Triangle Network Scheme

László Czap
EPFL, Switzerland
Email: laszlo.czap@epfl.ch

Vinod M. Prabhakaran
TIFR, India
Email: vinodmp@tifr.res.in

Suhas Diggavi
UCLA, USA
Email: suhas@ee.ucla.edu

Christina Fragouli
UCLA, USA, EPFL, Switzerland
Email: christina.fragouli@epfl.ch

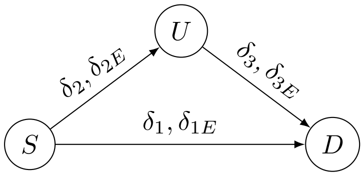

Figure 1. Triangle network

I. Main result

a) Model and definitions: We consider the network in Fig. 1 where a source S has a message W to send to a destination D, such that it remains secret from an eavesdropper Eve. The eavesdropper arbitrarily selects one of the three channels to wiretap.

All three channels are erasure channels with erasure probabilities δ_k and δ_{kE}, denoting the erasure probabilities toward the network node (U or D) and toward Eve (in case she is present on the given channel). All three channels are independent (e.g. operate in different frequency bands) and D can receive simultaneously over both $S-D$ and $U-D$.

The channel inputs are length L vectors of $\mathbb{F}_q$ symbols, which we call packets. To simplify notation, throughout the paper we express entropy and rate in terms of packets. We denote by $X_{k,i}$ the inputs of channel k in the ith transmission, while $Y_{k,i}$, $Z_{k,i}$ are the corresponding output at the network node and Eve respectively.

After each transmission, U and D causally send a public acknowledgment revealing the state of each channel, i.e. whether or not an erasure occurred ($\perp$ is the symbol of erasure). This feedback, after the ith transmission, is F_i and it is assumed to be publicly available to all network nodes as well as to Eve. Formally, we have that:

$$\begin{aligned}&\Pr\{Y_{1,i}, Y_{2,i}, Y_{3,i}, Z_{1,i}, Z_{2,i}, Z_{3,i} | X_{1,i}, X_{2,i}, X_{3,i}\}\\ &= \prod_{k=1}^{3} \Pr\{Y_{k,i}|X_{k,i}\} \Pr\{Z_{k,i}|X_{k,i}\}\end{aligned}$$

$$\Pr\{Y_{k,i}|X_{k,i}\} = \begin{cases} 1-\delta_k, & Y_{k,i} = X_{k,i} \\ \delta_k, & Y_{k,i} = \perp, \end{cases}, k \in \{1,2,3\}$$

$$\Pr\{Z_{k,i}|X_{k,i}\} = \begin{cases} 1-\delta_{kE}, & Z_{k,i} = X_{k,i} \\ \delta_{kE}, & Z_{k,i} = \perp, \end{cases}, k \in \{1,2,3\}$$

We assume that S and U can generate private randomness Θ_S, Θ_U of unlimited rate, independently of each other and from any other randomness in the system.

Message W consists of N packets. A secure communication scheme has parameters (N, ϵ, n) and satisfies the following reliability and security conditions:

Definition 1. *An (N, ϵ, n)–scheme has three sets of encoding functions $f_{k,i}$, $k \in \{1,2,3\}$ as well as a decoding map ϕ. The channel inputs are computed as*

$$\begin{aligned} X_{k,i} &= f_{k,i}(W, \Theta_S, F^{i-1}), \quad k \in \{1,2\} \\ X_{3,i} &= f_{3,i}(Y_2^{i-1}, F^{i-1}, \Theta_U). \end{aligned}$$

D can decode the message with high probability: $\Pr\{\phi(Y_1^n, Y_2^n) \neq W\} < \epsilon$. Furthermore, W remains secret from each eavesdropper:

$$I(W; Z_k^n) < \epsilon, \quad k \in \{1,2,3\}.$$

Definition 2. *A rate $R \in \mathbb{R}$ is securely achievable if for any $\epsilon > 0$ there exists a (N, ϵ, n)–scheme for which $R - \epsilon < \frac{1}{n}N$.*

In this paper, we characterize the secret message capacity of the triangle network, i.e. the largest securely achievable rate.

