &= h_1^* h_2 - h_1^* h_2 + \frac{h_3 h_3^*}{2} - \frac{h_3 h_3^*}{2} = 0. \end{aligned} \quad (71)$$

With $k = 3$, $x = 2$, and $y = 2$,

$$\begin{aligned} \hat{\mathbf{H}}(2,2) &= \mathbf{H}_{int}(2,2) + \mathbf{H}^*_{int}(5,5) + \mathbf{H}_{int}(2,5) + \mathbf{H}^*_{int}(5,2) \\ &= h_2 h_2^* + \frac{h_3 h_3^*}{2} + h_1 h_1^* + \frac{h_3 h_3^*}{2} + 0 + 0 \\ &= \|\mathbf{h}_{BU}\|^2. \end{aligned} \quad (72)$$

With $k = 3$, $x = 3$, and $y = 2$,

$$\begin{aligned} \hat{\mathbf{H}}(3,2) &= \mathbf{H}_{int}(3,2) + \mathbf{H}^*_{int}(6,5) + \mathbf{H}_{int}(3,5) + \mathbf{H}^*_{int}(6,2) \\ &= \frac{h_2 h_3^*}{\sqrt{2}} - \frac{h_2 h_3^*}{\sqrt{2}} - \frac{h_1 h_3^*}{\sqrt{2}} + \frac{h_1 h_3^*}{\sqrt{2}} = 0. \end{aligned} \quad (73)$$

With $k = 3$, $x = 1$, and $y = 3$,

$$\begin{aligned} \hat{\mathbf{H}}(1,3) &= \mathbf{H}_{int}(1,3) + \mathbf{H}^*_{int}(4,6) + \mathbf{H}_{int}(1,6) + \mathbf{H}^*_{int}(4,3) \\ &= \frac{h_1^* h_3}{\sqrt{2}} - \frac{h_1^* h_3}{\sqrt{2}} - \frac{h_2 h_3^*}{\sqrt{2}} + \frac{h_2 h_3^*}{\sqrt{2}} = 0. \end{aligned} \quad (74)$$

Algorithm 5 Symbol Conjugations

Inputs: $\mathbf{S}$, PC_i, $\mathbf{A}_{syms}$, $\mathbf{s}_{ext}$ {Extended data symbol vector}
Outputs: $\mathbf{S}$, PC_o, $\mathbf{A}_{syms}$

$PC_o = PC_i \cdot 2^k$ {Operation limit = 2^k}
for $it = 1$ to $2^k - 1$ **do**
  $\mathbf{v}_{sym} = de2bi(it, k, 2)$ {Symbol conjugation vector}
  for $m = 1$ to PC_i **do**
    $\mathbf{S}(:, :, :, PC_i \cdot it + m) = \mathbf{S}(:, :, :, m)$ {Initialize STC}
    $\mathbf{A}_{syms}(PC_i \cdot it + m, :) = \mathbf{A}_{syms}(m, :)$
    for $di = 1$ to d **do**
      for $ri = 1$ to r **do**
        for $ci = 1$ to c **do**
          if $\mathbf{S}(ri, ci, di, m) \neq 0$ **then**
            $ki =$ symbol index of $\mathbf{S}(ri, ci, di, m)$ {e.g., $ki = 1$ for $\mathbf{S}(ri, ci, di, m) == s_1$}
            if $\mathbf{v}_{sym}(ki) == 1$ **then**
              $w =$ weight applied to $\mathbf{S}(ri, ci, di, m)$ {e.g., $w = \frac{1}{\sqrt{2}}$ for $\mathbf{S}(ri, ci, di, m) == \frac{-s_1^*}{\sqrt{2}}$}
              $\phi =$ PR angle applied to $\mathbf{S}(ri, ci, di, m)$ {e.g., $\phi = \pi$ for $\mathbf{S}(ri, ci, di, m) == \frac{-s_1^*}{\sqrt{2}}$}
              if $\mathbf{S}(ri, ci, di, m)$ is conjugated **then**
                $g = 0$
              else
                $g = 1$
              end if
              $\mathbf{S}(ri, ci, di, PC_i \cdot it + m) = w \cdot e^{j\phi} \cdot \mathbf{s}_{ext}(ki + g \cdot k)$
            end if
          end if
        end for
      end for
    end for
  end for
end for
return $\mathbf{S}$, PC_o, $\mathbf{A}_{syms}$

