Title

The Inverted 3-Sum Box: General Formulation and Quantum Information Theoretic Optimality

Authors

Yao, Yuhang

Jafar, Syed A

Publication Date

2024-07-04

The Inverted 3-Sum Box: General Formulation and Quantum Information Theoretic Optimality

Yuhang Yao, Syed A. Jafar
Center for Pervasive Communications and Computing (CPCC)
University of California Irvine, Irvine, CA 92697
Email: {yuhangy5, syed}@uci.edu

Abstract

The N-sum box protocol specifies a class of $\mathbb{F}_d$ linear functions $f(W_1, \cdots, W_K) = \mathbf{V}_1 W_1 + \mathbf{V}_2 W_2 + \cdots + \mathbf{V}_K W_K \in \mathbb{F}_d^{m\times 1}$ that can be computed at information theoretically optimal communication cost (minimum number of qudits $\Delta_1, \cdots, \Delta_K$ sent by the transmitters Alice$_1$, Alice$_2$,$\cdots$, Alice$_K$, respectively, to the receiver, Bob, per computation instance) over a noise-free quantum multiple access channel (QMAC), when the input data streams $W_k \in \mathbb{F}_d^{m_k\times 1}, k \in [K]$, originate at the distributed transmitters, who share quantum entanglement in advance but are not otherwise allowed to communicate with each other. In prior work this set of optimally computable functions is identified in terms of a strong self-orthogonality (SSO) condition on the transfer function of the N-sum box. In this work we consider an 'inverted' scenario, where instead of a feasible N-sum box transfer function, we are given an arbitrary $\mathbb{F}_d$ linear function, i.e., arbitrary matrices $\mathbf{V}_k \in \mathbb{F}_d^{m\times m_k}$ are specified, and the goal is to characterize the set of all feasible communication cost tuples $(\Delta_1, \cdots, \Delta_K)$, not just based on N-sum box protocols, but across all possible quantum coding schemes. As our main result, we fully solve this problem for $K = 3$ transmitters ($K \geq 4$ settings remain open). Coding schemes based on the N-sum box protocol (along with elementary ideas such as treating qudits as classical dits, time-sharing and batch-processing) are shown to be information theoretically optimal in all cases. As an example, in the symmetric case where $\text{rk}(\mathbf{V}_1) = \text{rk}(\mathbf{V}_2) = \text{rk}(\mathbf{V}_3) \triangleq r_1$, $\text{rk}([\mathbf{V}_1, \mathbf{V}_2]) = \text{rk}([\mathbf{V}_2, \mathbf{V}_3]) = \text{rk}([\mathbf{V}_3, \mathbf{V}_1]) \triangleq r_2$, and $\text{rk}([\mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3]) \triangleq r_3$ (rk = rank), the minimum total-download cost is $\max\{1.5r_1 + 0.75(r_3 - r_2), r_3\}$.

1 Introduction

Distributed encoding of *classical* information into entangled *quantum* systems over *many-to-one* communication networks is a cross-cutting theme across a variety of active research areas that include quantum private information retrieval (QPIR) [1–3], quantum metrology and sensing [4–6], quantum machine learning [7, 8] and quantum simultaneous message passing [9, 10]. By exploiting uniquely quantum phenomena such as entanglement and superposition, the hybrid classical-quantum (CQ) paradigm promises precision, security, privacy and efficiency guarantees beyond the fundamental limits of purely classical systems. This may be accomplished, for example, by sending the entangled quantum systems to a central receiver that extracts the desired information through a joint measurement.

In order to understand the fundamental limits of many-to-one CQ systems it is imperative to study the classical information carrying capacity of a quantum multiple access (QMAC) channel. One approach in this direction focuses on the challenges posed by *noisy* quantum channels, both for *communication* tasks — where the receiver's goal is to recover the transmitters' data inputs (messages) [11–15], as well as *computation* tasks — where the receiver only wishes to retrieve a particular function (e.g., sum) of the inputs [16–18]. Advances in this direction tend to require quantum generalizations of classical random coding arguments, made especially challenging by the superadditivity of quantum capacity [19] which presents obstacles to single-letterization. Remarkably, even for a *point-to-point* noisy quantum channel, a computable closed form capacity expression is not always available.

An alternative approach, called the LC-QMAC problem [20], emerged relatively recently out of QPIR literature and focuses exclusively on the utility of transmitter-side[1] quantum *entanglement* for linear computation (LC) tasks under idealized assumptions on the QMAC, e.g., the channels through which the quantum systems are delivered to the receiver may be assumed to be noise-free. The noise-free model ensures that the capacity reflects the fundamental limits of *entanglement* as a resource for computation, rather than those of the underlying noise models and associated countermeasures. Essentially in this case, *the entanglement is the channel*, i.e., quantum entanglement introduces non-classical dependencies between the distributed quantum systems, which collectively constitute a non-trivial channel. The challenge is to optimally shape that channel through distributed coding schemes and joint measurements to match the desired computation task at the receiver, thereby *maximizing the efficiency (capacity) of the communication resource (qubits) required for the desired computation*. Idealized channel models make the problem more tractable — optimal coding schemes under this approach are more likely to be non-asymptotic, and the capacity more likely to be found in closed form, thus somewhat transparent and insightful. Indeed, this is the case when the function to be computed is simply a sum of the transmitters' inputs [20]. The LC-QMAC approach seeks a resource theoretic accounting as in [22], analogous to the degrees of freedom (DoF) studies of wireless networks [23] where the noise is similarly de-emphasized. It is a quantum extension of corresponding topics in classical network coding literature, including but not limited to *network function computation* [24–27].

It is important to note that despite the simplification afforded by idealized (rather than noisy) channel models the LC-QMAC problem remains challenging because of the long recognized [28] increased difficulty of characterizing the capacity for *computation* (rather than *communication*) tasks, as evident from the abundance of open problems in network function computation. The present

[1] Prior entanglement with the receiver is not assumed by default in the LC-QMAC, but can be modeled by including a dummy transmitter as in [21].

work falls under the LC-QMAC paradigm.

1.1 Background: N-sum Box for Linear Computation over a QMAC (LC-QMAC)

As the starting point for this work, consider the N-sum box protocol formalized in [29], which specifies a set of $\mathbb{F}_d$ linear functions that can be computed over an ideal (noise-free) N-to-1 QMAC, with N-qudits being transmitted to a central receiver, one each from each of N transmitters who share quantum entanglement in advance but are not otherwise allowed to communicate with each other. Specifically, if the n^{th} transmitter, $n \in [N]$, has classical inputs $(x_n, z_n) \in \mathbb{F}_d^2$ which it encodes into its own qudit by local Pauli X, Z operations, then after receiving 1 noise-free qudit per transmitter, following the N-sum box protocol, the receiver is able to obtain $\mathbf{y} = \mathbf{M}_x\mathbf{x} + \mathbf{M}_z\mathbf{z}$, where $\mathbf{x} = [x_1, \cdots, x_n]^\top, \mathbf{z} = [z_1, \cdots, z_n]^\top$, and $\mathbf{M}_x, \mathbf{M}_z$ are $N \times N$ matrices in $\mathbb{F}_d$ such that $\text{rank}[\mathbf{M}_x, \mathbf{M}_z] = N$ and $\mathbf{M}_x\mathbf{M}_z^\top = \mathbf{M}_z\mathbf{M}_x^\top$. The last condition is called the *strong self-orthogonality* (SSO) condition. It is worth mentioning that the N-sum box protocol emerged out of the QPIR literature [1–3] and was formalized in [29] primarily as a useful abstraction that hides the details of the underlying quantum coding schemes, and thereby makes these quantum coding applications accessible to classical coding and information theorists.