Theorem 1. *The secret message capacity of the triangle network is the solution of the following linear program (LP_1). All parameters are nonnegative: $m_i, k_i, c, c_i, r_i, R \geq 0$.*

$$\max R$$

$$\text{s.t.: } R \leq (1-\delta_1)m_1 + (1-\delta_3)m_3 \tag{1}$$

$$m_1(1-\delta_1)\frac{1-\delta_{1E}}{1-\delta_1\delta_{1E}} \leq (k_1+c_1)\delta_{1E}(1-\delta_1) + r_3 + c_3(1-\delta_3) \tag{2}$$

$$m_2(1-\delta_2)\frac{1-\delta_{2E}}{1-\delta_2\delta_{2E}} \leq k_2\delta_{2E}(1-\delta_2) + k_1(1-\delta_1) \tag{3}$$

$$\begin{aligned} m_3(1-\delta_3)\frac{1-\delta_{3E}}{1-\delta_3\delta_{3E}} &\leq (k_1+c_1)(1-\delta_1) + r_3\delta_{3E}\frac{1-\delta_3}{1-\delta_3\delta_{3E}} \\ &\quad + (k_3+c_3)\delta_{3E}(1-\delta_3) \end{aligned} \tag{4}$$

$$k_2(1-\delta_2) \geq c + r_3 \tag{5}$$

$$c \geq c_1(1-\delta_1\delta_{1E}) + c_3(1-\delta_3) \tag{6}$$

$$c \geq c_3(1-\delta_3\delta_{3E}) + c_1(1-\delta_1) \tag{7}$$

$$(1-\delta_3)m_3 \leq (1-\delta_2)m_2 + c_1(1-\delta_1) \tag{8}$$

$$k_1 + m_1 + c_1 \leq 1 \tag{9}$$

$$k_2 + m_2 \leq 1 \tag{10}$$

$$k_3 + m_3 + c_3 + \frac{r_3}{1-\delta_3} \leq 1 \tag{11}$$

The role of constraints (1)-(11) are explained in the next section. The matching outer bound and the detailed technical derivations are provided in [1].

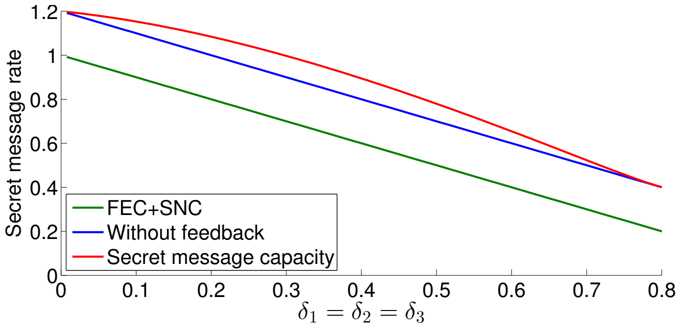

Figure 2. Comparison of secret message rates with/without exploiting erasures and with/without feedback. In all cases $\delta_{iE} = \delta_i + 0.2$.

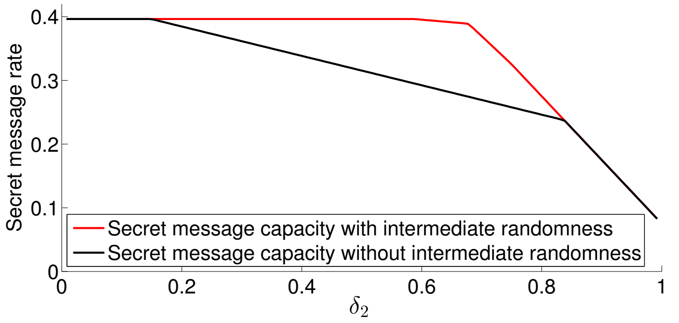

Figure 3. Secret message rates with/without randomness at U. $\delta_1 = \delta_{2E} = 0.8$, $\delta_{1E} = 0.5$, $\delta_3 = \delta_{3E} = 0.3$.

Solving the LP in Theorem 1 allows to evaluate 1) the benefit of exploiting erasures 2) the benefit of exploiting feedback 3) how much private randomness at the relay U can help. Fig. 2 compares four schemes: secret message capacity refers to our scheme in Theorem 1; we plot secret message capacity without feedback to show the benefits of exploiting erasures for secrecy yet without using feedback [2], [3]; and finally FEC+SNC refers to applying a link-by-link error correction coding (FEC) and then using the secure network coding scheme [4], [5].

It is clear that private randomness at the intermediate node can only help, but it is not obvious how significant the benefit is. Depending on the erasure probabilities, the benefit varies a lot. In some cases, there is no use of it at all (e.g. lossless channels), in other cases it can go up to more than 40 % gain in capacity. Fig. 3 gives a numerical example for illustration.

II. Background and previous work

In this section we summarize the former results that we build on when designing our scheme for the triangle network.