Algorithm 7 Symbol Phase Rotations

Inputs: $\mathbf{S}$, PC_i, $\mathbf{A}_{syms}$, $\mathbf{v}_{asr}$ {Allowed symbol PRs vector}

Outputs: $\mathbf{S}$, PC_o, $\mathbf{A}_{syms}$

$PC_o = PC_i \cdot 2^{a\cdot|\mathbf{v}_{asr}|}$ {Operation limit = $2^{a\cdot|\mathbf{v}_{asr}|}$}
for $it = 1$ to $2^{a\cdot|\mathbf{v}_{asr}|} - 1$ **do**
  $\mathbf{v}_{sym} = de2bi(it, |\mathbf{v}_{asr}|, 2^a)$ {Symbol PR vector}
  for $m = 1$ to PC_i **do**
    $\mathbf{A}_{syms}(PC_i \cdot it + m, :) = \mathbf{A}_{syms}(m, :)$ {Update symbols record array}
    $\mathbf{v}_{psr} = \mathbf{A}_{syms}(m, \mathbf{v}_{asr})$ {Permuted allowed symbol PRs vector}
    $\mathbf{v}_{csr} =$ vector of 0's of size 1-by-k {Current iteration full-length symbol PR vector}
    $\mathbf{v}_{csr}(\mathbf{v}_{psr}) = \mathbf{v}_{sym}$ {Insert $\mathbf{v}_{sym}$ into $\mathbf{v}_{csr}$ in permuted locations}
    for $di = 1$ to d **do**
      $\mathbf{T} =$ 2-dimensional array of 1's of size r-by-c
      for $ri = 1$ to r **do**
        for $ci = 1$ to c **do**
          if $\mathbf{S}(ri, ci, di, m) \neq 0$ **then**
            $ki =$ symbol index of $\mathbf{S}(ri, ci, di, m)$ {e.g., $ki = 3$ for $\mathbf{S}(ri, ci, di, m) == \frac{s_3}{\sqrt{2}}$}
            if $\mathbf{v}_{csr}(ki) \neq 0$ **then**
              if $\mathbf{S}(ri, ci, di, m)$ is conjugated **then**
                $g = -1$ {PR angle must be negated}
              else
                $g = 1$
              end if
              $x = \mathbf{v}_{csr}(ki)$ {Symbol PR angle index}
              $\mathbf{T}(ri, ci) = e^{j2\pi gx/2^a}$
            end if
          end if
        end for
      end for
      $\mathbf{S}(:, :, di, PC_i \cdot it + m) = \mathbf{S}(:, :, di, m) \circ \mathbf{T}$
    end for
  end for
end for
return $\mathbf{S}$, PC_o, $\mathbf{A}_{syms}$

With $k = 3$, $x = 2$, and $y = 3$,

$$\begin{aligned}\hat{\mathbf{H}}(2,3) &= \mathbf{H}_{int}(2,3) + \mathbf{H}^*_{int}(5,6) + \mathbf{H}_{int}(2,6) + \mathbf{H}^*_{int}(5,3)\\ &= \frac{h_2^* h_3}{\sqrt{2}} - \frac{h_2^* h_3}{\sqrt{2}} + \frac{h_1 h_3^*}{\sqrt{2}} - \frac{h_1 h_3^*}{\sqrt{2}} = 0. \end{aligned} \tag{75}$$

With $k = 3$, $x = 3$, and $y = 3$,

$$\begin{aligned}\hat{\mathbf{H}}(3,3) &= \mathbf{H}_{int}(3,3) + \mathbf{H}^*_{int}(6,6) + \mathbf{H}_{int}(3,6) + \mathbf{H}^*_{int}(6,3)\\ &= h_3 h_3^* + h_1 h_1^* + h_2 h_2^* + 0 + 0 = \|\mathbf{h}_{BU}\|^2. \end{aligned} \tag{76}$$

