Dynamic Forward Error Correction Coding to Avoid Detection in Airborne Tactical Networks

Lars Lundberg
Department of Computer Science
Blekinge Institute of Technology
371 79 Karlskrona, Sweden
lars.lundberg@bth.se

Alexander Westerhagen
Saab AB
Surveillance
371 30 Karlskrona, Sweden
alexander.westerhagen@saabgroup.com

Dragos Ilie
Department of Computer Science
Blekinge Institute of Technology
371 79 Karlskrona, Sweden
dragos.ilie@bth.se

Håkan Grahn
Department of Computer Science
Blekinge Institute of Technology
371 79 Karlskrona, Sweden
hakan.grahn@bth.se

Bo Granbom
Saab AB
Aeronautics
581 88 Linköping, Sweden
bo.granbom@saabgroup.com

***Abstract*—Here we present a novel routing protocol HDARP+ for airborne tactical networks that use directional antennas. HDARP+ extends the existing protocol HDARP (Hostile-Direction Aware Routing Protocol) by reducing the risk for detection by adversary aircraft even further. Compared to HDARP, the extension in HDARP+ introduces dynamic Forward Error Correction (FEC) coding. The FEC code is dynamic in the sense that different FEC codes, or no FEC code, will be used depending on the relative position of the receiver and adversary aircraft. We evaluate three different Reed-Solomon FEC codes based on three criteria: the ability to transmit in the presence of adversaries without being detected, the reduction of the effective communication bandwidth, and the implementation cost in terms of the sizes of lookup tables for encoding and decoding. We argue that (variations of) HDARP+ will be implemented in future airborne tactical networks.**

This paper was originally presented at the NATO Science and Technology Organization Symposium (ICMCIS) organized by the Information Systems Technology (IST) Panel, IST-205-RSY – the ICMCIS, held in Koblenz, Germany, 23-24 April 2024.

Keywords—forward error correction, avoiding detection, routing, directional antennas, Reed-Solomon codes, dynamic FEC

I. Introduction

Radio communication between airborne platforms is usually implemented using omnidirectional antennas. One problem with omnidirectional antennas is that they make it relatively easy for an adversary to position the sender, which is a major drawback particularly when defending one's own territory. The defenders typically have access to local infrastructure like ground-based radar stations, which helps to create a superior situational awareness (SA) picture of their own territory. This means that the defenders have an information advantage in the sense that they know the position and speed of the attacking aircraft; the attackers usually do not know the position of the defending aircraft. In order to maintain this tactical advantage as long as possible, it is important to avoid disclosing one's own position by emitting radio frequency energy that may be detected.

The interest for directional antennas has increased for military radio networks. There are several reasons for this: the transmission range can be extended by focusing the power in the direction of the receiver, space multiplexing based on different transmission directions and different angels of arrival for incoming signals becomes possible, and the directional antenna hardware makes it possible to do selective jamming of hostile nodes. For these and similar reasons, future military aircraft will be equipped with directional antennas.

By using directional antennas and smart routing solutions, it is possible to avoid, or at least significantly delay, detection by adversary nodes. In a previous study we have defined and evaluated a communication protocol HDARP (Hostile-Direction Aware Routing Protocol – patent pending) for radio communication using directional antennas [1]. Compared to existing routing protocols for networks using directional antennas, HDARP reduces the risk for detection by hostile nodes significantly. However, there are some situations where detection cannot be avoided unless one restrains from using radio communication entirely. In this paper we define an extension (HDARP+) of the HDARP protocol that uses dynamic Forward Error Correction (FEC) coding to increase the number of cases where it is possible to avoid detection by hostile nodes.

II. Background

A. The HDARP protocol

The HDARP protocol is a geographic routing protocol [2] for directional antennas. HDARP uses Situation Awareness (SA) data. SA data contain information about the velocity (speed and direction) and position of all (known) aircraft; both friendly and adversary aircraft. SA data are continuously updated and exchanged among the aircraft in the network. SA data contain important tactical information that the pilots need to carry out their mission. A key advantage of HDARP is the utilization of SA data in its routing decision making process. As a consequence, the overhead that HDARP needs to select appropriate routes is minimal, since SA data are needed by the pilots anyway.

The directional antenna can, using phasing and positive interference, create a signal beam in the direction of the receiver. The antenna system at each aircraft is arranged to give 360 degree coverage. The width of the transmission beam, or main lobe, is 15-30 degrees depending on the design of the antenna (due to the nature of radio signal interference there will also be smaller side lobes in other directions).

Detection of a message transmission can be achieved at approximately three times the distance at which the message can be received with an acceptable Bit Error Rate (BER). The reason for this is that the lowest Signal to Noise Ratio (SNR) value that gives an acceptable bit error rate during radio transmission is 10 dB, i.e., $SNR_{dB} = 10 = 10\log_{10}(P_s/P_n)$, where P_s is the average power during the transmission of a bit and P_n is the average power of the noise. For $SNR_{dB} = 10$, we get $P_s/P_n = 10$. As a consequence, the signal power is on par with the noise power after a distance that is three times as long as the distance where the SNR is 10 dB, i.e., $10/3^2 = 1.11 \approx 1$ (the signal power decreases proportional to the square of the distance).