A. Principle of key generation

It was shown that the erasure channel can be utilized for generating a shared key between the sender and the receiver in the presence of an eavesdropping adversary [2], [6]–[8]. We use the following result:

Theorem 2. *Consider an erasure channel with state-feedback and with parameters δ, δ_E. If the source sends n i.i.d. uniform random packets, then secret key of rate*

$$\delta_E(1-\delta)$$

can be generated, while if the source sends i.i.d. uniform random packets using ARQ through n transmissions, then a secret key of rate

$$(1-\delta)\frac{\delta_E(1-\delta)}{1-\delta\delta_E}$$

can be generated. In both cases the resulting key K is uniformly distributed and is produced as linear combinations of packets that the destination receives. Further, for any $\epsilon > 0$

$$I(K; Z^n) < \epsilon \tag{12}$$

is satisfied if n is large enough.

Proof: Theorem 2 is a reformulation of Corollary 2 from [7] and Lemma 1 from [8]. ■

In other words, if a source sends nk all different key generation packets it can generate a key of size k' such that $\lim_{n\to\infty} \frac{1}{n}k' = k\delta_E(1-\delta)$. While if it sends nk packets using ARQ a key of size k' can be generated and $\lim_{n\to\infty} \frac{1}{n}k' = k\frac{\delta_E(1-\delta)}{1-\delta\delta_E}$. Having these asymptotic results in mind, in our description of the scheme we will assume that sending nk random key generation packets result $nk\delta_E(1-\delta)$ key packets, while in case of ARQ $nk(1-\delta)$ key generation packets result a key of size $nk\frac{\delta_E(1-\delta)}{1-\delta\delta_E}$ after nk transmissions.

B. Principle of encryption

Once a key is set up between the source and the destination, a message sending phase follows. Using the key, the message packets are first encrypted.

Theorem 3. *Consider an erasure channel with state-feedback and with parameters δ, δ_E. Assume the source and the destination have access to a uniform random key K of rate κ such that*

$$I(K; E|W) < \epsilon,$$

where W denotes the message to be sent and E denotes all the random variables the eavesdropper observes before starting transmissions. Then a message W of rate $\min\{1-\delta, \kappa\frac{1-\delta\delta_E}{1-\delta_E}\}$ can be transmitted such that for any $\epsilon > 0$ and a large enough n

$$I(W; Z^n, E) < \epsilon$$

holds and the destination can decode W with probability at least $1-\epsilon$. Further, this rate is achievable by a strategy where the source sends encrypted message packets W' using ARQ, such that

$$W' = W \oplus KG,$$

where G is the generator matrix of an MDS code.

Proof: Theorem 3 is a reformulation of Theorem 2 from [9]. ■

Using the above theorem, a message sending phase of length nm needs a rate $m(1-\delta)\frac{1-\delta_E}{1-\delta\delta_E}$ rate key and delivers a message of size m' such that $\lim_{n\to\infty}\frac{1}{n}m' = m(1-\delta)$. To ease the description of our scheme we will assume that a message sending phase of nm transmissions requires $nm(1-\delta)\frac{1-\delta_E}{1-\delta\delta_E}$ key packets and delivers securely $nm(1-\delta)$ encrypted message packets.

C. Secrecy over a point-to-point erasure channel

The secrecy capacity of a point-to-point erasure channel with state-feedback was characterized in [9]. A two-phase approach was introduced. In the first phase secret keys shared between the sender and the receiver are generated. These keys are used for encryption in the second phase, where the encrypted message is sent to the receiver. The second phase itself implements a capacity achieving strategy, namely ARQ, for reliable transmission of the encrypted packets. We refer to these phases as key generation and message sending phase respectively. It was shown that it is possible to securely send a message at rate R if a secret key of rate $R\frac{1-\delta_E}{1-\delta\delta_E}$ is available. A key generation rate $\delta_E(1-\delta)$ was shown to be achievable. In the key generation phase independently generated uniform random packets are transmitted. Given these two components, the key generation rate and the key requirement of the second phase, a straightforward calculation gives that the secrecy capacity of the erasure channel with state-feedback is

$$C_S = \delta_E(1-\delta)\frac{1-\delta\delta_E}{1-\delta\delta_E^2}. \tag{13}$$

We can rewrite this result in the form of a linear program, which will be helpful in the sequel. Let $k+m \le 1$ such that nk is the length of the key generation phase (expressed in number of transmissions) and nm is the length of the message sending phase of the scheme. Then the secrecy capacity is the value of the following linear program (LP_2):

$$\max R \ \text{ such that:}$$
$$R \le (1-\delta)m \tag{14}$$
$$m(1-\delta)\frac{1-\delta_E}{1-\delta\delta_E} \le k\delta_E(1-\delta) \tag{15}$$
$$1 \ge m+k. \tag{16}$$

Here, (14) expresses that the message rate cannot be higher than the message rate we can achieve in the second phase. Constraint (15) ensures the security of the message. The LHS of (15) is the secret key rate we need to secure a message of rate $(1-\delta)m$, while the RHS expresses the rate of secret key we can create with a length nk key generation phase. The last inequality ensures that the two phases use no more than n transmissions. In this case the solution of the linear program is trivial, and it might seem to be an unnecessarily complicated way of description, but at the same time it captures more explicitly the components the scheme is built of than the pure formula (13).

