ISIT'2013 1569716387

Simultaneously Generating Multiple Keys in Many to One Networks

Lifeng Lai
Department of Electrical & Computer Engineering
Worcester Polytechnic Institute
Email: llai@wpi.edu

Lauren Huie
Air Force Research Lab
Information Directorate
Email: Lauren.Huie@rl.af.mil

***Abstract*—The problem of simultaneously establishing multiple keys, one for each user in a set of users, is considered with possible assist from a group of dedicated helpers. For the case in which all users are required to generate keys, we develop a scheme that is sum rate optimal. For the case with dedicated helpers, we develop an achievable scheme and derive an outer bound. We identify conditions under which the developed scheme achieves the full capacity region and conditions under which it is sum rate optimal. We then specialize the study to a pairwise independent network model, for which we convert the key generation problem to a single-source multi-commodity flow over a network problem. Coupling results from graph theory, we fully characterize the capacity region for the general case of generating multiple keys with multiple helpers under the PIN model.**

I. Introduction

Establishing secret keys to be shared by the sender and receiver in secure communications while keeping the keys secret from possible attackers is very challenging. [1], [2] introduced a key agreement approach in which legitimate users establish a secret key between them by exchanging information in public only. This line of work has been extended to models with additional helpers [3] and its specialization to pairwise independent network (PIN) model [4], and joint source-channel models [5], [6].

Until now, with a few exceptions [7], [8], most of the existing work along this line focuses on generating a *single* key to be shared by a pair [1], [2] or a group of users [3]. In practice, however, there are various scenarios that require us to generate *multiple* keys to be shared by different pairs of users. For example, in cellular systems, there are typically many wireless users, each of which needs to establish a key with the base-station to be used to protect the information exchanged between the wireless user and the base-station. In this paper, we focus on such a scenario and consider the problem of establishing multiple keys for multiple users, each to be shared by one user and the base station, under the source model.

A naive approach for this multiple-key establishment in the many to one network problem is to decompose this problem into multiple independent single-key generation problems. There are several issues with this approach. First, typically, the random sources at each user are correlated. Hence, the public discussion of user i might leak information about the key established by user j. Hence, if one treats these problems separately, the keys may no longer be secure. Second, as shown in [3], [9], user cooperation can bring tremendous gains for the key generation. Hence, these multiple users can cooperate with each other to increase the key rate. As a result, this naive approach might suffer performance losses.

In this paper, we rigorously formulate and provide solutions for the simultaneous multiple key generation problem. We first consider the scenario in which all the users are required to generate keys. We propose a scheme that is a careful concatenation of multiple modified single key generation problems. We show that the proposed scheme is sum rate optimal. Here, we comment on the difference between this scenario and the existing work [7], [8] that considered the generation of two keys. In [7], [8], the key of each user needs to be secured from the other user. In our case, we do not impose this condition. Our setup is useful when the users in a network cooperate with each other in generating multiple keys, in the same sprit as the existing work on generating a single key with helpers [3], [9] such that the key is not required to be hidden from the helpers[1].

We then study the scenario in which there are several dedicated helpers whose sole goal is to assist the key generation process for other users. The achievable scheme developed in the previous scenario can be extended to this scenario with proper modifications. We also develop a general upper bound by extending techniques developed in [3]. We further identify cases under which the performance of the proposed scheme matches the outer bound, and conditions under which the proposed scheme is sum rate optimal.

We also specialize the study to the PIN model [4], [10], [11]. In the PIN model, the correlation between the random variables possess a certain structure, which arises naturally in generating keys using wireless fading gains. Exploiting this structure, we construct a graph based key generation approach. Our approach effectively converts the key generation problem to a single-source multi-commodity flow over a network problem. Leveraging results from graph theory [12], we show that this graph based approach achieves the full capacity region for the general case of establishing multiple keys with the aid

The work of L. Lai was supported by the NSF CAREER Award under Grant CCF-1318980, NSF under Grant CNS-1321223 and Air Force Research Lab, Information Directorate.