$$\begin{aligned}\hat{\mathbf{s}}_{int} &= \mathbf{H}^{\dagger}_{CBU} \times \mathbf{Z}\\ &= \begin{bmatrix} h_1^* & 0 & \frac{-h_3^*}{2} & \frac{h_3^*}{2} \\ h_2^* & 0 & \frac{h_3^*}{2} & \frac{h_3^*}{2} \\ \frac{h_3^*}{\sqrt{2}} & \frac{h_3^*}{\sqrt{2}} & 0 & 0 \\ 0 & h_2^* & \frac{-h_3^*}{2} & \frac{-h_3^*}{2} \\ 0 & -h_1^* & \frac{-h_3^*}{2} & \frac{h_3^*}{2} \\ 0 & 0 & \frac{(h_1^*+h_2^*)}{\sqrt{2}} & \frac{(h_1^*-h_2^*)}{\sqrt{2}} \end{bmatrix} \times \begin{bmatrix} s_1 h_1 + s_2 h_2 + \frac{s_3 h_3}{\sqrt{2}} + n_1 \\ \frac{s_3 h_3}{\sqrt{2}} + s_1^* h_2 - s_2^* h_1 + n_2 \\ -\frac{s_1 h_3}{2} + \frac{s_2 h_3}{2} - \frac{s_1^* h_3}{2} - \frac{s_2^* h_3}{2} + \frac{s_3^*(h_1+h_2)}{\sqrt{2}} + n_3 \\ \frac{s_1 h_3}{2} + \frac{s_2 h_3}{2} - \frac{s_1^* h_3}{2} + \frac{s_2^* h_3}{2} + \frac{s_3^*(h_1-h_2)}{\sqrt{2}} + n_4 \end{bmatrix}\\ &= \begin{bmatrix} s_1 h_1 h_1^* + s_2 h_1^* h_2 + \frac{s_3 h_1^* h_3}{\sqrt{2}} + n_1 h_1^* + \frac{s_1 h_3 h_3^*}{2} + \frac{s_2^* h_3 h_3^*}{2} - \frac{s_3^* h_2 h_3^*}{\sqrt{2}} - \frac{n_3 h_3^*}{2} + \frac{n_4 h_3^*}{2} \\ s_1 h_1 h_2^* + s_2 h_2 h_2^* + \frac{s_3 h_2^* h_3}{\sqrt{2}} + n_1 h_2^* + \frac{s_2 h_3 h_3^*}{2} - \frac{s_1^* h_3 h_3^*}{2} + \frac{s_3^* h_1 h_3^*}{\sqrt{2}} + \frac{n_3 h_3^*}{2} + \frac{n_4 h_3^*}{2} \\ \frac{s_1 h_1 h_3^*}{\sqrt{2}} + \frac{s_2 h_2 h_3^*}{\sqrt{2}} + s_3 h_3 h_3^* + \frac{n_1 h_3^*}{\sqrt{2}} + \frac{s_1^* h_2 h_3^*}{\sqrt{2}} - \frac{s_2^* h_1 h_3^*}{\sqrt{2}} + \frac{n_2 h_3^*}{\sqrt{2}} \\ \frac{s_3 h_2^* h_3}{\sqrt{2}} + s_1^* h_2 h_2^* - s_2^* h_1 h_2^* + n_2 h_2^* - \frac{s_2 h_3 h_3^*}{2} + \frac{s_1^* h_3 h_3^*}{2} - \frac{s_3^* h_1 h_3^*}{\sqrt{2}} - \frac{n_3 h_3^*}{2} - \frac{n_4 h_3^*}{2} \\ -\frac{s_3 h_1^* h_3}{\sqrt{2}} - s_1^* h_1^* h_2 + s_2^* h_1 h_1^* - n_2 h_1^* + \frac{s_1 h_3 h_3^*}{2} + \frac{s_2^* h_3 h_3^*}{2} - \frac{s_3^* h_2 h_3^*}{\sqrt{2}} - \frac{n_3 h_3^*}{2} + \frac{n_4 h_3^*}{2} \\ -\frac{s_1 h_2^* h_3}{\sqrt{2}} + \frac{s_2 h_1^* h_3}{\sqrt{2}} - \frac{s_1^* h_1^* h_3}{\sqrt{2}} - \frac{s_2^* h_2^* h_3}{\sqrt{2}} + s_3^* h_1 h_1^* + s_3^* h_2 h_2^* + \frac{n_3(h_1^*+h_2^*)}{\sqrt{2}} + \frac{n_4(h_1^*-h_2^*)}{\sqrt{2}} \end{bmatrix}. \end{aligned} \tag{79}$$

Thus, [8, eq. (22)], which is given as

$$\hat{\mathbf{H}} = \|\mathbf{h}_{BU}\|^2 \cdot \mathbf{I}_k, \tag{77}$$

is validated, and the sufficient statistic is met for MRC.