Fig. 1 shows a scenario with three friendly (blue) aircraft and one adversary (red) aircraft. Since the distance from A to the adversary is less than three times the distance from A to C, and since C and the adversary are almost in the same direction from A (at least the direction difference is less than 15 degrees), node A cannot send a message directly to C if detection by the adversary should be avoided. Since the HDARP protocol has information about the location and velocity of all aircraft, the protocol will route messages from A to C via B.

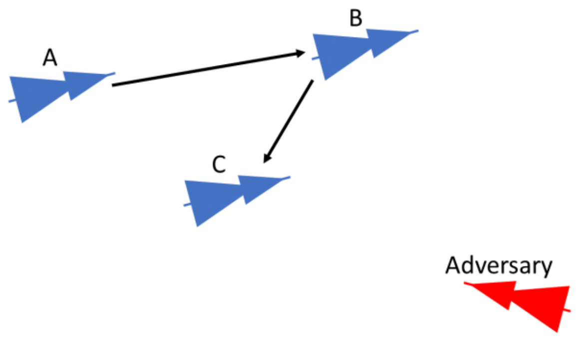

Fig. 1. Three blue aircraft (A, B and C). In order to avoid detection from the adversary, communication from A to C must go through B.

Unfortunately, HDARP cannot avoid detection in all situations. Fig. 2 shows a scenario with two friendly aircraft (A and B) and one adversary. If A needs to send a message to B, the message has to go directly from A to B since there are no other friendly aircraft. The radio signals from A to B will of course continue also beyond B, and the adversary will be able to detect these signals (even if the SNR for these radio signals is too low to decode them with a reasonable BER). Data on tactical links are usually encrypted, which makes it impossible for an adversary to interpret the content even if the BER is reasonable.

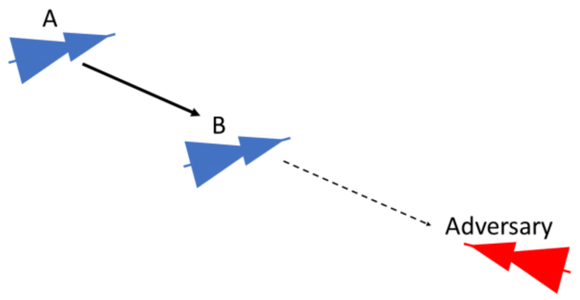

Fig. 2. If A needs to communicate with B and there are no other blue aircraft, detection from the adversary cannot be avoided.

B. Relation between SNR and BER.

If we assume that the power of the noise follows the normal distribution (Gaussian noise) and Binary Phase-Shift Keying (BPSK) modulation we can use the *Q*-function [3][26] to calculate the bit error rate (BER) based on the SNR. $Q(x)$ is the probability that a standard normal random variable takes a value larger than x.

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^{\infty} e^{-u^2/2} du \qquad (1)$$

When calculating the BER_{SNR} for a certain $SNR = P_s/P_n$ we set $x = \sqrt{2P_s/P_n}$ in (1) [26].

$$BER_{SNR} = Q\left(\sqrt{\frac{2P_s}{P_n}}\right) = \frac{1}{\sqrt{2\pi}} \int_{\sqrt{\frac{2P_s}{P_n}}}^{\infty} e^{-\frac{u^2}{2}} du \quad (2)$$

By using the *Q*-function in MATLAB we get $BER_{10} = Q(\sqrt{20}) = \frac{1}{\sqrt{2\pi}} \int_{\sqrt{20}}^{\infty} e^{-u^2/2} du \approx 3.87 \cdot 10^{-6}$ from (2). This means that $3.87 \cdot 10^{-6}$ is the highest acceptable bit error rate in the network we consider.

C. Forward Error Correction Codes

In Forward Error Correction (FEC) codes, the sender encodes the message in a redundant way. The redundancy allows the receiver to correct a limited number of errors. The cost for using FEC is reduced effective bandwidth due to the redundant error correction bits used, and the time and complexity needed to encode and decode the FEC code [4].

One well-known FEC code is the Reed-Solomon (RS) code [5]. RS codes operate on a block of data and achieve the optimal (largest possible) minimum distance between codewords of any linear code with the same block-length (the Singleton bound [25]). Compared to binary block codes which suffer from limited applicability due to coding the symbols with an alphabet only containing the values 0 and 1, RS coding can be defined as a non-binary code with symbols made up of bit sequences. The length of each symbol is s bits. Some symbols carry payload information, and some symbols are used for error recovery. Let k be the number of symbols in a block that carry payload information and n be the total number of symbols in the block, then $t = n - k$ is the number of symbols used for error correction. The normal notation for such a code is $RS(n,k)$ using s-bit symbols. An $RS(n,k)$ code can correct up to $\lfloor t/2 \rfloor$ erroneous symbols at arbitrary locations in the block, where $t = n - k$.