[1]We note that [3] also considers the case in which the key is required to be hidden from the helpers. The extension to this setup along the line of [7], [8] is important and interesting. However, we note that the extension along this line is a generalization of the problem of key generation with side-information at Eve, which itself is an open problem [1], [2]. Hence, the extension along this line will be challenging.

of multiple helpers.

The reminder of the paper is organized as follows. In Section II, we describe the model. In Section III, we study the case in which all users are required to generate keys. In Section IV, we present our results for the network with dedicated helpers. In Section V, we specialize the study to the PIN model. In Section VI, we offer conclusion remarks. Due to space limitations, we provide only outlines of the proofs. Details can be found in [13].

II. Model

The model considered in this paper is shown in Figure 1. In particular, we consider a scenario with $J+1$ users, indexed by $j \in \mathcal{J} \triangleq \{0, \cdots, J\}$. Each user $j \in \mathcal{M} \triangleq \{1, \cdots, M\}$ needs to establish a key K_j with user 0. Users indexed by $j \in \{M+1, \cdots, J\}$ act as dedicated helpers whose only purpose is to help the key generation process of the users in $\mathcal{M}$. Hence, in our problem, we need to generate M keys with $J-M$ dedicated helpers. One can think of user 0 as the base station in the cellular system. Each user observes a random sequence X_j^n taken from a set $\mathcal{X}_j$ with a finite alphabet size $|\mathcal{X}_j|$. The observations are correlated among the users but are independent and identically distributed (i.i.d.) over time, hence

$$P_{X_0^n,\cdots,X_J^n}(x_0^n, \cdots, x_J^n) = \prod_{t=1}^{n} P_{X_0,\cdots,X_J}(x_0(t), \cdots, x_J(t)).$$

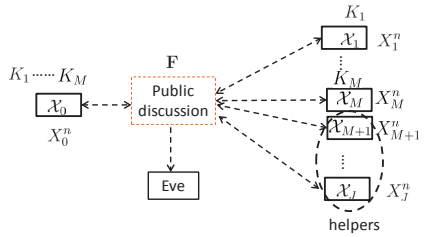

Fig. 1. System model.

These users are allowed to exchange information with each other in public. Without loss of generality, one can assume that these users take turns in sending public information for r rounds. We use $f_1, \cdots, f_{r(J+1)}$ to denote the public information exchanged. Here, f_t is the information sent by user $i = t \mod (J+1)$ at time t. f_t depends on the random sequences X_i^n at user i and the public discussion $f_1, \cdots, f_{t-1}$ that has occurred so far. We use $\mathbf{F}_j$ to denote the collection of public discussion sent by j during the public discussion phase, and $\mathbf{F} = [\mathbf{F}_0, \cdots, \mathbf{F}_J]$ to denote the whole set of public discussion. Eve knows the functions used at each user for generating the public information, and knows $\mathbf{F}$ perfectly.

After the public discussion is finished, by combining the public information $\mathbf{F}$ and its random sequence X_j^n, user $j \in \mathcal{M}$ generates a key K_j^* using a function g_j, that is $K_j^* = g_j(\mathbf{F}, X_j^n)$. Correspondingly, user 0 also generates a key K_j^0, using X_0^n and $\mathbf{F}$. We require that for each j,

$$\Pr\{K_j^* \neq K_j^0\} \leq \epsilon, \tag{1}$$

This condition implies that the key generated at user j and 0 are the same with a high probability. In the following, we will use K_j to denote this common key shared by user j and 0. In total, we will generate M keys $K_1, \cdots, K_M$, one for each user $j \in \mathcal{M}$ to be shared with user 0. In addition, we require

$$\frac{1}{n} I(K_1, \cdots, K_M; \mathbf{F}) \leq \epsilon. \tag{2}$$

This condition implies that Eve gains negligible amount of information about the keys $K_1, \cdots, K_M$. In addition to (1) and (2), we further require that these M keys to be uniformly distributed and independent of each other.