1.2 Motivating Examples

Let us motivate this work with three toy examples.

1.2.1 Toy Example 1

Given the matrices $\mathbf{M}_x = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \mathbf{M}_z = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 2 & 0 \\ 1 & 0 & 2 \end{bmatrix}$, say over $\mathbb{F}_d, d = 3$, it is readily verified that the SSO property is satisfied, giving us an N-sum box ($N = 3$) with output $\mathbf{y} = \begin{bmatrix} x_1+x_2+x_3 \\ z_1+2z_2 \\ z_1+2z_3 \end{bmatrix}$. The box can be used for example, in an LC-QMAC setting where we have 3 transmitters: Alice$_1$, Alice$_2$, Alice$_3$, with prior shared quantum entanglement, who are presented with independent classical input streams $(A, B), (C, D), (E, F)$, respectively, all symbols in $\mathbb{F}_3$, and a receiver (Bob) who wishes to compute,

$$f(A, B, C, D, E, F) = \begin{bmatrix} A+C+E \\ B+2D \\ B+2F \end{bmatrix}.$$

The total download cost incurred by the N-sum box solution in this case is 3 qudits. In fact, the scheme is information theoretically optimal in its communication cost because with i.i.d. uniform inputs the entropy $H(f(A, B, C, D, E, F)) = 3$ dits, and Holevo's bound implies that 3 dits (in this case meaning $d = 3$-ary digits) worth of information cannot be delivered by fewer than 3 qudits. By the same reasoning, given arbitrary SSO matrices $\mathbf{M}_x, \mathbf{M}_z$ we can identify the corresponding linear function that is optimally computed by the N-sum box protocol in an LC-QMAC setting.

1.2.2 Toy Example 2

Now let us consider an 'inverted' situation, i.e., instead of the $\mathbf{M}_x, \mathbf{M}_z$ matrices, we are given a desired linear function to be computed over a given QMAC. For example, suppose the three transmitters, Alice$_1$, Alice$_2$, Alice$_3$, have classical input data streams $(A), (B), (C)$, respectively, all symbols in $\mathbb{F}_3$, and Bob (the receiver) wishes to compute $g(A, B, C) = [\, A+B+C \,]$, i.e., the sum of the

three data-streams. Since the entropy of $g(A, B, C)$ is at most 1 dit per instance, Holevo's bound only indicates that the communication cost is at least 1 qudit per instance of g. One could try to *search* for an N-sum box (i.e., SSO matrices $\mathbf{M}_x, \mathbf{M}_z$) that can output $g(A, B, C)$ at the total communication cost equal to (or approaching asymptotically with joint coding across many computation instances) 1 qudit per instance, but such a search would be futile. This is because an information theoretic (min-cut) argument (cf. [20]) shows that no quantum coding scheme can allow Bob to recover $g(A, B, C)$ at a cost less than 1.5 qudits per computation.[2] The optimal total download cost is indeed 1.5 qudits in this case, and it is achievable with the N-sum box protocol [20] by coding over $L = 2$ instances so that $A = (A_1, A_2), B = (B_1, B_2), C = (C_1, C_2)$. In fact the same N-sum box as in the previous example suffices, by setting $\mathbf{x} = \left[\begin{smallmatrix} A_1 & B_1 & C_1 \end{smallmatrix}\right]^\top$ and $\mathbf{z} = \left[\begin{smallmatrix} A_2 & B_2 & C_2 \end{smallmatrix}\right]^\top$, which produces output $\left[\begin{smallmatrix} A_1+B_1+C_1 \\ A_2+2B_2 \\ A_2+2C_2 \end{smallmatrix}\right]$. Note that once Bob recovers both $A_2 + 2B_2$ and $A_2 + 2C_2$, he can add them and divide the sum by 2 to recover $A_2 + B_2 + C_2$. The inverted problem formulation — finding a suitable N-sum box protocol given the desired computation — is perhaps more natural. However, the inverted problem can be difficult to solve especially when the desired computation does not directly correspond to an SSO matrix structure, and therefore may need to be minimally expanded (e.g., by breaking $A_2 + B_2 + C_2$ into $A_2 + 2B_2$ and $A_2 + 2C_2$ as in this toy example) into a larger computation that does fit an SSO structure.

1.2.3 Toy Example 3

For our third example, suppose the 3 transmitters Alice$_1$, Alice$_2$, Alice$_3$, have classical input streams $(A), (B), (C, D)$, respectively, all symbols in $\mathbb{F}_3$, and Bob wishes to compute the function

$$h(A, B, C, D) = \left[\begin{smallmatrix} A+B+C \\ D \end{smallmatrix}\right].$$

Applying Holevo's bound for this case only shows that the communication cost must be at least 2 qudits. Min-cut arguments also produce the same bound. However, a search for such an N-sum box fails, leading to the question: *Does there always exist an N-sum box protocol that achieves the information theoretically minimal download cost per computation given an arbitrary desired linear computation over a QMAC?* Or, more generally, *what is the optimal communication cost per computation instance for an arbitrary desired linear computation over a QMAC, and how can it be achieved?* For the particular setting of Toy Example 3, it turns out that what is needed is a stronger information theoretic converse bound (see Theorem 2 in this work), that will show that the optimal communication cost is at least 2.5 qudits (per computation). Once this bound is found, an N-sum protocol that achieves it is quite apparent. Specifically, $A + B + C$ is computed as in the previous example with a total download cost of 1.5 qudits, and the remaining symbol D is recovered at the cost of 1 additional (unentangled) qudit simply by *treating qudits as classical dits* (TQC), i.e., by independent encoding of the qudit along the computational basis. In fact, if $\Delta_1, \Delta_2, \Delta_3$ represent the number of qudits (per computation instance) sent to Bob from Alice$_1$, Alice$_2$, Alice$_3$, respectively, then the (closure of) set of *all* feasible tuples is characterized as follows (see Theorem 3 in this work).