For instance, an RS(7,5) code (with $t = n - k = 7 - 5 = 2$), can correct $\lfloor t/2 \rfloor = 1$ symbol error in the block consisting of 7 symbols. If the symbol length s is 3, the length of each code block is $ns = 7 \cdot 3 = 21$ bits. The effective bandwidth is 5/7 = 0.71= 71% of the original bandwidth, since there are 5 payload symbols in a block consisting of 7 symbols (2 symbols carry error correction information). If we change the code from RS(7,5) to RS(7,3) we increase the error correction capabilities of the code, since 2 symbol errors can be corrected ($t = n - k = 7 - 3 = 4$). The cost for doing this is that the effective bandwidth is reduced to 3/7 = 0.43= 43% of the original bandwidth. If we change the code to RS(7,1), 3 symbol errors can be corrected. The cost for doing this is that effective bandwidth is reduced to 1/7 = 0.14 = 14% of the original bandwidth.

In general, longer RS codes have better performance in the sense that they can recover from many symbol errors with a limited reduction of the effective bandwidth. For example, in the NASA Voyager program, transmission between the space craft and Earth used RS(255,223) with 8-bit (one byte) symbols. In that case, the original message (consisting of 223 bytes) could be recovered as long as at most 16 bytes had errors during the transmission; the effective bandwidth was 223/255 = 0.87 = 87% of the original bandwidth.

Sending long blocks, for example 255 bytes, is not a viable solution in fast real-time communication between fighter aircraft, since encoding and decoding becomes computationally challenging, and thus time consuming, for large blocks and since no part of the message can be interpreted until all of the 255 bytes have been received and decoded.

III. RELATED WORKS

A. Routing Protocols in Airborne Tactical Networks

HDARP, and thus also HDARP+, is a routing protocol for Flying Ad-hoc NETwork (FANET). Oubbati et al. have done a survey of over 60 routing protocols for FANETs [14]. Most of these protocols are designed for UAVs in non-military contexts, and none of these protocols consider the risk of being detected by adversaries.

FANET is a special case of Mobile Ad-hoc NETwork (MANET). Some popular routing protocols for MANETs are AODV (Ad-hoc On-demand Distance Vector) [20], BATMAN (Better Approach To Mobile Ad-hoc Network) [21], DSDV (Destination-Sequenced Distance-Vector) [22], DSR (Dynamic Source Routing) [23], and OLSR (Optimized Link State Routing) [24]. These protocols have been compared to HDARP in radio networks with directional antennas, and none of them could avoid adversary detection to the same extent as HDARP [1].

Most studies on airborne tactical networks do not consider the implications that the presence of adversaries may have on routing [15][16][17][18]. Lu et al. [19] consider routing using directional antennas in ad-hoc networks with hostile nodes called detection systems. However, the nodes in that study are not mobile, and no FEC codes are used.

B. Dynamic Forward Error Correction

Dynamic FEC, in the form of RS coding, has been used for video transmission over unreliable networks [7]. As discussed in Section II.C, large RS blocks have better performance, but too long blocks can lead to intolerable delays in real-time video transmission. Depending on the (measured or estimated) dynamic BER, the size of the RS code can be dynamically tuned. In order to estimate or measure the dynamic BER, these video transmission systems contain some kind of feedback from the receiver to the sender [11][12].

The RS code can also be dynamic depending on the type of video frame that is transmitted, e.g., Intra (I) frames are often coded with a stronger FEC (i.e., with more error correction information) than Predicted (P) frames [6]. Dynamic FEC coding is also used to optimize video frame rates without increasing data loss [8]. Wu et al. propose adaptive (dynamic) joint source-FEC coding to maximize streaming video quality over TCP in wireless networks [9]. A similar study done by a partly overlapping group of authors looked at adaptive source-FEC coding for surveillance video over wireless networks [10].

Herrero and St-Pierre suggest that dynamic FEC coding should be used in wireless real-time IoT networks. Their algorithm dynamically balances the activation of FEC coding based on traffic conditions in order to optimize throughput and packet loss [13].

C. Research Gap

There was a research gap concerning routing protocols that avoid, or minimize the risk of, adversary detection. HDARP filled a part of that gap. From Section II.B it is clear that to the best of our knowledge no one has used dynamic FEC to reduce the risk of adversary detection in airborne tactical networks even further. In this paper we show that HDARP+ fills a part of that gap.

IV. HDARP+

The HDARP+ protocol extends the HDARP protocol by using dynamic FEC in situations where detection by an adversary cannot be avoided in HDARP, e.g., situations like the one shown in Fig. 2. The main idea in HDARP+ is to avoid detection by the adversary by reducing the transmission power in situations like the one shown in Fig. 2. However, when the transmission power is reduced the SNR decreases and the BER increases as discussed in Section II.B. To compensate for the increased BER, HDARP+ uses FEC. This means that if we want the BER, after error correction, to stay the same the SNR changes for different coding schemes. For instance, if we use a FEC code that can correct many errors, we can reduce the SNR and still keep an acceptable BER after error correction.