We say that a rate vector $(R_1, \cdots, R_M)$ is achievable, if there exists a public discussion scheme such that the above mentioned conditions are satisfied and

$$\frac{1}{n} H(K_j) \geq R_j - \epsilon. \tag{3}$$

The set of all achievable key rate vectors is called the key-rate region $\mathcal{C}$. We will also be interested in the sum of these key rates, and refer to the largest possible sum rate as the sum key rate capacity:

$$C_{sum} = \sup_{(R_1,\cdots,R_M)\in\mathcal{C}} \sum_{j=1}^{M} R_j. \tag{4}$$

III. No dedicated helpers

We first start with the case in which all users in the network are required to generate keys, i.e., there are no dedicated helpers hence $M = J$. In this section, we focus on characterizing the maximum sum key rate C_{sum}.

Theorem 1: The sum key rate capacity for generating J keys without dedicated helpers is

$$C_{sum} = I(X_0; X_1, \cdots, X_J). \tag{5}$$

Proof: (Outline) The rate in (5) is the secret key capacity when we allow all nodes except 0 to pool their observations together. This obviously serves as an upperbound for our setup in which all observations are distributed. In the following, we provide a sketch of our scheme that achieves this sum rate. Our scheme is a careful concatenation of multiple rounds of modified single key generation problem. Without loss of generality, we start with user J. At first, user J and user 0 generate the key K_J with a rate $I(X_0; X_J)$ using the correlation (X_J^n, X_0^n) and ignoring all other users. This is a single key generation problem [1], [2]. Let $\mathbf{F}_J$ be the public discussion sent by user J at this step. The existing single key generation scheme can guarantee that $I(K_J; \mathbf{F}_J) \leq \epsilon$. In addition, by the end of this step, user 0 can recover X_J^n. At the second step, user $J-1$ and user 0 generate the key K_{J-1} with rate $I(X_0; X_{J-1}|X_J)$ by treating user J as an Eve. This is a modified single key generation problem in the sense that Eve (user J) has additional side-information (X_J^n) and one of the legitimate users (user 0) has Eve's side-information. This step is related to but different from two scenarios that have been studied in the literature: that of key generation with side information at Eve [1], in which Eve has additional correlated observations, and that of key generation with a helping Eve [1], [9], in which Eve reveals its observations to both users. The difference between this step and the standard key generation with side-information at Eve is that, in our

case one of the legitimate users (user 0) knows the side-information at Eve. The difference between this step and the key generation with a helping Eve is that in our case only user 0 knows the observation at Eve, while in the key generation with a helping Eve, all users involved know the observation at Eve. It can be shown that there exists $\mathbf{F}_{J-1}$ (the public information sent by user $J-1$) and $K_{J-1} = g_{X_{J-1}}(X_{J-1}^n)$ such that $I(K_{J-1}; \mathbf{F}_{J-1}, X_J^n) \leq n\epsilon$. Furthermore, by the end of this step, user 0 is able to recover X_{J-1}^n. At the third step, user $J-2$ and user 0 generate the key K_{J-2} with rate $I(X_0; X_{J-2}|X_{J-1}, X_J)$ by treating users $J-1$ and J as colluding Eves. Here, by colluding Eves, we mean that user $J-1$ and J creates a super-Eve with side-information (X_{J-1}^n, X_J^n). Again, this is a modified single key generation problem in the sense that Eve (super Eve created from users $J-1$ and J) has additional side-information (X_{J-1}^n, X_J^n) and one of the legitimate users (user 0) has Eve's side-information. It can be shown that there exists $\mathbf{F}_{J-2}$ (the public information sent by user $J-2$) and $K_{J-2} = g_{X_{J-2}}(X_{J-2}^n)$ such that $I(K_{J-2}; \mathbf{F}_{J-2}, X_{J-1}^n, X_J^n) \leq n\epsilon$. In addition, by the end of this step, user 0 will be able to recover X_{J-2}^n. This process continues until we reach user 1. In summary,