[2] We will occasionally drop the qualifier 'per computation' for the sake of brevity, with the understanding that download costs are always measured per instance of the desired function computation.

$$\mathfrak{D}^* = \left\{ \begin{bmatrix} \Delta_1 \\ \Delta_2 \\ \Delta_3 \end{bmatrix} \in \mathbb{R}^3 \;\middle|\; \begin{array}{l} \Delta_1 \geq 1/2 \\ \Delta_2 \geq 1/2 \\ \Delta_3 \geq 1 \\ \Delta_1 + \Delta_2 + \Delta_3 \geq 5/2 \end{array} \right\}, \tag{1}$$

which is illustrated in Fig. 1.

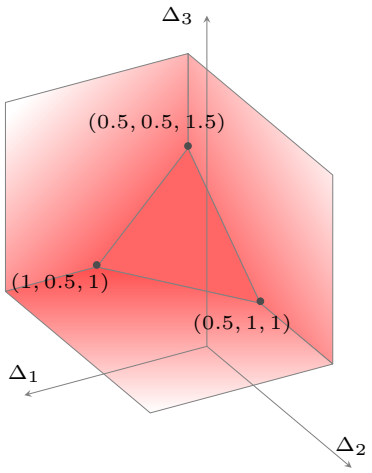

Figure 1: $\mathfrak{D}^*$ for Toy Example 3.

1.3 Summary of Contribution

To summarize the motivating examples, while the N-sum box abstraction specifies what can be computed given any choice of SSO matrices $\mathbf{M}_x, \mathbf{M}_z$, typically we are interested in the *inverted* problem formulation, where we are given only the desired $\mathbb{F}_d$ linear function f of the transmitters' inputs, and need to find the information theoretically optimal quantum coding protocol. Notably, the case where f is simply the sum of inputs has been settled in [20], and coding schemes based on the N-sum box are shown to be capacity achieving in that case. However, in the general case where f can be an arbitrary vector linear function, it is far from obvious what the optimal cost might be for computing f on a QMAC; whether that cost is achievable with an N-sum box protocol; if so, then how can it be achieved; and if not, then what else may be needed. In particular, the SSO constraint that limits the scope of N-sum box functionality is quite intriguing. Does it represent a fundamental information theoretic limitation? If so, then how does it translate into entropic constraints? Or is it merely an artifact of the N-sum box protocol that may be circumvented by other, more general constructions? Remarkably, it follows from [20] that the SSO constraint does not pose a limitation for the $K = 2$ transmitter setting.[3] Therefore, the smallest case that is open is the 3-to-1 LC-QMAC setting, which is indeed our main focus in this paper. The main contribution of this work is to answer the aforementioned questions fully for the $K = 3$ transmitter setting.

Specifically, our main result is a solution to the inverted problem identified above, hence labeled an *inverted* 3*-sum box*. Given *any* desired $\mathbb{F}_d$ linear computation f (not limited to scalar linear functions as in [20]) on a 3-to-1 QMAC, the *inverted* 3*-sum box* solution provides,

[3]This is because for linear computations the 2-sum box allows full cooperation between the two transmitters [20].

- a region $\mathfrak{D}^*$ of download cost (per computation instance) *tuples* $(\Delta_1, \Delta_2, \Delta_3)$ corresponding to Alice$_1$, Alice$_2$, Alice$_3$, such that each of these tuples is sufficient for the desired computation (note that this is a *region* of tuples, so we are not limited to just the total download cost, or to symmetric download costs),

- a coding scheme that makes use of only the N-sum box protocol and TQC to achieve the desired computation for any feasible download cost tuple in $\mathfrak{D}^*$, and

- an information theoretic converse which shows that for any download cost tuple outside the set $\mathfrak{D}^*$ the function f cannot be computed by *any* coding scheme (not limited to just the N-sum box or TQC schemes).

The result establishes the information theoretic optimality of the N-sum box protocol for the $K = 3$ transmitter LC-QMAC. Interestingly, this is indicative of the information theoretic significance of the SSO constraint, since the achievable schemes that are limited primarily by the SSO constraint, end up being information theoretically optimal.

Last but not the least, since we focus on the 3 transmitter LC-QMAC, let us recall a somewhat surprising observation from [20], that 3-way entanglement is never *necessary* to achieve capacity in the Σ-QMAC. The Σ-QMAC is a special case of the LC-QMAC where the desired computation is simply a sum of data-streams, like the setting of Toy Example 2. Recall that coding schemes based on the N-sum box are sufficient for achieving the capacity of the Σ-QMAC in [20]. In particular, [20] shows that any coding scheme for a Σ-QMAC that utilizes a 3-sum box, can be translated into an equally efficient coding scheme that utilizes only 2-sum boxes, and therefore only 2-way entanglements. For instance, in Toy Example 2, we note that $A + B + C$ can be computed equally efficiently with only 2-sum boxes by computing $f_1(A, B) = A_1 - A_2 + B_1$, $f_2(B, C) = B_2 - C_1 + C_2$, $f_3(A, C) = A_2 + C_1$, each of which requires only a 2-sum box, and then recovering the desired computations as $f_1 + f_3 = A_1 + B_1 + C_1 = g(A_1, B_1, C_1)$ and $f_2 + f_3 = A_2 + B_2 + C_2 = g(A_2, B_2, C_2)$, for the same total download cost of 1.5 qudits per computation instance. Remarkably, we find that this is no longer the case when the scope of desired computations is expanded from the Σ-QMAC to the LC-QMAC, i.e., instead of only a sum of inputs, the desired computation can be an arbitrary *vector* linear combination of inputs, as in this paper. Indeed, 3-way entanglements are *necessary* in general for *vector* linear computations. For instance, 3-way entanglements between the transmitters are necessary in the 3-transmitter LC-QMAC setting of Toy Example 1 in order to achieve the optimal cost of 3 qudits per computation. Specifically, we prove in Appendix B that with only 2-way entanglements (which allow 2-sum boxes) the total download cost for Toy Example 1 cannot be less than 3.5 qudits per computation.

Notation: For $n \in \mathbb{N}$, define $[n] \triangleq \{1, 2, \cdots, n\}$. For $a < b \in \mathbb{N}$ define $[a : b] = \{a, a + 1, \cdots, b\}$. Given a set $\mathcal{S}$, define $A_{\mathcal{S}} \triangleq \{A_s \mid s \in \mathcal{S}\}$. $\mathbb{F}_d$ denotes the finite field with order d being a power of a prime. For a matrix $M \in \mathbb{F}_d^{a\times b}$, rk$(M)$ denotes its rank over $\mathbb{F}_d$. $\mathbb{R}$ and $\mathbb{Q}$ denote the set of reals and rationals, respectively. For vectors u, v of the same length, $u \geq v$ is equivalent to $u_i \geq v_i, \forall i$ where u_i, v_i are the i^{th} component of u and v, respectively. Given a tripartite quantum system ABC in the state ρ, $H(A)_\rho$ denotes the entropy of A with respect to the state ρ. The conditional entropy $H(A \mid B)_\rho$ is defined as $H(AB)_\rho - H(B)_\rho$ and the conditional mutual information is defined as $I(A; B \mid C)_\rho = H(A \mid C)_\rho + H(B \mid C)_\rho - H(AB \mid C)_\rho$. The subscript in the information measures may be omitted for compact notation when the underlying state is obvious from the context. If the state additionally depends on a classical random variable X with distribution p_X,

and say ρ denotes the joint state of the classical-quantum system, then $H(A \mid X = x)_\rho$ denotes the entropy of A conditioned on the event $X = x$. Similar to classical information measure, we have $H(A \mid X)_\rho = \sum_x p_X(x) H(A \mid X = x)_\rho$.