For the communications link given in Section VII-A with $N_R = 1$, the r-by-1 received sample vector is represented as

$$\mathbf{Z} = \mathbf{G} \times \mathbf{h}_{BU} + \mathbf{n} = \mathbf{H}_{CBU} \times \mathbf{s}_{ext} + \mathbf{n} \tag{78}$$

$$= \begin{bmatrix} s_1 & s_2 & \frac{s_3}{\sqrt{2}} \\ -s_2^* & s_1^* & \frac{s_3}{\sqrt{2}} \\ \frac{s_3^*}{\sqrt{2}} & \frac{s_3^*}{\sqrt{2}} & \frac{(-s_1-s_1^*+s_2-s_2^*)}{2} \\ \frac{s_3^*}{\sqrt{2}} & \frac{-s_3^*}{\sqrt{2}} & \frac{(s_2+s_2^*+s_1-s_1^*)}{2} \end{bmatrix} \times \begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix} + \begin{bmatrix} n_1 \\ n_2 \\ n_3 \\ n_4 \end{bmatrix}$$

$$= \begin{bmatrix} h_1 & h_2 & \frac{h_3}{\sqrt{2}} & 0 & 0 & 0 \\ 0 & 0 & \frac{h_3}{\sqrt{2}} & h_2 & -h_1 & 0 \\ \frac{-h_3}{2} & \frac{h_3}{2} & 0 & \frac{-h_3}{2} & \frac{-h_3}{2} & \frac{(h_1+h_2)}{\sqrt{2}} \\ \frac{h_3}{2} & \frac{h_3}{2} & 0 & \frac{-h_3}{2} & \frac{h_3}{2} & \frac{(h_1-h_2)}{\sqrt{2}} \end{bmatrix} \times \begin{bmatrix} s_1 \\ s_2 \\ s_3 \\ s_1^* \\ s_2^* \\ s_3^* \end{bmatrix} + \begin{bmatrix} n_1 \\ n_2 \\ n_3 \\ n_4 \end{bmatrix}$$

$$= \begin{bmatrix} s_1 h_1 + s_2 h_2 + \frac{s_3 h_3}{\sqrt{2}} + n_1 \\ \frac{s_3 h_3}{\sqrt{2}} + s_1^* h_2 - s_2^* h_1 + n_2 \\ -\frac{s_1 h_3}{2} + \frac{s_2 h_3}{2} - \frac{s_1^* h_3}{2} - \frac{s_2^* h_3}{2} + \frac{s_3^*(h_1+h_2)}{\sqrt{2}} + n_3 \\ \frac{s_1 h_3}{2} + \frac{s_2 h_3}{2} - \frac{s_1^* h_3}{2} + \frac{s_2^* h_3}{2} + \frac{s_3^*(h_1-h_2)}{\sqrt{2}} + n_4 \end{bmatrix}.$$

With this received vector, the first step of the MRC sequence given in [8, eq. (19)] is calculated in (79), shown at the bottom of the previous page.

The second step given in [8, eq. (20)] is then used iteratively for $x \in \{1, \ldots, k\}$ to obtain the estimated data symbol vector, $\hat{\mathbf{s}}$.