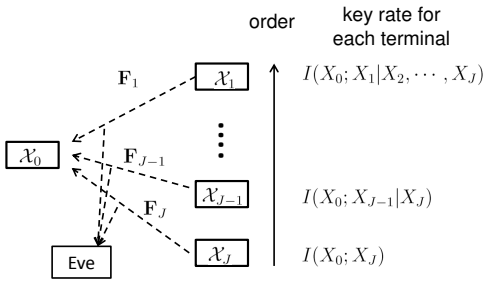

Fig. 2. An outline of the key generation scheme.

each user takes turns in establishing keys with user 0 while treating all users that have finished the key establishment steps as colluding Eves with side-information. Each step is a modified single-key generation problem such that Eve has additional side-information, which is also available only at user 0. Furthermore, the amount of side-information increases as the process progresses. While it is relatively simple to guarantee that the public discussion in each step does not leak information about the key generated at that particular step, it does not automatically guarantee that this public discussion will not leak information about keys generated at other steps. Furthermore, the requirement in our system model is more strict. We require that the collection of all public discussions do not leak any information about all the keys. Our proof shows our scheme satisfies the security requirement (2). Details can be found in [13]. ■

Remark 2: We note that in the proof, the order of the process is from user J to 1. It is easy to see that any order will allow us to achieve the sum-rate capacity. The convex hull of these $J!$ tuples, one for each possible order and each of them lies on the outer-bound, with $(0, \cdots, 0)$ consists of an inner bound of the whole capacity region.

IV. Helper Nodes

In this section, we discuss the case with dedicated helpers. Here, we note that in our model it is not required that the generated keys be kept secure from these helper nodes. In this section, we state the results for the case of $M = 2$ and $J = 3$ in detail. That is, in addition to users 0, 1 and 2, which are required to generate keys, there is another user that can help the establishment of key K_1 between user 0 and user 1, and key K_2 between user 0 and user 2. The results developed in this section can be generalized to any values of M and J, but the form will be complicated[2].

Theorem 3: The key capacity region of generating two keys with a dedicated helper is inner-bounded by the convex hull of the point $(0, 0)$ and the following four pairs of (R_1, R_2):

$$P_1 = \left(0, \min_{\{B \subseteq \mathcal{J}: 0 \in B, 2 \in B^c\}} I(X_B; X_{B^c})\right), \quad (6)$$

$$P_2 = (I(X_0; X_1|X_2, X_3), I(X_0; X_2|X_3) + \min\{I(X_0; X_3), I(X_2; X_3)\}), \quad (7)$$

$$P_3 = (I(X_0; X_1|X_3) + \min\{I(X_0; X_3), I(X_1; X_3)\}, I(X_0; X_2|X_1, X_3)), \quad (8)$$

$$P_4 = \left(\min_{\{B \subseteq \mathcal{J}: 0 \in B, 1 \in B^c\}} I(X_B; X_{B^c}), 0\right). \quad (9)$$

The following theorem provides an outer bound for the one helper scenario.

Theorem 4: The secret key rate region for generating two keys with a dedicated helper is outer bounded by the following region:

$$\begin{aligned}
R_1 &\leq \min_{\{B \subseteq \mathcal{J}: 0 \in B, 1 \in B^c\}} I(X_B; X_{B^c}), \\
R_2 &\leq \min_{\{B \subseteq \mathcal{J}: 0 \in B, 2 \in B^c\}} I(X_B; X_{B^c}), \\
R_1 + R_2 &\leq \min\{I(X_0; X_{\{0\}^c}), I(X_{\{0,3\}}; X_{\{1,2\}})\}, \quad (10) \\
R_1 + 2R_2 &\leq 2H(X_{\mathcal{J}}) - \max\{H(X_0|X_{\{0\}^c}) \\
&\quad + H(X_{\{1,2\}}|X_{\{0,3\}}) + H(X_{\{2,3\}}|X_{\{0,1\}}) + H(X_{\{2\}^c}|X_2)), \\
&\quad H(X_{\{0\}^c}|X_0) + H(X_{\{0,3\}}|X_{\{1,2\}}) \\
&\quad + H(X_{\{0,1\}}|X_{\{2,3\}}) + H(X_2|X_{\{2\}^c}))\} \\
2R_1 + R_2 &\leq 2H(X_{\mathcal{J}}) - \max\{H(X_0|X_{\{0\}^c}) \\
&\quad + H(X_{\{1,2\}}|X_{\{0,3\}}) + H(X_{\{1,3\}}|X_{\{0,2\}}) + H(X_{\{1\}^c}|X_1)), \\
&\quad H(X_{\{0\}^c}|X_0) + H(X_{\{0,3\}}|X_{\{1,2\}}) \\
&\quad + H(X_{\{0,2\}}|X_{\{1,3\}}) + H(X_1|X_{\{1\}^c}))\}.
\end{aligned}$$