2 Problem Formulation

2.1 LC-QMAC

An LC-QMAC setting is specified by the parameters $(\mathbb{F}_d, K, \mathbf{V}_1, \cdots, \mathbf{V}_K)$. $\mathbb{F}_d$ is a finite field of order d. K is the number of transmitters (denoted as Alice$_k, k \in [K]$). For $k \in [K]$, $\mathbf{V}_k$ is an $m \times m_k$ matrix with elements in $\mathbb{F}_d$. Alice$_k, k \in [K]$ has a data stream W_k, which takes values in $\mathbb{F}_d^{m_k \times 1}$, and the receiver, Bob, wants to compute an arbitrary $\mathbb{F}_d$ linear function of the data streams, $F = \mathbf{V}_1 W_1 + \cdots + \mathbf{V}_K W_K \in \mathbb{F}_d^{m \times 1}$. Without loss of generality we assume that for all $k \in [K]$,

1. $m_k \leq m$;
2. $\mathbf{V}_k$ has full column rank.

The desired computation is to be performed multiple times, for successive instances of the data streams. Specifically, for $\ell \in \mathbb{N}$, the realization of the data stream W_k corresponding to the ℓ^{th} instance of the computation is denoted as W_k^ℓ. Denote $W_k^{[L]} = [W_k^1, W_k^2, \cdots, W_k^L]$. The ℓ^{th} instance of the function to be computed is then identified as F^ℓ and we have the compact notation $F^{[L]} = [F^1, F^2, \cdots, F^L]$.

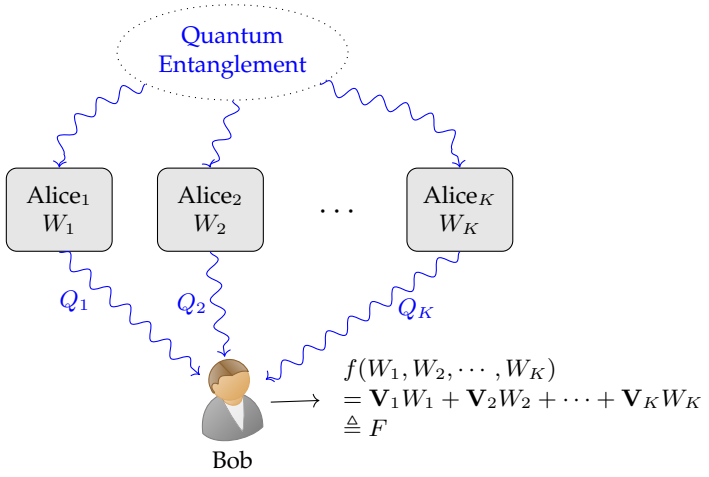

Figure 2: LC-QMAC$(\mathbb{F}_d, K, \mathbf{V}_1, \mathbf{V}_2, \cdots, \mathbf{V}_K)$. $Q_1, Q_2, \cdots, Q_K$ are entangled quantum systems. Alice$_k$ encodes W_k into Q_k, and Bob measures the joint system $Q_1 Q_2 \cdots Q_K$ to obtain the desired computation F.

2.2 Coding Schemes for LC-QMAC

For the LC-QMAC $(\mathbb{F}_d, K, \mathbf{V}_1, \cdots, \mathbf{V}_K)$, a (quantum) coding scheme involves the following elements.

- A batch size $L \in \mathbb{N}$, which represents the number of computation instances to be encoded together by the coding scheme.
- A composite quantum system $Q = Q_1 Q_2 \cdots Q_K$ comprised of K subsystems, with initial state of Q specified by the density matrix ρ^{init}.
- A set of encoders represented as quantum channels $\{\mathcal{E}_k^{(w_k)} \colon k \in [K], w_k \in \mathbb{F}_d^{m_k \times L}\}$, such that the output dimension of each $\mathcal{E}_k^{(w_k)}$ is equal to δ_k.
- A set of operators $\{\Lambda_y \colon y \in \mathcal{Y}\}$ that specify a POVM.

The coding scheme is explained as follows. There are three stages, referred to as the preparation stage, the encoding stage, and the decoding stage.

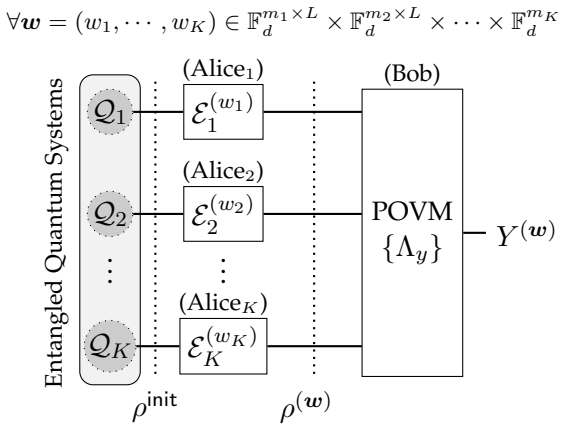

Figure 3: A quantum coding scheme for the LC-QMAC. The output measured at the receiver, $Y^{(\boldsymbol{w})}$, must be equal to $\mathbf{V}_1 w_1 + \mathbf{V}_2 w_2 + \cdots + \mathbf{V}_K w_K$, for all realizations of $(w_1, w_2, \cdots, w_K)$.

1. **(Preparation stage):** A K partite quantum system $Q_1 Q_2 \cdots Q_K$ is prepared in the initial state ρ^{init} and distributed to the Alices such that for all $k \in [K]$, Alice$_k$ has the subsystem Q_k.
2. **(Encoding stage):** For data realization (over L instances)

   $$(W_1^{[L]}, W_2^{[L]}, \cdots, W_K^{[L]}) = (w_1, w_2, \cdots, w_K),$$

   Alice$_k$ applies $\mathcal{E}_k^{(w_k)}$ to $\mathcal{Q}_k$ for $k \in [K]$. The output state of the composite quantum system is thus determined as,

   $$\rho^{(w_1, \cdots, w_K)} = \mathcal{E}_1^{(w_1)} \otimes \mathcal{E}_2^{(w_2)} \otimes \cdots \otimes \mathcal{E}_K^{(w_K)}(\rho^{\mathsf{init}}). \tag{2}$$

3. **(Decoding stage):** Bob measures $Q_1 Q_2 \cdots Q_K$ with POVM $\{\Lambda_y \colon y \in \mathcal{Y}\}$ to obtain the output random variable Y, such that,

   $$\Pr(Y = y) = \mathrm{Tr}(\rho^{(w_1, \cdots, w_K)} \Lambda_y), \ \ \forall y \in \mathcal{Y}. \tag{3}$$

A feasible coding scheme must satisfy the following correctness condition,

$$\text{[Correctness]} \qquad \Pr(Y = F^{[L]}) = 1, \tag{4}$$

for all realizations of the data streams $(w_1, \cdots, w_K) \in \mathbb{F}_d^{m_1 \times L} \times \cdots \times \mathbb{F}_d^{m_K \times L}$.