With $k = 3$ and $x = 1$,

$$\hat{\mathbf{s}}(1) = \frac{\hat{\mathbf{s}}_{int}(1) + \hat{\mathbf{s}}_{int}^*(4)}{\|\mathbf{h}_{BU}\|^2} = s_1 + \frac{n_1 h_1^* + n_2^* h_2 - \frac{h_3^*(n_3 - n_4)}{2} - \frac{h_3(n_3^* + n_4^*)}{2}}{\|\mathbf{h}_{BU}\|^2}. \tag{80}$$

With $k = 3$ and $x = 2$,

$$\hat{\mathbf{s}}(2) = \frac{\hat{\mathbf{s}}_{int}(2) + \hat{\mathbf{s}}_{int}^*(5)}{\|\mathbf{h}_{BU}\|^2} = s_2 + \frac{n_1 h_2^* - n_2^* h_1 + \frac{h_3^*(n_3 + n_4)}{2} - \frac{h_3(n_3^* - n_4^*)}{2}}{\|\mathbf{h}_{BU}\|^2}. \tag{81}$$

With $k = 3$ and $x = 3$,

$$\hat{\mathbf{s}}(3) = \frac{\hat{\mathbf{s}}_{int}(3) + \hat{\mathbf{s}}_{int}^*(6)}{\|\mathbf{h}_{BU}\|^2} = s_3 + \frac{\frac{h_3^*(n_1 + n_2)}{\sqrt{2}} + \frac{h_1(n_3^* + n_4^*)}{\sqrt{2}} + \frac{h_2(n_3^* - n_4^*)}{\sqrt{2}}}{\|\mathbf{h}_{BU}\|^2}. \tag{82}$$

TABLE 7. Tabular form KR showing the relationships and locations of each element for all symbol groups in the MPO334 variant code given in (83).

SGI	Relationship	Location of Variant 1	Location of Variant 2
1.1.1	Conj. -1	(1, 1)	(2, 2)
1.2.1	Conj.	(2, 1)	(1, 2)
1.1.2	Conj. -1	(4, 1)	(2, 2)
1.2.2	Conj.	(2, 1)	(4, 2)
2.1.1	Conj. -1	(1, 1)	(3, 2)
2.2.1	Conj.	(3, 1)	(1, 2)
2.1.2	Conj. -1	(4, 1)	(3, 2)
2.2.2	Conj.	(3, 1)	(4, 2)
3.1.1	-1	(1, 1)	(4, 1)
3.2.1	-1	(1, 1)	(4, 1)
3.3.1	1	(1, 2)	(4, 2)
4.1.1	Conj. -1	(1, 1)	(2, 3)
4.2.1	Conj.	(2, 1)	(1, 3)
4.1.2	Conj.	(4, 1)	(2, 3)
4.2.2	Conj. -1	(2, 1)	(4, 3)
5.1.1	Conj. -1	(1, 1)	(3, 3)
5.2.1	Conj.	(3, 1)	(1, 3)
5.1.2	Conj.	(4, 1)	(3, 3)
5.2.2	Conj. -1	(3, 1)	(4, 3)
6.1.1	1	(1, 1)	(4, 1)
6.2.1	1	(1, 1)	(4, 1)
6.3.1	-1	(1, 3)	(4, 3)
7.1.1	1	(1, 2)	(4, 2)
7.1.2	-1	(1, 3)	(4, 3)
8.1.1	Conj. -1	(2, 2)	(3, 3)
8.2.1	Conj.	(3, 2)	(2, 3)
9.1.1	$\frac{1}{\sqrt{2}}$	(1, 2)	(1, 2)
9.2.1	1	(2, 2)	(2, 2)
9.3.1	1	(3, 2)	(3, 2)
9.1.2	$\frac{1}{\sqrt{2}}$	(4, 2)	(4, 2)

APPENDIX C
ADDITIONAL KEY RESIDUES

This appendix includes additional example matrix and tabular form KRs to allow for a better understanding of the KR definition provided in Section VI-B.

A variant of the MPO334 STC is generated using row and column permutation algorithms from [8]. The resulting variant code is represented as

$$\mathbf{G} = \begin{bmatrix} \frac{(-s_1 - s_1^* + s_2 - s_2^*)}{2} & \frac{s_3^*}{\sqrt{2}} & \frac{s_3^*}{\sqrt{2}} \\ \frac{s_3}{\sqrt{2}} & -s_2^* & s_1^* \\ \frac{s_3}{\sqrt{2}} & s_1 & s_2 \\ \frac{(s_2 + s_2^* + s_1 - s_1^*)}{2} & \frac{s_3^*}{\sqrt{2}} & \frac{-s_3^*}{\sqrt{2}} \end{bmatrix}. \tag{83}$$