As a consequence of this Theorem, we have the following tight result that shows our scheme achieves the full capacity under certain conditions.

Lemma 5: If

$$\begin{aligned}
I(X_0; X_1X_2X_3) &= \min_{\{B \subseteq \mathcal{J}: 0 \in B, 1 \in B^c\}} I(X_B; X_{B^c}) \\
&= \min_{\{B \subseteq \mathcal{J}: 0 \in B, 2 \in B^c\}} I(X_B; X_{B^c}), \quad (11)
\end{aligned}$$

[2]In the next section, we will address this general case under the PIN model. Under the PIN model, we show that the capacity region has a simple structure.

the inner bound in Theorem 3 matches the outer bound in Theorem 4, and the capacity region is given by

$$R_1 + R_2 \leq I(X_0; X_1X_2X_3). \tag{12}$$

The conditions in Lemma 5 are quite restrictive. We have the following less restrictive conditions under which our scheme is sum-rate optimal.

Lemma 6: If any of the following conditions is satisfied, the scheme in Theorem 3 is sum-rate optimal.

1) $X_1 \to X_2 \to X_3$, or
2) $I(X_0; X_3) \leq I(X_2; X_3)$, or
3) $X_2 \to X_1 \to X_3$, or
4) $I(X_0; X_3) \leq I(X_1; X_3)$.

In the following, we discuss the case of generating two keys without helpers. This can be viewed as a special case of the above scenario by setting $\mathcal{X}_3 = \Phi$. In the following, we will use $A = I(X_0; X_1X_2)$, $B = I(X_1; X_0X_2)$ and $C = I(X_2; X_0X_1)$. In particular, we have

Lemma 7: The region for the case of generating two keys without helpers is inner bounded by the convex hull of the point $(0, 0)$ and the following four pairs of (R_1, R_2):

$$\begin{aligned} P_1 &= (0, \min\{A, C\}), && (13)\\ P_2 &= (I(X_0; X_1|X_2), I(X_0; X_2)), && (14)\\ P_3 &= (I(X_0; X_1), I(X_0; X_2|X_1)), && (15)\\ P_4 &= (\min\{A, B\}, 0). && (16) \end{aligned}$$

Furthermore, the region is outer bounded by

$$\begin{aligned} R_1 &\leq \min\{A, B\}, && (17)\\ R_2 &\leq \min\{A, C\}, && (18)\\ R_1 + R_2 &\leq A. && (19) \end{aligned}$$

The inner and outer bounds are illustrated in Figure 3. From the figure, we can see that the scheme is sum key rate optimal.

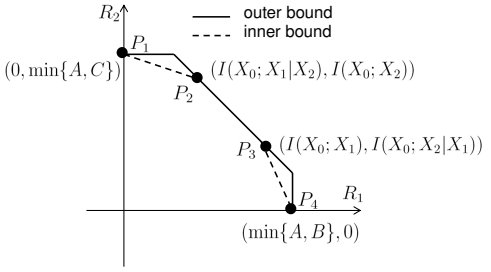

Fig. 3. Inner and outer bounds for two users without dedicated helpers.

Furthermore, if $A \leq \min\{B, C\}$, the inner and outer bounds in Lemma 7, and hence the full capacity region is characterized. However, if $A > \min\{B, C\}$, there is a gap between the inner and outer bounds. We will discuss this gap in more details in the next section.