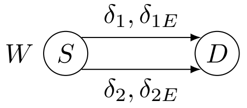

Figure 4. Two parallel channels

D. Secrecy over two parallel channels

Consider the setting displayed in Figure4 where there are two parallel independent erasure channels with erasure parameters δ_1, δ_{1E}, δ_2, δ_{2E}. We assume that there is one eavesdropper who might select any one of the channels to eavesdrop on. Equivalently we can consider two eavesdroppers, on on each channel, who do not cooperate. The secrecy capacity of this setting was characterized in [8]. The following linear program (LP_3) characterizes the secrecy capacity:

$$\max R \ \text{ such that:} \tag{17}$$
$$R \le (1-\delta_1)m_1 + (1-\delta_2)m_2 \tag{18}$$
$$m_1\frac{(1-\delta_{1E})(1-\delta_1)}{1-\delta_1\delta_{1E}} \le k_2(1-\delta_2) + k_1\delta_{1E}(1-\delta_1) \tag{19}$$
$$m_2\frac{(1-\delta_{2E})(1-\delta_2)}{1-\delta_2\delta_{2E}} \le k_1(1-\delta_1) + k_2\delta_{2E}(1-\delta_2) \tag{20}$$
$$1 \ge m_1 + k_1 \tag{21}$$
$$1 \ge m_2 + k_2. \tag{22}$$

This linear program follows the structure of LP_2. In (19)-(20) besides the key generation terms as seen for the point-to-point channel, terms $k_2(1-\delta_2)$ and $k_1(1-\delta_1)$ also appear. These terms capture the fact that key generation packets received through the second (first) channel can be used as secret keys on the other channel. Indeed, these packets remain secret from the eavesdropper who eavesdrops only the other channel.

It should be noted that the solution of this linear program is not trivial any more. Given (13), it is clear that if we knew that the eavesdropper eavesdrops on the first channel the secrecy capacity would be

$$(1-\delta_2) + \delta_{1E}(1-\delta_1)\frac{1-\delta_1\delta_{1E}}{1-\delta_1\delta_{1E}^2},$$

whereas if we knew that she selects the second channel it would be

$$(1-\delta_1) + \delta_{2E}(1-\delta_2)\frac{1-\delta_2\delta_{2E}}{1-\delta_2\delta_{2E}^2}.$$

One might expect that if her selection is not known we can possibly achieve

$$\min\left\{(1-\delta_2) + \delta_{1E}(1-\delta_1)\frac{1-\delta_1\delta_{1E}}{1-\delta_1\delta_{1E}^2}, (1-\delta_1) + \delta_{2E}(1-\delta_2)\frac{1-\delta_2\delta_{2E}}{1-\delta_2\delta_{2E}^2}\right\}. \tag{23}$$

The formula (23) gives a trivial upper bound, however – as was shown in [8] – it is not achievable in general. In some cases the solution of LP_3 is strictly smaller than (23).

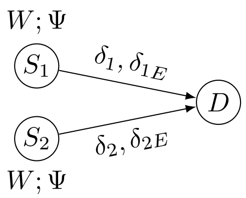

Figure 5. V-network

E. Secrecy in the V-network

In [8] the role of common randomness in secrecy capacity was investigated in the setting depicted in Figure 5. The two sources S_1 and S_2 are connected to a common destination D through independent erasure channels out of which any one is eavesdropped. S_1 and S_2 can generate unlimited amount of private randomness, but they have access to only a rate limited common random source Ψ. The rate of common randomness plays a crucial role in key generation and hence in the achievable secret message rate, since it limits the use of key generation packets sent through the other channel. For the optimal use of the common randomness new methods for key generation were introduced. The secrecy capacity is again characterized by a linear program (LP_4):

$$\max R \text{ such that:}$$

$$R \leq (1-\delta_1)m_1 + (1-\delta_2)m_2 \tag{24}$$

$$H(\Psi) \geq c + r_1 + r_2 \tag{25}$$

$$m_1\frac{(1-\delta_{1E})(1-\delta_1)}{1-\delta_1\delta_{1E}} \leq r_2 + r_1\frac{\delta_{1E}(1-\delta_1)}{1-\delta_1\delta_{1E}} + c_2(1-\delta_2) + (c_1+k_1)\delta_{1E}(1-\delta_1) \tag{26}$$