The matrix form KR corresponding to (83) is determined by phase rotating symbol s_2, conjugating symbols s_2 and s_3, and permuting symbols s_1 and s_2. Note that since the left-hand column contains summation terms, the second column is used as the reference column. Also, the top and bottom terms in the reference column are weighted by $1/\sqrt{2}$; therefore, these terms are considered $s_3/\sqrt{2}$. The KR is represented as

$$\mathbf{G}_{KR} = \begin{bmatrix} \frac{(-s_2 - s_2^* + s_1 - s_1^*)}{2} & \frac{s_3}{\sqrt{2}} & \frac{s_3}{\sqrt{2}} \\ \frac{s_3^*}{\sqrt{2}} & s_1 & s_2^* \\ \frac{s_3^*}{\sqrt{2}} & s_2 & -s_1^* \\ \frac{(-s_1 - s_1^* + s_2 - s_2^*)}{2} & \frac{s_3}{\sqrt{2}} & \frac{-s_3}{\sqrt{2}} \end{bmatrix}. \tag{84}$$

The tabular form of this KR is provided in Table 7.

The matrix form KR corresponding to the NO442 base code given in (27) is determined by phase rotating and conjugating s_2 and s_4. Note that the first and third columns are the reference columns as they are the left-most columns that collectively contain all data symbols within the data symbol vector. Column two is not a reference column as it only contains data symbols that are also contained within the first reference column. The KR is represented as

$$\mathbf{G}_{KR} = \begin{bmatrix} s_1 & -s_2^* & s_3 & -s_4^* \\ s_2 & s_1^* & s_4 & s_3^* \end{bmatrix}. \tag{85}$$

REFERENCES

[1] A. D. Wyner, "The wire-tap channel," *Bell Syst. Techn. J.*, vol. 54, no. 8, pp. 1355–1387, Oct. 1975.

[2] A. Thangaraj, S. Dihidar, A. R. Calderbank, S. W. McLaughlin, and J.-M. Merolla, "Applications of LDPC codes to the wiretap channel," *IEEE Trans. Inf. Theory*, vol. 53, no. 8, pp. 2933–2945, Aug. 2007.

[3] O. Souihli and T. Ohtsuki, "The two-way MIMO wire-tap channel," in *Proc. IEEE Int. Conf. Commun.*, Dresden, Germany, 2009, pp. 1–6.

[4] A. Subramanian, A. Thangaraj, M. Bloch, and S. W. McLaughlin, "Strong secrecy on the binary erasure wiretap channel using large-girth LDPC codes," *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 3, pp. 585–594, Sep. 2011.

[5] C. Sahin, B. Katz, and K. R. Dandekar, "Secure and robust symmetric key generation using physical layer techniques under various wireless environments," in *Proc. IEEE Radio Wireless Symp. (RWS)*, Austin, TX, USA, 2016, pp. 211–214.

[6] L. Zhao, X. Zhang, J. Chen, and L. Zhou, "Physical layer security in the age of artificial intelligence and edge computing," *IEEE Wireless Commun.*, vol. 27, no. 5, pp. 174–180, Oct. 2020.

[7] M. R. Cribbs, R. A. Romero, and T. T. Ha, "Orthogonal STBC set building and physical layer security application," in *Proc. IEEE 21st Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Atlanta, GA, USA, May 2020, pp. 1–5.

[8] M. R. Cribbs, R. A. Romero, and T. T. Ha, "Physical layer security for multiple-input multiple-output systems by alternating orthogonal space-time block codes," *IEEE Open J. Commun. Soc.*, vol. 1, pp. 1503–1523, 2020.

[9] R. Ma, L. Dai, Z. Wang, and J. Wang, "Secure communication in TDS-OFDM system using constellation rotation and noise insertion," *IEEE Trans. Consum. Electron.*, vol. 56, no. 3, pp. 1328–1332, Aug. 2010.

[10] M. I. Husain, S. Mahant, and R. Sridhar, "CD-PHY: Physical layer security in wireless networks through constellation diversity," in *Proc. IEEE Mil. Commun. Conf. (MILCOM)*, Orlando, FL, USA, 2012, pp. 1–9.