HDARP+ uses RS codes with n = 7 and s = 3. Three different variations of these RS codes are used corresponding to k = 5, 3 and 1, respectively. This means that we use the three codes RS(7,5), RS(7,3) and RS(7,1). In all three cases the block length is ns = 7*3 = 21 bits. The reason for selecting this block length is that the blocks are short enough to make it possible to do encoding and decoding based on table lookup. We will discuss this further in the next section.

The use of FEC in HDARP+ is dynamic in two aspects:

- FEC coding is not always used. For instance, if B wants to send a message to A in Fig. 2, no FEC is needed since there is no adversary behind A that could detect the transmission from B to A.
- The amount of redundancy, i.e., the value of k depends on the relative difference between the distance between the sender and the receiver and the distance between the sender and the adversary.

Fig 3. shows a group of l friendly aircraft (l = 4 in this case). We now define the concept *distance factor*. Based on SA data, each node (aircraft) maintains a table with l-1 distance factors f_i. A distance factor f_i denotes the distance from the sender to the closest adversary in the direction of node i (as seen from the sender) divided with the distance from the sender (i.e., the node where the table is maintained) to node i. Table I shows the distances from node A to the other nodes in Fig. 3. This information is part of SA data. Based on the values in Table I and the location of the aircraft in Fig. 3 (adversary X is in the same direction as B and adversary Y is in the same direction as C, as seen from node A), node A will compute the following distance factors f_i: f_B = 30/9 = 3.33, f_C = 25/11 = 2.27, and f_D = ∞/12 = ∞ (there is no adversary in the same direction as node D).

If $f_i > 3$, no FEC is needed when sending to node i. If $f_i \leq 3$, some FEC is needed if it is important that the sender should not be detected by an adversary aircraft. This means that in the scenario shown in Fig. 3 node A will not use any FEC when sending to nodes B and D, but some FEC will be used when sending to node C. The exact values of f_i when different FEC codes should be used will be discussed in the next section.

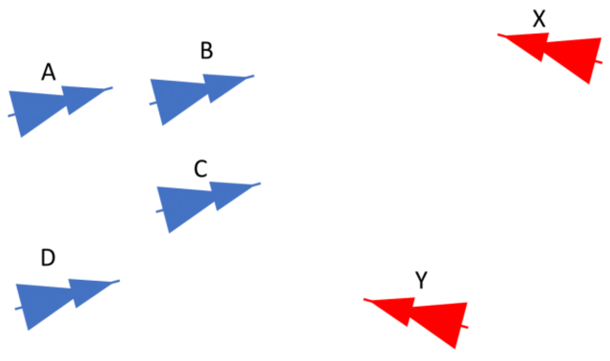

Fig. 3. A group with four friendly aircraft (A, B, C and D) and two adversaries (X and Y).

TABLE I. DISTANCE FROM NODE A TO THE OTHER NODES IN FIG. 3.

Node	B	C	D	X	Y
Distance	9	11	12	30	25

V. PERFORMANCE EVALUATION

The performance of the three FEC codes (i.e., RS(7,5), RS(7,3) and RS(7,1) using 3-bit symbols) used in HDARP+ will be evaluated based on three criteria:

- How low distance factors f_i the FEC code can handle without adversary detection.
- The effective bandwidth when using the FEC code.
- The implementation cost of the FEC code in terms of the size of the tables used for encoding and decoding.

The performance of the three FEC codes is summarized in Table II.

A. *Evaluating RS(7,5)*

Let the probability of receiving a symbol correctly be α. Since we are using 3-bit symbols we have

$$\alpha = (1 - \text{BER})^3 \quad (3)$$

where BER is the bit error rate. Note that a symbol error may be caused by 1 to 3 bit errors. In general, 3 bit errors may cause 1 to 3 symbol errors depending on where the bit errors occur. However, if we assume that the probability of a bit error is independent of any previous bit errors (i.e., no burst error model), Equation (3) covers all cases and denotes the probability that a symbol is received correctly.

Let the probability that we can correctly decode the five symbols containing payload information in the RS(7,5)-block consisting of seven symbols be β, then

$$\beta = \alpha^7 + 7(1-\alpha)\alpha^6 \quad (4)$$

where α^7 is the probability that all seven symbols are received correctly, and $7(1-\alpha)\alpha^6$ is the probability that one of the seven symbols is received incorrectly. Any additional symbols received incorrectly cannot be corrected by this FEC code.