2.3 Download Cost Tuple

Given a feasible coding scheme, define

$$\mathbf{\Delta} = (\Delta_1, \cdots, \Delta_K) = \left(\frac{\log_d \delta_1}{L}, \cdots, \frac{\log_d \delta_K}{L} \right) \tag{5}$$

as the normalized download cost tuple (simply referred to as the cost tuple in the rest of the paper) achieved by the coding scheme. A cost tuple is said to be *achievable* if it is achieved by some feasible coding scheme.

2.4 Optimal Cost Region

For an LC-QMAC, the optimal cost region $\mathfrak{D}^*$ is defined as the closure of the set of all achievable cost tuples. Specifically, let $\mathfrak{C}_L$ denote the set of feasible coding schemes with batch size L. Let $\mathbf{\Delta}(\mathcal{C})$ denote the cost tuple achieved by the coding scheme $\mathcal{C}$. Define $\mathfrak{D}_L = \{\mathbf{\Delta}(\mathcal{C}) \colon \mathcal{C} \in \mathfrak{C}_L\}$. Then

$$\mathfrak{D}^* \triangleq \overline{\bigcup_{L=1}^{\infty} \mathfrak{D}_L}, \tag{6}$$

where $\overline{X}$ denotes the closure of X in $\mathbb{R}^K$.

3 Preliminaries

We briefly review some relevant known results.

3.1 N-sum box

Formally, an N-sum box is specified by a finite field $\mathbb{F}_q$, a matrix $\mathbf{M} = [\mathbf{M}_x, \mathbf{M}_z]$ where $\mathbf{M}_x$, $\mathbf{M}_z \in \mathbb{F}_q^{N \times N}$ such that $\mathrm{rk}(\mathbf{M}) = N$ and $\mathbf{M}_x \mathbf{M}_z^\top = \mathbf{M}_z \mathbf{M}_x^\top$, which is referred to as the strong self-orthogonality (SSO) property. The matrix $\mathbf{M}$ is called the *transfer* matrix. The following lemma summarizes the functionality of the N-sum box.

Lemma 1 (N-sum box [29])**.** *There exists a set of orthogonal quantum states, denoted as* $\{|\mathbf{v}\rangle_{\mathbf{M}}\}_{\mathbf{v} \in \mathbb{F}_q^{N \times 1}}$ *defined on* $\mathcal{H}_q^{\otimes N}$*, the Hilbert space of* N *q-dimensional quantum subsystems* $Q_1, Q_2, \cdots, Q_N$*, such that when* $\mathsf{X}(x_i)\mathsf{Z}(z_i)$ *is applied to* Q_i *for all* $i \in [N]$*, the state of the composite quantum system* Q *changes from* $|\mathbf{a}\rangle_{\mathbf{M}}$ *to* $|\mathbf{a} + \mathbf{M}[\begin{smallmatrix}\mathbf{x}\\ \mathbf{z}\end{smallmatrix}]\rangle_{\mathbf{M}}$ *(with global phases omitted), i.e.,* $\otimes_{i \in [N]} \mathsf{X}(x_i)\mathsf{Z}(z_i) \, |\mathbf{a}\rangle_{\mathbf{M}} \equiv |\mathbf{a} + \mathbf{M}[\begin{smallmatrix}\mathbf{x}\\ \mathbf{z}\end{smallmatrix}]\rangle_{\mathbf{M}}$*, for all* $\mathbf{x} \triangleq [x_1, \cdots, x_N]^\top \in \mathbb{F}_q^{N \times 1}$ *and* $\mathbf{z} \triangleq [z_1, \cdots, z_N]^\top \in \mathbb{F}_q^{N \times 1}$*.*

Note that each of these q^N orthogonal quantum states is uniquely indexed by a vector in $\mathbb{F}_q^N$. According to the lemma, if the input state is chosen as $|\mathbf{0}\rangle_{\mathbf{M}}$, then the output state is $|\mathbf{M}[\begin{smallmatrix}\mathbf{x}\\ \mathbf{z}\end{smallmatrix}]\rangle_{\mathbf{M}}$. Since the states are orthogonal, $\mathbf{y} = \mathbf{M}[\begin{smallmatrix}\mathbf{x}\\ \mathbf{z}\end{smallmatrix}]$ can be obtained with certainty by jointly measuring the quantum system $Q_1 Q_2 \cdots Q_N$ in the basis $\{|\mathbf{v}\rangle_{\mathbf{M}}\}_{\mathbf{v} \in \mathbb{F}_q^{N \times 1}}$.

It is noteworthy that coding schemes based on the N-sum box have been shown to be capacity achieving for the Σ-QMAC (where the desired computation is simply a sum of the transmitters' inputs) with arbitrarily distributed entanglements in [20], for the Σ-QEMAC, i.e., the Σ-QMAC where the channels are subject to erasures [21], and for several QPIR applications [1, 2].

3.2 Classical communication capacity of a noiseless quantum channel

The classical *communication* capacity of a point-to-point noisy quantum channel was studied in [12,30,31], and the special case of a noiseless channel is particularly well understood (e.g., see [12, Table I]). The noiseless channel capacity result is informally summarized as follows:

Fact 1: Without receiver-side entanglement, a δ-dimensional quantum system can carry at most $\log_d \delta$ dits of classical information;

Fact 2: With unlimited receiver-side entanglement, a δ-dimensional quantum system can carry at most $2\log_d \delta$ dits of classical information.

For our *computation* problem, the point to point *communication* capacity results yield elementary *converse bounds* through cut-set arguments [32], i.e., by separating the parties into two groups and allowing full-cooperation within each group, collectively considering each group as the transmitter or the receiver, and bounding the communication costs in the resulting communication problem. Remarkably, while cut-set arguments were sufficient to obtain tight converse bounds in the Σ-QMAC [20], these bounds will not suffice for the vector LC-QMAC problem considered in this work.