$$m_2\frac{(1-\delta_{2E})(1-\delta_2)}{1-\delta_2\delta_{2E}} \leq r_1 + r_2\frac{\delta_{2E}(1-\delta_2)}{1-\delta_2\delta_{2E}} + c_1(1-\delta_1) + (c_2+k_2)\delta_{2E}(1-\delta_2) \tag{27}$$

$$c \geq (1-\delta_1\delta_{1E})c_1 + (1-\delta_2)c_2 \tag{28}$$

$$c \geq (1-\delta_2\delta_{2E})c_2 + (1-\delta_1)c_1 \tag{29}$$

$$1 \geq k_1 + m_1 + c_1 + \frac{r_1}{1-\delta_1} \tag{30}$$

$$1 \geq k_2 + m_2 + c_2 + \frac{r_2}{1-\delta_2} \tag{31}$$

A brief summary of the different key generation techniques is as follows. The common randomness Ψ is divided into three independent parts: c, r_1, r_2. S_1 sends nr_1 independent random packets using ARQ. These packets contribute to the key of S_1 with rate $r_1\frac{\delta_{1E}(1-\delta_1)}{1-\delta_1\delta_{1E}}$. These packets are known by S_2, but not the eavesdropper on the second channel, so they also contribute to the key of S_2 with rate r_1. source S_2 uses the nr_2 packets from the common randomness is the same way.

From the remaining part c of the common randomness S_1 sends nc_1 packets while S_2 sends nc_2 packets. These packets are not necessarily independent, but they are always innovative for D and the eavesdropper (taken together). Constraints (28)-(29) ensure this property of these packets. These packets act like the key generation packets in the two parallel channel's case enabling a key rate $c_1\delta_{1E}(1-\delta_1) + c_2(1-\delta_2)$ for S_1 and $c_2\delta_{2E}(1-\delta_2) + c_1(1-\delta_1)$ for S_2.

The third kind of key generation packets are generated from private randomness. S_1 sends nk_1 of those and S_2 sends nk_2. These packets contribute to only to the key of the given source as seen in the case of a point-to-point erasure channel.

III. Triangle network

In this paper we consider the setting in Figure 1.

There are three nodes in this network: a source S, a destination D and an intermediate node U. All the three channels are independent erasure channels with parameters δ_i, δ_{iE}. We assume that state-feedback from each channel is publicly available. We consider the case where there is one eavesdropper Eve in the network, who arbitrarily selects one of the three channels to eavesdrop on. Or equivalently we can think of three noncolluding eavesdroppers, one on each link.

Our scheme builds on the techniques developed for the network in Section II-E with sources accessing limited rate common randomness. S and U can be considered as the two sources, however, there are two key differences:

- There is no common random source that S and U shares, what they have in common has to be transmitted by S through the $S-U$ channel. New randomness, which is independent from the message arrives to U during the key generation phase that takes place on the $S-U$ channel. So the rate of common randomness is limited by the length of key generation on the $S-U$ channel. Despite of the source common randomness, the same key generation techniques are applicable.
- U does not have direct access to the message W, so even if it had a perfect channel ($\delta_3 = 0, \delta_{3E} = 1$) it might not be able to utilize all transmissions for the message sending phase.

Beside the known techniques for key generation we also utilize new algorithms to process packets at the intermediate node. We give intuition in the following subsections.

A. Recombination of encrypted packets

The key generation packets received through the direct $S-D$ channel can be used on the $S-U$ channel as keys. Notice, that we do not require U to be able to decode the message packets that it receives in the message sending phase that takes place on the $S-U$ channel. Although the key generation packets received by D through the $S-D$ channel do not form a shared key between S and U, they can be used as keys for encryption the $S-U$ channel.

The message packets that U receives are already encrypted, thus U can utilize the random components in these packets against the eavesdropper on the $U-D$ channel. Still, D has to be able to decrypt the packets it receives, so U first needs to remove those packets from the linear combinations that only S and U knows. These are keys generated on the $S-U$ channel. After that, U needs to produce linear combinations such that the remaining random components are sufficient to secure the resulting message packets. The resulting packets can be sent without using additional keys. After all it is the key generation packets that D receives through the $S-D$ channel that secure these packets, hence U can secure as many message packets using this technique as if U had direct access to both the key packets and the message packets. We give details and formal description about how to compute these linear combinations later.

B. Keys used as message packets

In this section we make the observation that certain key packets can be treated as if they were encrypted message packets. One should notice that using a one-time-pad encryption the interpretation of packets as keys or as encrypted packets is arbitrary. Consider the following example. Let K be a secret key and let $K+W$ be the encrypted message. We can equally say that let $K' = K + W$ be the shared secret and then $K = K' + W$ is interpreted as the encrypted message.