[11] T. Allen, J. Cheng, and N. Al-Dhahir, "Secure space-time block coding without transmitter CSI," *IEEE Wireless Commun. Lett.*, vol. 3, no. 6, pp. 573–576, Dec. 2014.

[12] T. Xiong, W. Lou, J. Zhang, and H. Tan, "MIO: Enhancing wireless communications security through physical layer multiple inter-symbol obfuscation," *IEEE Trans. Inf. Forensics Security*, vol. 10, pp. 1678–1691, 2015.

[13] B. Chen, C. Zhu, W. Li, J. Wei, V. C. M. Leung, and L. T. Yang, "Original symbol phase rotated secure transmission against powerful massive MIMO eavesdropper," *IEEE Access*, vol. 4, pp. 3016–3025, 2016.

[14] M. Xiang-Ning, L. Kai-Jia, and L. Hao, "A physical layer security algorithm based on constellation," in *Proc. IEEE 17th Int. Conf. Commun. Technol. (ICCT)*, Chengdu, China, 2017, pp. 50–53.

[15] D. Xu, P. Ren, Q. Du, L. Sun, and Y. Wang, "Physical layer security improvement by constellation selection and artificial interference," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, San Francisco, CA, USA, 2017, pp. 1–6.

[16] J. Hua, S. Jiang, W. Lu, Z. Xu, and F. Li, "A novel physical layer encryption algorithm based on statistical characteristics of time-selective channels," *IEEE Access*, vol. 6, pp. 38225–38233, 2018.

[17] P. Ramabadran *et al.*, "A novel physical layer encryption scheme to counter eavesdroppers in wireless communications," in *Proc. 25th IEEE Int. Conf. Electron. Circuits Syst. (ICECS)*, Bordeaux, France, 2018, pp. 69–72.

[18] G. T. Rendon, W. K. Harrison, M. A. C. Gomes, and J. P. Vilela, "Nested QPSK encoding for information theoretic security," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Kansas City, MO, USA, 2018, pp. 1–6.

[19] C. Xi, Y. Gao, S. Nan, and P. Lei, "Constellation symbol obfuscation design approach for physical layer security," in *Proc. 10th Int. Conf. Commun. Softw. Netw. (ICCSN)*, Chengdu, China, 2018, pp. 264–269.

[20] S. V. Pechetti, A. Jindal, and R. Bose, "Exploiting mapping diversity for enhancing security at physical layer in the Internet of Things," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 532–544, Feb. 2019.

[21] N. Sanandaji and A. Falahati, "A hidden OSTBC scheme to enhance physical layer security by employing a pseudorandom precoder," *IEEE Trans. Commun.*, vol. 68, no. 1, pp. 363–375, Jan. 2020.

[22] V. Tarokh, H. Jafarkhani, and A. R. Calderbank, "Space-time block codes from orthogonal designs," *IEEE Trans. Inf. Theory*, vol. 45, no. 5, pp. 1456–1467, Jul. 1999.

[23] A. Lozano and N. Jindal, "Transmit diversity vs. spatial multiplexing in modern MIMO systems," *IEEE Trans. Wireless Commun.*, vol. 9, no. 1, pp. 186–197, Jan. 2010.

[24] *IEEE Standard for Air Interface for Broadband Wireless Access Systems*, IEEE Standard 802.16-2017, 2018.

[25] D. Wübben, R. Böhnke, J. Rinas, V. Kühn, and K. D. Kammeyer, "Efficient algorithm for decoding layered space-time codes," *Electron. Lett.*, vol. 37, no. 22, pp. 1348–1350, 2001.

[26] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger, "Closest point search in lattices," *IEEE Trans. Inf. Theory*, vol. 48, no. 8, pp. 2201–2214, Aug. 2002.

[27] L. G. Barbero and J. S. Thompson, "A fixed-complexity MIMO detector based on the complex sphere decoder," in *Proc. IEEE 7th Workshop Signal Process. Adv. Wireless Commun.*, Cannes, France, 2006, pp. 1–5.

[28] Z. Guo and P. Nilsson, "Algorithm and implementation of the K-best sphere decoding for MIMO detection," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 3, pp. 491–503, Mar. 2006.

[29] C. Studer, A. Burg, and H. Bolcskei, "Soft-output sphere decoding: Algorithms and VLSI implementation," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 2, pp. 290–300, Feb. 2008.