4 Results

4.1 Converse bounds on $\mathfrak{D}^*$

For any set of indices $\mathcal{K} = \{k_1, k_2, \cdots, k_{|\mathcal{K}|}\} \subseteq [K]$, let us define,

$$\mathbf{V}_{\mathcal{K}} = [\mathbf{V}_{k_1}, \mathbf{V}_{k_2}, \cdots, \mathbf{V}_{k_{|\mathcal{K}|}}]. \tag{7}$$

Further, let us introduce the compact notation,

$$r_{\mathcal{K}} = \text{rk}(\mathbf{V}_{\mathcal{K}}), \tag{8}$$

$$s_{\mathcal{K}} = \text{rk}(\mathbf{V}_{[K]}) - \text{rk}(\mathbf{V}_{[K]\setminus\mathcal{K}}), \tag{9}$$

$$\Delta_{\mathcal{K}} = \sum_{k\in\mathcal{K}} \Delta_k. \tag{10}$$

Using this compact notation, let us first formalize for our LC-QMAC setting a baseline result that follows from existing work as mentioned in Section 3.2.

Theorem 1 (Communication bounds)**.** *The following bounds hold for the LC-QMAC*$(\mathbb{F}_d, K, \mathbf{V}_1, \cdots, \mathbf{V}_K)$,

$$\Delta_{[K]} \geq r_{[K]}, \tag{11}$$

$$2\Delta_{\mathcal{K}} \geq r_{\mathcal{K}}, \qquad \forall \mathcal{K} \subseteq [K]. \tag{12}$$

This theorem essentially follows from the known capacity results of quantum communication channels (e.g., [12]) together with a cut-set argument in network coding (e.g., [32]). A formal proof is provided in Section 5.1. The following discussion elaborates upon the cut-set argument.

1. Consider Alice$_1$ – Alice$_K$ together as one transmitter that has all the data and Bob as the receiver. The receiver must be able to recover $\mathbf{V}_1 W_1^{[L]} + \mathbf{V}_2 W_2^{[L]} + \cdots + \mathbf{V}_K W_K^{[L]}$, which is $L \times r_{[K]}$ dits of information. According to Fact 1 in Section 3.2, $\log \delta_1 + \log \delta_2 + \cdots + \log \delta_K \geq L \times r_{[K]} \implies \Delta_{[K]} \geq r_{[K]}$. This gives us the bound (11).

2. Let $\mathcal{K} \subseteq [K]$. Consider the Alices with indices in $\mathcal{K}$ collectively as the transmitter, and the rest of the Alices joining Bob together as the receiver (making their data and entangled quantum resource available to Bob for free). Then the receiver must be able to recover $\sum_{k\in\mathcal{K}} \mathbf{V}_k W_k^{[L]}$ from the merged transmitter. Note that what the receiver recovered constitutes $L \times r_{\mathcal{K}}$ dits of information. According to Fact 2 in Section 3.2, we have $2\sum_{k\in[K]} \log \delta_k \geq L \times r_{\mathcal{K}} \implies 2\Delta_{\mathcal{K}} \geq r_{\mathcal{K}}$, which is the bound (12).

Next, as the first significant contribution of this work, we present the following stronger converse bounds.

Theorem 2 (Multiparty computation bounds)**.** *Let $\{\mathcal{K}_1, \mathcal{K}_2, \cdots, \mathcal{K}_T\}$ be a partition of $[K]$. Then the following bounds hold,*

$$2\Delta_{[K]} \geq s_{\mathcal{K}_1} + r_{\mathcal{K}_1} + r_{\mathcal{K}_2} + \cdots + r_{\mathcal{K}_T}, \qquad \forall T \geq 1, \tag{13}$$

$$\begin{aligned} 2\big(\Delta_{\mathcal{K}_1} + \Delta_{\mathcal{K}_2}\big) + 4\big(\Delta_{\mathcal{K}_3} + \cdots \Delta_{\mathcal{K}_T}\big) \geq \big(s_{\mathcal{K}_1} + s_{\mathcal{K}_2}\big) + \big(r_{\mathcal{K}_1} + r_{\mathcal{K}_2}\big) \\ + 2\big(r_{\mathcal{K}_3} + \cdots + r_{\mathcal{K}_T}\big), \qquad \forall T \geq 2. \end{aligned} \tag{14}$$

The proof of Theorem 2 is presented in Section 5.2. Note that (11) is recovered as a special case of (13) by setting $T = 1, \mathcal{K}_1 = [K]$ which corresponds to $r_{[K]} = s_{[K]}$. Next let us illustrate the theorem with a couple of toy examples.

4.1.1 Toy Example 4

To see how the converse bounds from Theorem 2 can be significantly stronger than those from Theorem 1, consider the following example. Suppose $K \geq 2$ and $\mathbf{V}_1 = I_{K\times K}$, $\mathbf{V}_k = \mathbf{e}_1, \forall k \in \{2, 3, \cdots, K\}$, where $I_{K\times K}$ denotes the $K \times K$ identity matrix and $\mathbf{e}_1$ is the first column of $I_{K\times K}$.[4] It is not difficult to verify that the best bound implied by Theorem 1 for the total download cost $\Delta_{[K]}$ is $\Delta_{[K]} \geq K$, whereas Theorem 2 implies $\Delta_{[K]} \geq 3K/2 - 1$. Thus, we note that the gap between the two bounds can be of the order of K. In other words, the additive gap between the baseline cut-set bounds of Theorem 1 and the optimal value of the sum-download cost $\Delta_{[K]}$, is unbounded in general.

4.1.2 Toy Example 5

Consider an LC-QMAC with $K = 4$ transmitters, namely Alice$_k$, $k \in [4]$. Each Alice$_k$ has data (x_k, z_k), say all symbols in $\mathbb{F}_3$, and sends one qudit $(d = 3)$ to Bob. Then is it possible for Bob to obtain

$$\mathbf{y} = \begin{bmatrix} x_1 \\ x_2+x_3+x_4 \\ z_1 \\ z_2+z_3+z_4 \end{bmatrix} = \mathbf{M}[\begin{smallmatrix}\mathbf{x}\\ \mathbf{z}\end{smallmatrix}],$$

where

$$\mathbf{M} = \begin{bmatrix} 1&0&0&0&0&0&0&0 \\ 0&1&1&1&0&0&0&0 \\ 0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&1&1&1 \end{bmatrix}, \qquad \mathbf{x} = [x_1, x_2, x_3, x_4]^\top, \qquad \mathbf{z} = [z_1, z_2, z_3, z_4]^\top \tag{15}$$

[4]For example, this setting includes the case of $K = 3$ transmitters, namely Alice$_1$, Alice$_2$, Alice$_3$, who have data $(A, B, C), (D), (E)$, respectively, and the receiver (Bob) desires the vector $(A + D + E, B, C)$.

by measuring the four qudits? We cannot immediately construct such an N-sum box protocol because this $\mathbf{M}$ does not satisfy the SSO condition. But could this be achieved through some other construction? Theorem 1 does not preclude the existence of such a construction because the constraints (11) and (12) are not violated. However, Theorem 2 shows that such a computation is not possible by *any* construction, i.e., it violates the laws of quantum physics. To see this, consider the $T = 4$ way partition $(\mathcal{K}_1, \mathcal{K}_2, \mathcal{K}_3, \mathcal{K}_4) = (\{1\}, \{2\}, \{3\}, \{4\})$. We have $s_{\{1\}} = 2$ and $r_{\{t\}} = 2$ for $t = 1, 2, 3, 4$, so Condition (13) in Theorem 2 implies that $2\Delta_{[4]} \geq 10$, i.e., at least a total of 5 qudits must be sent from the four Alices to Bob in order for Bob to recover such an output function.