The possibility of different interpretation of packets leads to a nontrivial observation: the number of different message packets that U can send in the message sending phase on the $U-D$ channel is not restricted by the number of packets U receives in the message sending phase on the $S-U$ channel. Consider the following scenario. Assume S and U can both generate a random packet C, which is not yet known to D. If S sends $C+W$ to D, while U sends C, with a different interpretation of the packets we can equally say that U sends $C'+W$, where $C' = C + W$. Hence, although U does not know W, it can send an encrypted packet of the form $C'+W$.

The above observation is counter intuitive for our usual flow-based interpretation of network traffic. For D it is not always possible to tell through which path a certain message packet has arrived, because it depends on the interpretation of the packet. This gives us some flexibility and it overrules the common sense that it should not be possible to send more message packets on the $U-D$ channel than what was received by U in the message sending phase on the $S-U$ channel.

This reveals that in some cases the two phase interpretation of the scheme leaves a choice on where we separate the phases. We follow the convention that we call a key generation packet that appears as a random packet to the receiver upon reception and call an encrypted message packet that enables immediately the decryption of a message packet or a linear combination of message packets with the help of previously received packets.

IV. Scheme

In this section we show the direct part of Theorem 1. We need to prove that whenever the above linear program is feasible there exists a scheme that achieves rate R.

A. Key generation phase

1) $S-U$ channel: S sends nk_2 i.i.d. uniform random packets.

2) $S-D$ channel: S first sends nk_1 i.i.d. uniform random packets. From the $k_2(1-\delta_2)$ packets that U receives in the key generation phase two disjoint set of nc and nr_3 packets C and R_3 are selected. Further, let G be an $nc \times n(c_1+c_3)$ matrix such that G is the generator of an MDS code. Then compute

$$CG = \begin{bmatrix} C_1 & C_3 \end{bmatrix},$$

where C_1 is a matrix of nc_1 packets and C_3 is a matrix of nc_3 packets. S sends the nc_1 packets from C_1 XOR-ed with a message packet. All the nc_1 such transmissions use a different packet from C_1, but the same message packet is used again in the next transmission to form the XOR-ed packet in case D does not receive a transmission.

3) $U-D$ channel: U first sends nk_3 i.i.d. uniform random packets. Then, U sends the nc_3 packets from matrix C_3. Following this U sends the nr_3 packets in R_3 using ARQ.

B. Key rates

Using Theorem 2 we can calculate the key rates these key generation strategies allow. We have to note that the key generation packets received by U give rise to a common randomness of rate $k_2(1-\delta_2)$ between S and U. Packets C_1, C_3 and R_3 generated from this common randomness can be used as if they were i.i.d. uniform random key generation packets. This follows from the MDS property of G as well as from constraints (5)-(7). This property was shown by Lemma 3 in [8].

1) $S-U$ channel: The nk_2 key generation packets allow a key rate $k_2\delta_{2E}(1-\delta_2)$. Beside this, the nk_1 key generation packets sent through the $S-D$ channel can be used for encryption resulting an overall key rate

$$k_2\delta_{2E}(1-\delta_2) + k_1(1-\delta_1). \tag{32}$$

2) $S-D$ channel: From the $S-D$ channel's perspective there is no difference between the nk_1 i.i.d. random packets and the nc_1 packets formed by XOR-ing packets from C_1 and W. Indeed, these packets are i.i.d. random packets and they are independent of the message packets that are to be sent in the message sending phase of this channel. This property is ensured by constraint (6). Beside these keys S can also use packets that D receives from U. There are $nr_3 + nc_3(1-\delta_3)$ such packets that S can also generate. This results an overall key rate

$$(k_1+c_1)\delta_{1E}(1-\delta_1) + r_3 + c_3(1-\delta_3). \tag{33}$$

3) $U-D$ channel: From the $U-D$ channel's perspective there is no difference between the nk_3 private random packets and the nc_3 packets generated from the common randomness between S and U. Beside these we take into account the nr_3 packets sent using ARQ which provides access to a secret key at rate

$$(k_3+c_3)\delta_{3E}(1-\delta_3) + r_3\frac{\delta_{3E}(1-\delta_3)}{1-\delta_3\delta_{3E}} \tag{34}$$

C. Encryption and message sending phase

The message packets are split into parts as follows. S considers the $nc_1(1-\delta)$ message packets that are sent XOR-ed with the packets from C_1 already delivered. The rest of the message is divided into two parts: $nm_1(1-\delta)$ packets W_1 to be sent through the $S-D$ channel and $nm_2(1-\delta_2)$ packets W_2 to be sent through the $S-U$ channel.