V. Pairwise Independent Network Model

In this section, we consider a special case of the correlation model: the PIN model introduced in [10]. In the case of $J + 1$ users considered in this paper, each X_j in the PIN model has J components, such that each component is correlated with one component of another user. In particular, $X_j = [X_{j,0}, \cdots, X_{j,j-1}, X_{j,j+1}, \cdots, X_{j,J+1}]$ with $X_{j,k}$ being the component in user j that is correlated with $X_{k,j}$, the component in user k that is correlated with user j. Furthermore the pairs $(X_{j,k}, X_{k,j})$ are mutually independent. As the result, in the PIN model,

$$P_{X_0, \cdots, X_J} = \prod_{0 \leq j < k \leq J} P_{X_{j,k}, X_{k,j}}(x_{j,k}, x_{k,j}). \tag{20}$$

The special structure in the PIN model allows us to obtain better results. In particular, we can fully characterize the capacity region for the general case of generating M keys with $J - M$ dedicated helpers.

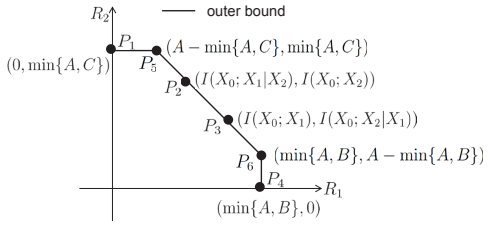

Fig. 4. The capacity region for the PIN model with two users and no dedicated helpers.

We use the two users without dedicated helpers case as an example to illustrate the main ideas that allow us to get tight results for the PIN model. As discussed in Lemma 7, points P_1, P_2, P_3 and P_4 in Figure 4 are achievable. Here, we show that, under the PIN model, P_5 and P_6 in Figure 4 are also achievable, hence there is no gap between the inner and outer bounds under the PIN model. We discuss point P_6 in detail. First, we know that

$$\begin{aligned} &\min\{A, B\} = \min\{I(X_0; X_1X_2), I(X_1; X_0X_2)\}\\ &= I(X_0; X_1|X_2) + \min\{I(X_0; X_2), I(X_1; X_2)\}. \end{aligned}$$

As mentioned above, P_4 is shown to be achievable in [9]. In [9], roughly speaking, to achieve $R_1 = \min\{A, B\}$, user 2 divides all X_2^n sequences into $2^{n\{\max\{H(X_2|X_0), H(X_2|X_1)\}+\epsilon\}}$ bins and sends the bin number as the public discussion information. By combining their local observations with the information sent by user 2, both user 0 and user 1 will be able to decode X_2^n, from which both user 0 and user 1 create an additional key (in addition to the key that can be generated from the correlation between X_0 and X_1 with a rate of $I(X_0; X_1|X_2)$) with a rate

$$\begin{aligned} &H(X_2) - \max\{H(X_2|X_0), H(X_2|X_1)\}\\ &= \min\{I(X_0; X_2), I(X_1; X_2)\}, \end{aligned}$$

which is the contribution of node 2 in achieving $R_1 = \min\{A, B\}$. This scheme is sufficient for achieving P_4. However, this scheme is not able to achieve P_6 when $\min\{A, B\} = B$. The issue is that by requiring both user 0 and user 1 to fully recover X_2^n, user 2 reveals too much information about its observations. With the special pairwise independent structure, one can achieve $R_1 = \min\{A, B\}$ by asking user 2 to reveal less information about X_2^n, and user 0 and user 2 recover only part of X_2^n. In this way, user 0 and user 2 can use the leftover

undisclosed common randomness to generate the key K_2 with rate $A - \min\{A, B\}$. In particular, one can use the following simple two-step approach to achieve Point P_6.

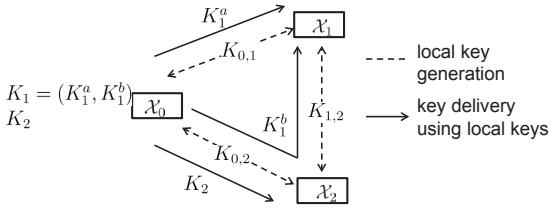

Fig. 5. The scheme to achieve P_6.