4.2 Capacity for $K = 3$

As the main result of this work, we now characterize the capacity for LC-QMAC when $K = 3$, establishing in the process that the bounds from Theorem 1 and Theorem 2 together provide a tight converse.

Theorem 3. *For the LC-QMAC problem $(K = 3, \mathbb{F}_d, \mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3)$, the optimal cost region is,*

$$\mathfrak{D}^* = \left\{ \begin{bmatrix} \Delta_1 \\ \Delta_2 \\ \Delta_3 \end{bmatrix} \in \mathbb{R}^3 \middle| \mathbf{A} \begin{bmatrix} \Delta_1 \\ \Delta_2 \\ \Delta_3 \end{bmatrix} \geq \mathbf{B} \begin{bmatrix} \mathrm{rk}(\mathbf{V}_1) \\ \mathrm{rk}(\mathbf{V}_2) \\ \mathrm{rk}(\mathbf{V}_3) \\ \mathrm{rk}([\mathbf{V}_1, \mathbf{V}_2]) \\ \mathrm{rk}([\mathbf{V}_1, \mathbf{V}_3]) \\ \mathrm{rk}([\mathbf{V}_2, \mathbf{V}_3]) \\ \mathrm{rk}([\mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3]) \end{bmatrix} \right\}, \tag{16}$$

where

$$\mathbf{A} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \\ 1 & 1 & 1 \\ 2 & 2 & 2 \\ 2 & 2 & 2 \\ 2 & 2 & 2 \\ 4 & 2 & 2 \\ 2 & 4 & 2 \\ 2 & 2 & 4 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & -1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & -1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & -1 & 1 \\ 2 & 1 & 1 & -1 & -1 & 0 & 2 \\ 1 & 2 & 1 & -1 & 0 & -1 & 2 \\ 1 & 1 & 2 & 0 & -1 & -1 & 2 \end{bmatrix}. \tag{17}$$

The proof is divided into two parts. The direct part (achievability), i.e., (RHS of (16)) $\subseteq \mathfrak{D}^*$ is proved in Section 6. The converse, i.e., $\mathfrak{D}^* \subseteq$ (RHS of (16)) is proved in Section 5. According to Theorem 3, $\mathfrak{D}^*$ is characterized by 10 linear inequalities on $(\Delta_1, \Delta_2, \Delta_3)$ that appear in Condition (16), and thus the region $\mathfrak{D}^*$ is a polyhedron. For the achievability proof, it suffices to show that each of the corner points of the polyhedron is achievable, because the achievability of all other points then follows from a standard time-sharing argument. For the converse, we shall show that all 10 bounds hold for the cost tuple achieved by *any* feasible coding scheme, based on Theorem 1 and Theorem 2.

Remark 1. *To see Toy Example* 3 *in terms of the notation used for the problem formulation, note that we have* $W_1 = A, W_2 = B, W_3 = [C, D]^\top$ *and* $f(A, B, C, D) = [A + B + C, D]^\top$. *This corresponds to,*

$$\mathbf{V}_1 = \left[\begin{smallmatrix}1\\0\end{smallmatrix}\right], \mathbf{V}_2 = \left[\begin{smallmatrix}1\\0\end{smallmatrix}\right], \mathbf{V}_3 = \left[\begin{smallmatrix}1&0\\0&1\end{smallmatrix}\right], \textit{ and } \begin{bmatrix} \mathrm{rk}(\mathbf{V}_1) \\ \mathrm{rk}(\mathbf{V}_2) \\ \mathrm{rk}(\mathbf{V}_3) \\ \mathrm{rk}([\mathbf{V}_1, \mathbf{V}_2]) \\ \mathrm{rk}([\mathbf{V}_1, \mathbf{V}_3]) \\ \mathrm{rk}([\mathbf{V}_2, \mathbf{V}_3]) \\ \mathrm{rk}([\mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3]) \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 2 \\ 1 \\ 2 \\ 2 \\ 2 \end{bmatrix}, \tag{18}$$

which, by Theorem 3, produces the region $\mathfrak{D}^*$ *specified in* (1)*, and illustrated in Fig. 1.*

Remark 2. *In the symmetric case where* $\mathrm{rk}(\mathbf{V}_1) = \mathrm{rk}(\mathbf{V}_2) = \mathrm{rk}(\mathbf{V}_3) \triangleq r_1$, $\mathrm{rk}([\mathbf{V}_1, \mathbf{V}_2]) = \mathrm{rk}([\mathbf{V}_2, \mathbf{V}_3]) = \mathrm{rk}([\mathbf{V}_3, \mathbf{V}_1]) \triangleq r_2$, *and* $\mathrm{rk}([\mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3]) \triangleq r_3$, *the optimal value of the total-download cost from Theorem 3 is found to be* $\max\{1.5r_1 + 0.75(r_3 - r_2), r_3\}$.

4.2.1 Toy Example 6

As one more example, consider $K = 3$ transmitters, and let $(A, B, C, D, E, F, G, H, I)$ be 9 variables in a finite field $\mathbb{F}_d$. Let

$$W_1 = [A, D, G]^\top, W_2 = [B, E, H]^\top, W_3 = [C, F, I]^\top$$

and

$$f(A, B, \cdots, I) = [A + B + C,\ D + E + F,\ G,\ H,\ I]^\top.$$

From this, we obtain,

$$\mathbf{V}_1 = \left[\begin{smallmatrix}1&0&0\\0&1&0\\0&0&1\\0&0&0\\0&0&0\end{smallmatrix}\right], \mathbf{V}_2 = \left[\begin{smallmatrix}1&0&0\\0&1&0\\0&0&0\\0&0&1\\0&0&0\end{smallmatrix}\right], \mathbf{V}_3 = \left[\begin{smallmatrix}1&0&0\\0&1&0\\0&0&0\\0&0&0\\0&0&1\end{smallmatrix}\right], \tag{19}$$

$$\begin{bmatrix} \mathrm{rk}(\mathbf{V}_1) \\ \mathrm{rk}(\mathbf{V}_2) \\ \mathrm{rk}(\mathbf{V}_3) \\ \mathrm{rk}([\mathbf{V}_1, \mathbf{V}_2]) \\ \mathrm{rk}([\mathbf{V}_1, \mathbf{V}_3]) \\ \mathrm{rk}([\mathbf{V}_2, \mathbf{V}_3]) \\ \mathrm{rk}([\mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3]) \end{bmatrix} = \begin{bmatrix} 3 \\ 3 \\ 3 \\ 4 \\ 4 \\ 4 \\ 5 \end{bmatrix}, \tag{20}$$

and

$$\mathfrak{D}^* = \left\{ \begin{bmatrix} \Delta_1 \\ \Delta_2 \\ \Delta_3 \end{bmatrix} \in \mathbb{R}^3 \;\middle|\; \begin{array}{l} \Delta_1 \geq 3/2 \\ \Delta_2 \geq 3/2 \\ \Delta_3 \geq 3/2 \\ 2\Delta_1 + \Delta_2 + \Delta_3 \geq 7 \\ \Delta_1 + 2\Delta_2 + \Delta_3 \geq 7 \\ \Delta_1 + \Delta_2 + 2\Delta_3 \geq 7 \end{array} \right\}, \tag{21}$$

which is illustrated in Fig. 4.