As discussed in Section II.B, the maximum acceptable bit error rate is $3.87*10^{-6}$. When using RS(7,5) with 3-bit symbols we transmit payload information consisting of 5 symbols that contain 3 bits each, i.e., 15 bits of payload in each block of 21 bits. Let γ denote the minimal acceptable probability that these 15 bits are received correctly without any FEC coding. In that case, $\gamma = (1 - 3.87 * 10^{-6})^{15} \approx 0.99994195$. We now want to find a value α so that the probability β (five symbols can be correctly decoded using RS(7,5)) becomes equal to γ. For α = 0.9983327, we get $\beta = \alpha^7 + 7(1-\alpha)\alpha^6$ = 0.99994195. This means that when using RS(7,5) the BER that corresponds to $3.87*10^{-6}$ without RS coding is

$$\text{BER} = 1 - \alpha^{1/3} = 1 - 0.9983327^{1/3} = 0.000556 \quad (5)$$

By using the *Q*-function in MATLAB we see that Q(3.2605) = 0.000556. This corresponds to an SNR = P_s/P_n = $3.2605^2/2$ = 5.32. Since the power decreases proportional to the square of the distance we get an SNR of 5.32 at a distance that is x times further away from the sender according to the formula $10/x^2 = 5.32$, which gives x = 1.37. This means that when using RS(7,5) we could extend the communication range with 37% without increasing the risk of being detected by a hostile node. This means that we can use RS(7,5) when communicating with nodes for which $3 \geq f_i > 3/1.37 = 2.19$.

As discussed in Section II.C, the effective bandwidth is reduced to 5/7 = 0.7143 = 71% of the original bandwidth.

Obviously, encoding and decoding of messages must be done in real-time. In order to avoid delays, encoding and decoding should be done through table lookup. To encode a payload message consisting of 15 bits to an RS(7,5) block with seven 3-bit symbols, a word length of 21 bits is needed (there are 21 bits in an RS(7,5)-block). Since we encode 15 bits, we need 2^{15} = 32k such words. If we assume 32-bit words in the aircraft's computer system, we need 4 bytes * 32k = 128 kB memory for the encoding table. In the decoding table we need words with 15 bits (the decoded information will be five 3-bit symbols). Since there are 21 bits in the RS(7,5)-block, we need 2^{21} = 2M such words. If we use two bytes to store the 15 bits, we need 2 byte * 2M = 4 MB memory for the decoding table.

B. *Evaluating RS(7,3)*

Let again the probability of receiving a symbol correctly be α. Since we are using 3-bit symbols we again have

$$\alpha = (1 - \text{BER})^3 \quad (6)$$

Let the probability that we can correctly decode the three symbols containing payload information in the RS(7,3)-block consisting of seven symbols be β, then

$$\beta = \alpha^7 + 7(1-\alpha)\alpha^6 + 21\,(1-\alpha)^2\alpha^5 \quad (7)$$

where α^7 is the probability that all seven symbols are received correctly, $7(1-\alpha)\alpha^6$ is the probability that one of the seven symbols is received incorrectly, and $21\,(1-\alpha)^2\alpha^5$ is the probability that two of the seven symbols are received incorrectly. Any additional symbols received incorrectly cannot be corrected by this FEC code.

As discussed previously, the maximum acceptable bit error rate is $3.87*10^{-6}$. When using RS(7,3) with 3-bit symbols we transmit payload information consisting of 3 symbols that contain 3 bits each, i.e., 9 bits of payload in each block of 21 bits. Let γ denote the minimal acceptable probability that these 9 bits are received correctly without any FEC coding. In that case, $\gamma = (1 - 3.87 * 10^{-6})^9 \approx 0.99996517$. We now want to find a value α so that the probability β (three symbols can be correctly decoded using RS(7,3)) becomes equal to γ. For α = 0.989915, we get $\alpha^7 + 7(1-\alpha)\alpha^6 + 21\,(1-\alpha)^2\alpha^5$ = 0.9996517. This means that when using RS(7,3) the BER that corresponds to 3.87*10^{-6} without RS coding is

$$\text{BER} = 1 - \alpha^{1/3} = 1 - 0.989915^{1/3} = 0.003373 \quad (8)$$

By using the *Q*-function in MATLAB we see that *Q*(2.7091) = 0.003373. This corresponds to an $SNR = P_s/P_n$ = 2.7091^2/2 = 3.67. Since the power decreases proportional to the square of the distance we get an SNR of 3.67 at a distance that is *x* times further away from the sender according to the formula 10/x^2 = 3.67, which gives *x* = 1.65. This means that when using RS(7,3) we could extend the communication range with 65% without increasing the risk of being detected by a hostile node. This means that we can use RS(7,3) when communicating with nodes for which 2.19 ≥ f_i > 3/1.65 = 1.82. For performance reasons, the system will switch to RS(7,5) for nodes where f_i > 2.19, or No FEC when f_i > 3.

As discussed in Section II.C, the effective bandwidth is reduced to 3/7 = 0.43 = 43% of the original bandwidth.