Figure 5 summarizes the steps of our approach. In the first step, these three users generate local keys using only the correlated component. In particular, we have $X_n^2 = (X_{2,0}^n, X_{2,1}^n)$ in the PIN model. User 2 randomly divides all $X_{2,0}^n$ sequences into $2^{n(H(X_{2,0}|X_{0,2})+\epsilon)}$ and sends the bin number to user 0. By combining $X_{0,2}^n$ with this bin number, user 0 can recover $X_{2,0}^n$ with a high probability, from which user 2 and user 0 can generate a key $K_{0,2}$ with a rate $I(X_{0,2}; X_{2,0}) - \epsilon$. Similarly, user 2 randomly divides all $X_{2,1}^n$ sequences into $2^{n(H(X_{2,1}|X_{1,2})+\epsilon)}$ bins and sends the bin number to user 1. By combining $X_{1,2}^n$ with this bin number, user 1 can recover $X_{2,1}^n$ with a high probability, from which user 2 and user 1 can generate a key $K_{1,2}$ with a rate $I(X_{1,2}; X_{2,1})$. In the same manner, user 0 and user 1 can generate a key $K_{0,1}$ with a rate $I(X_{0,1}; X_{1,0}) - \epsilon$ using the correlation $(X_{0,1}^n, X_{1,0}^n)$. Using these local keys, we construct a graph with three nodes and the capacity of the edge between i and j is $n(I(X_{i,j}; X_{j,i}) - \epsilon)$. In the second step, user 0 generates keys K_1 and K_2 and delivers them to user 1 and user 2 using local keys generated in the first step through routing. In particular, user 0 can randomly generate a uniformly distributed key $K_1 = (K_1^a, K_1^b)$ in which K_1^a has a rate $I(X_{0,1}; X_{1,0}) - \epsilon = I(X_0; X_1|X_2) - \epsilon$ and K_1^b has a rate $\min\{I(X_{0,2}; X_{2,0}), I(X_{1,2}; X_{2,1})\} - \epsilon = \min\{I(X_0; X_2), I(X_1; X_2)\} - \epsilon$. User 0 delivers K_1^a, which has $n(I(X_{0,1}; X_{1,0}) - \epsilon)$ bits, directly to user 1 by encrypting it using $K_{0,1}$ via the one-time pad scheme. User 1 can then recover K_1^a using $K_{0,1}$. User 0 delivers K_1^b, which has $n(\min\{I(X_0; X_2), I(X_1; X_2)\} - \epsilon)$ bits, to user 1 through user 2. In particular, user 0 uses a part of $K_{0,2}$ to encrypt K_1^b using the one-time pad and sends the encrypted message to user 2, which will then decrypt it using $K_{0,2}$ and then re-encrypt it using $K_{1,2}$ and sends it to user 1. User 1 can recover K_1^b using $K_{1,2}$. Since the rate of $K_{1,2}$ is $I(X_0; X_2) - \epsilon = I(X_{0,2}; X_{2,0}) - \epsilon$ and the rate of $K_1^b = \min\{I(X_0; X_2), I(X_1; X_2)\} - \epsilon = \min\{I(X_{0,2}; X_{2,0}), I(X_{1,2}; X_{2,1})\} - \epsilon$, part of $K_{1,2}$ is not needed for delivering K_1^b. This unused part can then used to deliver K_2 to user 2. The leftover rate is $I(X_{0,2}; X_{2,0}) - \min\{I(X_{0,2}; X_{2,0}), I(X_{1,2}; X_{2,1})\} = A - \min\{A, B\}$, hence $R_2 = A - \min\{A, B\}$ is achievable.

Essentially, the approach discussed above converts the key generation problem to a two-commodity flows with a single source problem. For this simple example, it is easy to check that by adjusting the rate of keys delivered using different routes, one can not only achieve Point P_6, but also achieve the whole capacity region. This approach can be extended to handle the general case of generating multiple keys with multiple helpers. Coupled with multi-commodity flows with a single source results in the graph theory, we can characterize the full capacity region.