1) $S-U$ channel: The encryption and message sending phase is straightforward on the $S-U$ channel. Let K_2 denote the matrix formed of the $nk_1(1-\delta_1)$ received by D through the $S-D$ channel in the first step of the key generation (denoted by $K_2^{(1)}$), together with the $nk_2\delta_{2E}(1-\delta_2)$ keys generated on the $S-U$ channel (denoted by $K_2^{(2)}$). Then, the encrypted packets W_2' are calculated as

$$W_2' = W_2 \oplus K_2 G_2 = W_2 \oplus \begin{bmatrix} K_2^{(1)} & K_2^{(2)} \end{bmatrix} \begin{bmatrix} G_2^{(1)} \\ G_2^{(2)} \end{bmatrix} \qquad (35)$$

where G_2 is a $n(k_1(1-\delta_1)+k_2\delta_{2E}(1-\delta_2)) \times nm_2(1-\delta_2)$ matrix and is a generator of an MDS code. In notation we distinguish the first $nk_1(1-\delta_1)$ rows of G_2 and the last $nk_2\delta_{2E}(1-\delta_2)$ rows of it by $G_2^{(1)}, G_2^{(2)}$. The encrypted packets are then sent using ARQ.

2) $S-D$ channel: S forms a key K_1 according to (33) and encrypts the packets W_1 as

$$W_1' = W_1 \oplus K_1 G_1, \qquad (36)$$

where G_1 is a $n(k_1\delta_{1E}(1-\delta_1)+c_3(1-\delta_3)+r3) \times nm_1(1-\delta_1)$ matrix and is a generator of an MDS code. The encrypted packets are then sent using ARQ.

3) $U-D$ channel: We need to define the operations that U performs on the packets it receives. U sends three set of packets interpreted as encrypted message packets. It first calculates

$$W_2'' = W_2' \oplus K_2^{(2)} G_2^{(2)} = W_2 \oplus K_1^{(1)} G_2^{(1)}, \qquad (37)$$

The resulting packets W_2'' are linear combinations of the message W and the key generation packets from the $S-D$ channel. U computes

$$\begin{bmatrix} W_{3a}' & W_{3b}' \end{bmatrix} = W_2'' G_3 = W_2'' \begin{bmatrix} G_{3a} & G_{3b} \end{bmatrix}, \qquad (38)$$

where G_3 is an $nm_2(1-\delta_2) \times nm_2(1-\delta_2)$ invertible matrix such that G_{3a} is of size $nm_2(1-\delta_2) \times \min\{nk_1(1-\delta_1)\frac{1-\delta_3\delta_{3E}}{1-\delta_{3E}}, nm_2(1-\delta_2)\}$ such that $G_2^{(1)}G_{3a}$ is the generator of an MDS code. Encrypted packets W_{3a}' are then sent using ARQ. The remaining W_{3b}' packets are considered as unencrypted message packets to be sent after further encryption.

The second set of packets U sends are the $nc_1(1-\delta_1)$ packets from C_1 that were received by D XOR-ed with a message packet. These packets are sent using ARQ. These packets enable D to decode $nc_1(1-\delta_1)$ message packets. Also, for the eavesdropper on the $U-D$ channel these are random packets independent from the message, thus these packets allow the generation of further $nc_1(1-\delta_1)\frac{\delta_{3E}(1-\delta_3)}{1-\delta_3\delta_{3E}}$ key packets to be used by U. Thus besides the keys from the key generation phase (34), U can use these additional secret keys to from its key K_3.

U uses K_3 to encrypt the remaining part of the message W_{3b}', which results the third set of message packets U sends:

$$W_{3b}'' = W_{3b}' \oplus K_3 G_3', \qquad (39)$$

where G_3' is a $|K_3| \times (nm_2(1-\delta_2) - nk_1(1-\delta_1)\frac{1-\delta_3\delta_{3E}}{1-\delta_{3E}})^+$ matrix and is a generator of an MDS code. Packets W_{3b}'' are then sent using ARQ.

D. Analysis

1) Rate: Let

$$nm_3 = \frac{nc_1(1-\delta_1)}{1-\delta_3} + \frac{nm_2(1-\delta_2)}{1-\delta_3}, \qquad (40)$$

i.e. the number of transmissions that U uses in the message sending phase. Every received packet allows D to decode a message packet. Besides, D receives $nm_1(1-\delta_1)$ packets through the $S-D$ channel. Clearly, a message rate $(1-\delta_1)m_1 + (1-\delta_3)m_3$ is achievable as long as $m_3(1-\delta_3) \leq c_1(1-\delta_1) + m_2(1-\delta_2)$, which is ensured by (8).

Constraints (9)-(11) ensure that scheme described above is feasible, i.e. no more than n transmissions are used on each channel.