To encode a payload message consisting of 9 bits to a RS(7,3) block with 3-bit symbols, a word length of 21 bits is needed (there are 21 bits in the RS(7,3)-block). Since we encode 9 bits, we need 2^9 = 512 such words. If we assume 32-bit words in the aircraft's computer system, we need 4 bytes * 512 = 2 kB memory for the encoding table. In the decoding table we need words with 9 bits (the decoded information will be three 3-bit symbols). Since there are 21 bits in the RS(7,3)-block, we need 2^{21} = 2M such words. If we use two bytes to store the 9 bits, we need 2 byte * 2M = 4 MB memory for the decoding table.

C. Evaluating RS(7,1)

Let again the probability of receiving a symbol correctly be α. Since we are using 3-bit symbols we again have

$$\alpha = (1 - \text{BER})^3 \quad (9)$$

Let the probability that we can correctly decode the symbol containing payload information in the RS(7,1)-block consisting of seven symbols be β, then

$$\beta = \alpha^7 + 7(1-\alpha)\alpha^6 + 21\,(1-\alpha)^2\alpha^5 + 35\,(1-\alpha)^3\alpha^4 \quad (10)$$

where α^7 is the probability that all seven symbols are received correctly, and $7(1-\alpha)\alpha^6$ is the probability that one of the seven symbols is received incorrectly, $21\,(1-\alpha)^2\alpha^5$ is the probability that two of the seven symbols are received incorrectly, and $35\,(1-\alpha)^3\alpha^4$ is the probability that three of the seven symbols are received incorrectly.

As discussed previously, the maximum acceptable bit error rate is 3.87*10^{-6}. When using RS(7,1) with 3-bit symbols we transmit payload information consisting of one symbol that contains 3 bits, i.e., 3 bits of payload in each block of 21 bits. Let γ denote the minimal acceptable probability that these 3 bits are received correctly without any FEC coding. In that case, $\gamma = (1 - 3.87 * 10^{-6})^3 \approx 0.99998839$. We now want to find a value α so that the probability β (one symbol can be correctly decoded using RS(7,1)) becomes equal to γ. For α = 0.975645, we get $\alpha^7 + 7(1-\alpha)\alpha^6 + 21\,(1-\alpha)^2\alpha^5 + 35\,(1-\alpha)^3\alpha^4$ = 0.99998839, i.e., when using RS(7,1) the BER that corresponds to 3.87*10^{-6} without RS coding is

$$\text{BER} = 1 - \alpha^{1/3} = 1 - 0.975645^{1/3} = 0.008185 \quad (11)$$

By using the Q-function in MATLAB we see that Q(2.40055) = 0.008185. This corresponds to an SNR = P_s/P_n = 2.40055^2/2 = 2.88. Since the power decreases proportional to the square of the distance we get an SNR of 2.88 at a distance that is *x* times further away from the sender which according to the formula 10/x^2 = 2.88 gives *x* = 1.86. This means that when using RS(7,1) we could extend the communication range with 86% without increasing the risk of being detected by a hostile node. This means that we can use RS(7,1) when communicating with nodes for which 1.82 ≥ f_i > 3/1.86 = 1.61.

As discussed in Section II.C, the effective bandwidth is reduced to 1/7 = 0.14 = 14% of the original bandwidth.

To encode a payload message consisting of 3 bits to a RS(7,1) block with seven 3-bit symbols, a word length of 21 bits is needed. Since we encode 3 bits, we need 2^3 = 8 such words. If we assume 32-bit words in the aircraft's computer system, we need 4 bytes * 8 = 32 bytes memory for the encoding table. In the decoding table we need words with 3 bits (the decoded information will be one 3-bit symbol). Since there are 21 bits in the RS(7,1)-block, we need 2^{21} = 2M such words. If we use one byte to store 2x3 bits, we need 0.5 byte * 2M = 1 MB memory for the decoding table.

TABLE II. PERFORMANCE OF THE FEC CODES USED IN HDARP+.

Node	f_i range	Effective bandwidth	Table size	
			encode	decode
No FEC	f_i > 3	1	0	0
RS(7,5)	3 ≥ f_i > 2.19	0.71	128 kB	4 MB
RS(7,3)	2.19 ≥ f_i > 1.82	0.43	2 kB	4 MB
RS(7,1)	1.82 ≥ f_i > 1.61	0.14	32 bytes	1 MB

VI. ANALYSIS AND DISCUSSION

Table II shows that the implementation cost, in terms of memory size for the lookup tables used for encoding and decoding, is very limited. We need less than 10 MB for all tables, which is completely unproblematic. The table also shows that when going from "No FEC" to RS(7,5) there is a significant expansion of the safe communication area, i.e., the area where we can maintain undetectable communication with friendly nodes (the distance factor f_i is reduced from 3 to 2.19). Moreover, the reduction of effective bandwidth is limited when going from "No FEC" to RS(7,5) (reduced from 1 to 0.71). However, the trade-off in terms of expanding the safe communication area and reducing the effective bandwidth becomes less favorable when we go to RS codes with more redundancy, e.g., when going from RS(7,3) to RS(7,1) there is a rather limited expansion of the safe area (the distance factor f_i is reduced from 1.82 to 1.61), while the reduction of effective bandwidth is rather significant (reduced from 0.43 to 0.14). Due to this rather unfavorable cost/benefit trade-off, RS(7,1) may be excluded in future real-world implementations of HDARP+ in airborne platforms. However, RS(7,5) and RS(7,3) are highly relevant in real-world implementations.