Theorem 8: Let $\mathcal{B} = \{B \subset \mathcal{J} : 0 \in B^c, B \cap \mathcal{M} \neq \Phi\}$. The capacity region for generating M keys with $J+1$ users (key $K_j, j \in \mathcal{M}$ for user j to be shared with user 0, and users indexed from $M+1$ to J act as helpers) is the union of all rate tuples $(R_1, \cdots, R_M)$ that satisfy the following conditions:

$$\sum_{m \in \mathcal{M} \cap B} R_m \leq \sum_{(i,j): i \in B, j \in B^c} I(X_{i,j}; X_{j,i}), \forall B \in \mathcal{B}. \quad (21)$$

VI. Conclusion

We have formulated the problem of simultaneously establishing multiple keys, one for each user in a set of users. For the case of generating multiple keys with no dedicated helpers, we have developed a scheme that achieves the sum key rate capacity. We have also extended the study to the scenario in which there are several dedicated helpers. We have developed a simple achievable scheme and derived an outer bound for the general case. We have characterized the conditions under which the developed scheme achieves the full capacity regions and conditions under which the scheme is sum-rate optimal. We have also considered the PIN model, in which we have fully characterized the capacity region for the general case of generating multiple keys with multiple dedicated helpers.

References

[1] R. Ahlswede and I. Csiszár, "Common randomness in information theory and cryptography, Part I: Secret sharing," *IEEE Trans. Inform. Theory*, vol. 39, pp. 1121–1132, July 1993.

[2] U. Maurer, "Secret key agreement by public discussion from common information," *IEEE Trans. Inform. Theory*, vol. 39, pp. 733–742, May 1993.

[3] I. Csiszár and P. Narayan, "Secrecy capacities for multiple terminals," *IEEE Trans. Inform. Theory*, vol. 50, pp. 3047–3061, Dec. 2004.

[4] S. Nitinawarat, C. Ye, A. Barg, P. Narayan, and A. Reznik, "Secret key generation for a pairwise independent network model," *IEEE Trans. Inform. Theory*, vol. 56, pp. 6482–6489, Dec. 2010.

[5] A. Khisti, S. N. Diggavi, and G. W. Wornell, "Secret-key generation using correlated sources and channels," *IEEE Trans. Inform. Theory*, vol. 58, pp. 652–670, Feb. 2012.

[6] V. Prabhakaran, K. Eswaran, and K. Ramchandran, "Secrecy via sources and channels: A secret key-secret message rate trade-off region," in *Proc. IEEE Intl. Symposium on Inform. Theory*, (Toronto, Canada), July 2008.

[7] C. Ye and P. Narayan, "The private key capacity region for three terminals," in *Proc. IEEE Intl. Symposium on Inform. Theory*, (Chicago, IL), p. 44, Jun./Jul. 2007.

[8] C. Ye and P. Narayan, "The secret key-private key capacity region for three terminals," in *Proc. IEEE Intl. Symposium on Inform. Theory*, (Adelaide, Australia), pp. 2142–2146, Sept. 4-9, 2005.

[9] I. Csiszár and P. Narayan, "Common randomness and secret key generation with a helper," *IEEE Trans. Inform. Theory*, vol. 46, pp. 344–366, Mar. 2000.

[10] C. Ye and A. Reznik, "Group secret key generation algorithms," in *Proc. IEEE Intl. Symposium on Inform. Theory*, (Nice, France), pp. 2896–2900, Jun. 2007.

[11] C. Ye, S. Mathur, A. Reznik, W. Trappe, and N. Mandayam, "Information-theoretic key generation from wireless channels," *IEEE Trans. Inform. Forensics and Security*, vol. 5, pp. 240–254, Jun. 2010.

[12] R. Ahuja, T. Magnanti, and J. Orlin, *Network Flows*. Upper Saddle River, NJ: Prentice Hall, 1993.

[13] L. Lai and L. Huie, "Simultaneously generating multiple keys in many to one networks," *IEEE Trans. Inform. Theory*, 2012. Submitted. Available at "wpi.edu/∼llai".