2) Security: We need to see if a sufficient key rate is available against all eavesdroppers whenever we send encrypted packets. The security of the scheme then follows from Theorem 3.

a) $S-U$ channel: It is clear from (3) that the key rate (32) available on this channel is sufficient to secure a message sending phase of length nm_2.

b) $S-D$ channel: In the same way (2) ensures that the key rate (33) is sufficient to secure the message sending phase of length nm_1.

c) $U-D$ channel: Packets W_{3a}' are of the form

$$W_{3a}' = W_2 G_{3a} \oplus K_1^{(1)} G_2^{(1)} G_{3a}. \qquad (41)$$

We see the same form of encryption as in Theorem 3, applied on the linear combination $W_2 G_{3a}$ as message packets and matrix $G_2^{(1)}G_{3a}$ for combining the keys $K_1^{(1)}$. The key rate of $K_1^{(1)}$ is $k_1(1-\delta_1)$. While the rate of W_{3a}' is

$$k_1(1-\delta_1)\frac{1-\delta_3\delta_{3E}}{1-\delta_{3E}}, \qquad (42)$$

hence the rate of $K_1^{(1)}$ is sufficient to secure this message rate by Theorem 3.

The second set of packets (a subset of C_1) are random packets that are independent of the message, thus no encryption is required and they cannot reveal any information to Eve about the message.

Consider the message packets W_{3b}'. The rate of W_{3b}' is $(m_2(1-\delta_2) - k_1(1-\delta_1)\frac{1-\delta_3\delta_{3E}}{1-\delta_{3E}})^+$, while $|K_3|$ has rate $(k_3+c_3)\delta_{3E}(1-\delta_3) + (r_3 + c_1(1-\delta_1))\frac{\delta_{3E}(1-\delta_3)}{1-\delta_3\delta_{3E}}$. Hence, for security we need that

$$m_2(1-\delta_2) - k_1(1-\delta_1)\frac{1-\delta_3\delta_{3E}}{1-\delta_{3E}} \leq$$
$$(k_3+c_3)\delta_{3E}(1-\delta_3)\frac{1-\delta_3\delta_{3E}}{1-\delta_{3E}} + (r_3+c_1(1-\delta_1))\frac{\delta_{3E}(1-\delta_3)}{1-\delta_{3E}} \qquad (43)$$

Using (40) we get:

$$m_3(1-\delta_3) - c_1(1-\delta_1) - k_1(1-\delta_1)\frac{1-\delta_3\delta_{3E}}{1-\delta_{3E}} \leq \\ (k_3+c_3)\delta_{3E}(1-\delta_3)\frac{1-\delta_3\delta_{3E}}{1-\delta_{3E}} + (r_3+c_1(1-\delta_1))\frac{\delta_{3E}(1-\delta_3)}{1-\delta_{3E}} \tag{44}$$

After rearranging terms this condition becomes constraint (4), hence the security of message packets W'_{3b} is ensured by the feasibility of LP_1.

This concludes the proof of the direct part of Theorem 1. We have seen that the scheme is feasible, it achieves the claimed rate and it ensures security against each eavesdropper.

References

[1] [Online]. Available: http://arni.epfl.ch/ czap/triangle.html

[2] A. D. Wyner, “The wire-tap channel,” *The Bell system Technical Journal*, vol. 54, no. 8, pp. 1355–1387, 1975.

[3] T. Cui, “Coding for wireless broadcast and network secrecy,” Ph.D. dissertation, California Institute of Technology, 2010.

[4] N. Cai and R. Yeung, “Secure network coding,” in *International Symposium on Information Theory (ISIT)*. IEEE, 2005, p. 323.

[5] ——, “Secure network coding on a wiretap network,” *IEEE Transactions on Information Theory,*, vol. 57, no. 1, pp. 424–435, 2011.

[6] U. Maurer, “Secret key agreement by public discussion from common information,” *IEEE Transactions on Information Theory*, vol. 39, no. 3, pp. 733–742, May 1993.

[7] M. Jafari Siavoshani, S. Diggavi, C. Fragouli, U. K. Pulleti, and K. Argyraki, “Group Secret Key Generation over Broadcast Erasure Channels,” in *Asilomar Conference on Signals, Systems, and Computers*, 2010, pp. 719–723.

[8] L. Czap, V. Prabhakaran, S. Diggavi, and C. Fragouli, “Exploiting common randomness: a resource for network secrecy,” in *Information Theory Workshop (ITW)*, 2013.

[9] L. Czap, V. Prabhakaran, C. Fragouli, and S. Diggavi, “Secret message capacity of erasure broadcast channels with feedback,” in *Information Theory Workshop (ITW)*, 2011, pp. 65–69.