In the normal case, all friendly nodes have access to the same SA data, and in that case the receiver can easily determine which FEC code (if any) the sender has used. This means that decoding becomes unambiguous. However, since the network operates in a dynamic real-time environment there could be situations when all friendly nodes may not have identical SA data. For that reason, we may need to do parallel decoding of the received packages and select the correct version after decoding. Since we have a limited number of codes (three RS codes and the no-FEC alternative), and since all decoding is done through table lookup, the overhead for doing parallel decoding is very small. The problem with potentially different SA data in different nodes will probably not be very important since the cases when SA data could differ typically are dynamic combat situations when the positions of friendly and hostile aircraft change frequently. Obviously, in combat situations both sides will know the positions of the aircraft involved. Avoiding adversary detection by HDARP+ is most important in pre-combat situations, and in these cases SA data will generally be stable and consistent among the friendly aircraft.

If needed, the direction of the Tx beam from the directional antenna can be adjusted continuously while tracking the receiver. The use of SA data that include the position and velocity of all aircraft makes it possible to accurately predict the position of the aircraft, at least for a couple of seconds ahead. Therefore, the change of angle between aircraft due to their high (relative) speed will not be a problem in HDARP+.

In this paper we have discussed dynamic FEC coding to avoid adversary detection in networks with directional antennas. Since one of the benefits of directional antennas in airborne tactical networks is that adversary detection can be avoided, or at least delayed, it is probably most important to use the HDARP+ idea with dynamic FEC codes in networks with directional antennas. However, the idea could also be used in traditional networks with omnidirectional antennas. In that case, the distance factors f_i discussed in Section IV will be the distance from the sender to the closest adversary aircraft divided with the distance from the sender to the receiver.

FEC coding is one way of decreasing the BER. Lower order modulation [26] (resulting in lower bit rate) is another way of decreasing the BER. One of the benefits of using FEC coding (instead of lower order modulation or other alterations of the waveform) for compensating for a lower signal strength is that the solution can be implemented entirely in software. The only required interaction with the radio hardware is that the signal strength needs to be adapted depending on the distance factor f_i and the distance to the receiver; the waveform is not modified.

VII. Conclusion and Future Work

In Section III.C, research gap was identified related to the limited number of results regarding protocols that avoid adversary detection. The HDARP protocol was one important step in this direction, and HDARP+ presented here is another important step towards avoiding detection by adversaries. Compared to HDARP, HDARP+ increases the number of cases when detection can be avoided by novel and innovative use of dynamic FEC coding. Since we base our routing on SA data in HDARP+, both the sender and the receiver can, based on the position of hostile nodes, determine when different RS codes should be used. This means that we adapt the FEC code dynamically based on SA-data.

HDARP+ can be implemented in software, and no modification of the radio hardware is needed. Moreover, due to the relatively short length of the RS blocks no real-time processing is needed for encoding and decoding, since encoding and decoding can be done through table lookup.

In Section III.B we saw that dynamic FEC codes are mostly used in video streaming. No other study has used dynamic FEC codes for avoiding detection by adversaries.

The next step in this project will be to investigate how other FEC codes than RS codes could be used when avoiding adversary detection, i.e., we will evaluate such FEC codes in the same way as the RS codes were evaluated in Section V. The calculations in this paper are based on the assumption that there is Gaussian noise. One direction for future work is to look at how other models of the radio channel affect the performance of HDARP+. An additional next step in this project is to evaluate how the introduction of dynamic FEC coding in HDARP+ affects the time to adversary detection as well as network performance parameters such as average packet lifetime and average number of packets. These evaluations should be based on recorded real-world scenarios of tactical flight missions with simulated adversaries.

In parallel to the additional scientific evaluations and extensions discussed above, the technology readiness level (TRL) of HDARP+ will be increased by gradually providing implementations using real hardware. The goal is that HDARP+, or at least the idea of dynamic FEC coding to avoid detection, will be used in future generations of airborne tactical networks.

Acknowledgment

This work was funded in part by the Swedish Governmental Agency for Innovation Systems (VINNOVA) under Grant 2023-01949. The authors would like to thank Anders Höök at Saab for inspiration and valuable ideas.

References

[1] D. Ilie *et al*, "Avoiding Detection by Hostile Nodes in Airborne Tactical Networks," *Future Internet*, vol. 15, *(6)*, pp. 204, 2023.

[2] A. Zollinger, "Geographic Routing, " in: Kao, MY. (eds) Encyclopedia of Algorithms. Springer, Boston, MA. https://doi.org/10.1007/978-0-387-30162-4_164, 2008

[3] A. Goel and J. Gupta, "Novel approximation of Q-function using Gauss quadrature rule," *Sadhana (Bangalore)*, vol. 48, *(4)*, 2023.