[30] D. Wubben, R. Bohnke, V. Kuhn, and K.-D. Kammeyer, "MMSE extension of V-BLAST based on sorted QR decomposition," in *Proc. IEEE 58th Veh. Technol. Conf. (VTC-Fall)*, vol. 1. Orlando, FL, USA, 2003, pp. 508–512.

[31] M. da Silva and F. Monteiro, Eds., *MIMO Processing for 4G and Beyond: Fundamentals and Evolution*. Boca Raton, FL, USA: Taylor Francis, 2014.

[32] S. Baro, G. Bauch, A. Pavlic, and A. Semmler, "Improving BLAST performance using space-time block codes and turbo decoding," in *Proc. IEEE. Global Telecommun. Conf. Rec. (GLOBECOM)*, vol. 2. San Francisco, CA, USA, 2000, pp. 1067–1071.
[33] C.-Y. Hung and T.-H. Sang, "A sphere decoding algorithm for MIMO channels," in *Proc. IEEE Int. Symp. Signal Process. Inf. Technol.*, Vancouver, BC, Canada, 2006, pp. 502–506.
[34] *Technical Specification Group Radio Access Network; NR; Physical Channels and Modulation, Version 16.2.0.*, 3GPP Standard (TS) 38.211, Jun. 2020. [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3213
[35] C. E. Shannon, "Communication theory of secrecy systems," *Bell Syst. Techn. J.*, vol. 28, no. 4, pp. 656–715, Oct. 1949.
[36] S. M. Alamouti, "A simple transmit diversity technique for wireless communications," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 8, pp. 1451–1458, Oct. 1998.
[37] *MATLAB Documentation*, The MathWorks, Inc., Natick, MA, USA. Accessed: Apr. 30, 2021 [Online]. Available: https://www.mathworks.com/help/index.html
[38] M. J. Wiener, "Exhaustive key search," in *Encyclopedia of Cryptography and Security*, H. C. A. van Tilborg, Ed. Boston, MA, USA: Springer, 2005, pp. 206–209. [Online]. Available: https://doi.org/10.1007/0-387-23483-7_147

MICHAEL R. CRIBBS (Graduate Student Member, IEEE) received the B.S.E.E. degree from the University of Kansas, Lawrence, KS, USA, in 2009, and the M.S.E.E. degree from the Naval Postgraduate School, Monterey, CA, USA, in 2015, where he is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering.

He enlisted in the United States Navy in 2004 and received his commission in the United States Navy in 2009. He continues to serve on active duty as an Engineering Duty Officer with Naval Postgraduate School. His research interests include communications and coding theory.

RIC A. ROMERO (Senior Member, IEEE) received the B.S.E.E. degree from Purdue University, West Lafayette, IN, USA, in 1999, the M.S.E.E. degree from the University of Southern California, Los Angeles, CA, USA, in 2004, and the Ph.D. degree in electrical and computer engineering from The University of Arizona, Tucson, AZ, USA, in 2010.

He was a Senior Multidisciplined Engineer II with Raytheon Missile Systems, Tucson, from 1999 to 2010. He was involved in various communications, radar, and research and development programs. He was also a Graduate Research Assistant with the Laboratory for Sensor and Array Processing, The University of Arizona, from 2007 to 2010. He is currently an Associate Professor with the Department of Electrical and Computer Engineering, Naval Postgraduate School, Monterey. His research interests include the general areas of radar, sensor information processing, and communications.

Dr. Romero was awarded the 2004 Corporate Excellence in Technology Award, which is a company-wide technical prize at Raytheon Corporation. He was also granted the Raytheon Advanced Scholarship Program Fellowships, from 2002 to 2004 and from 2005 to 2007.

TRI T. HA (Life Fellow, IEEE) has been a Professor with the Department of Electrical and Computer Engineering, Naval Postgraduate School, Monterey, CA, USA, since 1989. He was with Fairchild Industries and General Telephone and Electronics Corporation and as an Associate Professor with the Virginia Polytechnic Institute and State University, Blacksburg, VA, USA, for four years. He has authored three textbooks. His current research interests are in wireless communications and cyber warfare.