[4] T. K. Moon, Error Correction Coding: Mathematical Methods and Algorithms. (2;Second; ed.) 2020.

[5] V. Guruswami and M. Sudan, "Improved decoding of reed-solomon and algebraic-geometric codes," in 1998, . DOI: 10.1109/SFCS.1998.743426.

[6] J. Wu *et al*, "Enabling Adaptive High-Frame-Rate Video Streaming in Mobile Cloud Gaming Applications," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, *(12)*, pp. 1988-2001, 2015.

[7] J. Xiao, T. Tillo, C. Lin and Y. Zhao, "Dynamic Sub-GOP Forward Error Correction Code for Real-Time Video Applications," in *IEEE Transactions on Multimedia*, vol. 14, no. 4, pp. 1298-1308, Aug. 2012, doi: 10.1109/TMM.2012.2194274.

[8] J. Wu *et al*, "Enabling Adaptive High-Frame-Rate Video Streaming in Mobile Cloud Gaming Applications," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, *(12)*, pp. 1988-2001, 2015.

[9] J. Wu *et al*, "Enabling Adaptive High-Frame-Rate Video Streaming in Mobile Cloud Gaming Applications," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, *(12)*, pp. 1988-2001, 2015.

[10] J. Wu, B. Cheng and M. Wang, "Adaptive Source-FEC Coding for Energy-Efficient Surveillance Video Over Wireless Networks," *IEEE Transactions on Communications*, vol. 66, *(5)*, pp. 2153-2168, 2018.

[11] J. Xiao, T. Tillo and Y. Zhao, "Real-Time Video Streaming Using Randomized Expanding Reed-Solomon Code," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 23, *(11),* pp. 1825-1836, 2013.

[12] Y. Weng *et al,* "Real-time video streaming using prediction-based forward error correction," *Computer Networks (Amsterdam, Netherlands : 1999),* vol. 92, pp. 134-147, 2015.

[13] R. Herrero and C. St-Pierre, "Dynamic forward error correction in wireless real-time Internet of Things networks," *IET Networks,* vol. 6, *(6),* pp. 218-223, 2017.

[14] O. S. Oubbati *et al*, "Routing in Flying Ad Hoc Networks: Survey, Constraints, and Future Challenge Perspectives," *IEEE Access,* vol. 7, pp. 81057-81105, 2019.

[15] B. Cheng *et al,* "Design considerations for next-generation airborne tactical networks," *IEEE Communications Magazine,* vol. 52, *(5),* pp. 138-145, 2014. . DOI: 10.1109/MCOM.2014.6815904.

[16] Y. Pan, N. Lyu and C. Yang, "GOR: Group-oblivious multicast routing in airborne tactical networks under uncertainty," *Journal of Network and Computer Applications,* vol. 207, pp. 103509, 2022.

[17] J. Miao, N. Lv, K. Chen, Q. Gao and X. Wang, "Dynamic Reliability-Aware Virtual Network Embedding for Airborne Tactical Networks," *Wireless Communications and Mobile Computing,* 2022.

[18] K. Chen, S. Zhao and N. Lv, "Network Monitoring Information Collection in the SDN-Enabled Airborne Tactical Network," *International Journal of Aerospace Engineering,* vol. 2018, pp. 1-20, 2018.

[19] X. Lu *et al,* "Minimizing Detection Probability Routing in Ad Hoc Networks Using Directional Antennas," *EURASIP Journal on Wireless Communications and Networking,* vol. 2009, *(1),* 2009.

[20] M. Singh and S. Kumar, "A Survey: Ad-hoc on Demand Distance Vector (AODV) Protocol," *International Journal of Computer Applications,* 2017.

[21] R. Sanchez-Iborra, M. Cano and J. Garcia-Haro, "Performance Evaluation of BATMAN Routing Protocol for VoIP Services: A QoE Perspective," *IEEE Transactions on Wireless Communications,* vol. 13, *(9),* pp. 4947-4958, 2014.

[22] C. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," in proceedings of the ACM SIGCOMM'94, London, UK, 1994. DOI: 10.1145/190314.190336.

[23] D. Zhang *et al*, "Novel dynamic source routing protocol (DSR) based on genetic algorithm-bacterial foraging optimization (GA-BFO)," *International Journal of Communication Systems*, vol.31, *(18)*, pp. n/a, 2018.

[24] H. Sinky and B. Hamdaoui, "Optimized link state routing for quality-of-service provisioning: implementation, measurement, and performance evaluation," *Wireless Communications and Mobile Computing*, vol. 14, *(10)*, pp. 937-948, 2014.

[25] R. C. Singleton, "Maximum Distance Q-Nary Codes," *IEEE Transactions on Information Theory,* vol. 10, *(2),* pp. 116-118, 1964.

[26] A. Goldsmith, *Wireless Communications.* Cambridge;New York;: Cambridge University Press, 2005.