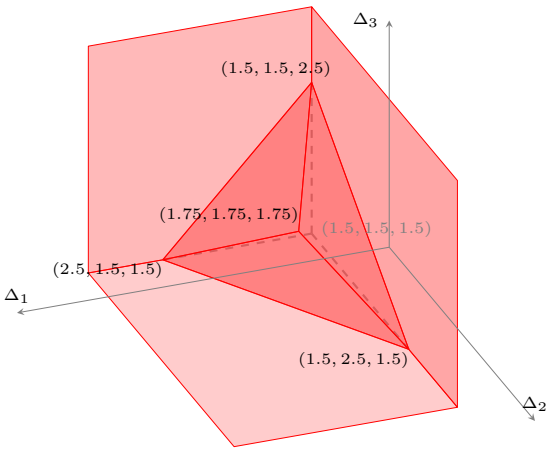

Figure 4: $\mathfrak{D}^*$ for Toy Example 6.

5 Proof of Converse Bounds

In this section we present the proof for Theorem 1 and Theorem 2. Consider any feasible LC-QMAC coding scheme with batch size L. Since the scheme must be correct for all realizations of

$$(W_1^{[L]}, W_2^{[L]}, \cdots, W_K^{[L]}) = (w_1, w_2, \cdots, w_K) \in \mathbb{F}_d^{m_1 \times L} \times \mathbb{F}_d^{m_2 \times L} \times \cdots \times \mathbb{F}_d^{m_K \times L},$$

it must be correct even under the additional assumption that $(W_1^{[L]}, W_2^{[L]}, \cdots, W_K^{[L]})$ are generated uniformly in $\mathbb{F}_d^{m_1 \times L} \times \mathbb{F}_d^{m_2 \times L} \times \cdots \times \mathbb{F}_d^{m_K \times L}$. Note that this assumption implies that $W_1^{[L]}, W_2^{[L]}, \cdots, W_K^{[L]}$ are independent. For compact notation, in the remainder of this section, we omit the superscript '$[L]$' over the data streams. Let ρ denote the state of the joint classical-quantum system $W_1 W_2 \cdots W_K Q_1 Q_2 \cdots Q_K$ in the **encoding stage**.

Lemma 2 (No-communication)**.** *$I(W_{\mathcal{J}}; W_{\mathcal{I}} Q_{\mathcal{I}})_\rho = 0$ for exclusive subsets $\mathcal{I}, \mathcal{J} \subseteq [K]$. Since conditional mutual information is non-negative, this directly implies that $I(W_{\mathcal{J}}; Q_{\mathcal{I}} \mid W_{\mathcal{I}})_\rho = 0$ and that $H(Q_{\mathcal{I}} \mid W_{\mathcal{I}}, W_{\mathcal{J}})_\rho = H(Q_{\mathcal{I}} \mid W_{\mathcal{I}})_\rho$.*

Proof. Since $W_1, W_2, \cdots, W_K$ are assumed independent, this implies $W_{\mathcal{I}}$ and $W_{\mathcal{J}}$ are independent. The lemma now follows from the no-communication theorem, e.g., see [33]. □

5.1 Proof of Theorem 1

Recall that in the decoding stage, Bob measures $Q_{[K]}$ to obtain Y, from which he gets $F^{[L]}$ (written as F to simplify notation). Therefore, for any $\mathcal{K} \subseteq [K]$,

$$\begin{aligned}
& L \times r_{\mathcal{K}} \\
&= I(W_{\mathcal{K}}; F \mid W_{[K]\setminus\mathcal{K}}) && (22)\\
&= I(W_{\mathcal{K}}; Y \mid W_{[K]\setminus\mathcal{K}}) && (23)\\
&\leq I(W_{\mathcal{K}}; Q_{[K]} \mid W_{[K]\setminus\mathcal{K}}) && (24)\\
&\leq H(Q_{[K]}) && (25)
\end{aligned}$$

$$\leq \sum_{k\in[K]} \log \delta_k \tag{26}$$

$$\implies \Delta_{[K]} \geq r_{\mathcal{K}} \tag{27}$$

Plugging in $\mathcal{K} = [K]$ proves (11). Information measures on and after Step (24) are with respect to the state ρ. Step (24) follows from Holevo bound, since Bob measures $Q_{[K]}$ to obtain Y. Step (25) is because $W_{\mathcal{K}}$ and $W_{[K]\setminus\mathcal{K}}$ are classical, and thus conditioning on any realization of $W_{[K]\setminus\mathcal{K}}$, the mutual information between $W_{\mathcal{K}}$ and $Q_{[K]}$ is not greater than $H(Q_{[K]})$.

Continuing from (24),

$$\begin{aligned}
&L \times r_{\mathcal{K}} \\
&\leq I(W_{\mathcal{K}}; Q_{[K]} \mid W_{[K]\setminus\mathcal{K}}) \\
&= I(W_{\mathcal{K}}; Q_{\mathcal{K}} \mid Q_{[K]\setminus\mathcal{K}}, W_{[K]\setminus\mathcal{K}}) && (28)\\
&= H(Q_{\mathcal{K}} \mid Q_{[K]\setminus\mathcal{K}}, W_{[K]\setminus\mathcal{K}}) - H(Q_{\mathcal{K}} \mid Q_{[K]\setminus\mathcal{K}}, W_{[K]}) && (29)\\
&\leq H(Q_{\mathcal{K}} \mid Q_{[K]\setminus\mathcal{K}}, W_{[K]\setminus\mathcal{K}}) + H(Q_{\mathcal{K}} \mid W_{\mathcal{K}}) && (30)\\
&\leq 2H(Q_{\mathcal{K}}) && (31)\\
&\leq 2\sum_{k\in\mathcal{K}} \log \delta_k && (32)\\
&\implies 2\Delta_{\mathcal{K}} \geq r_{\mathcal{K}} && (33)
\end{aligned}$$

This proves (12). Step (28) follows from Lemma 2, which implies $I(W_{\mathcal{K}}; Q_{[K]\setminus\mathcal{K}} \mid W_{[K]\setminus\mathcal{K}}) = 0$. Step (30) follows from the Araki-Lieb triangle inequality, by conditioning on $W_{[K]}$, and noting that $H(Q_{\mathcal{K}} \mid W_{[K]}) = H(Q_{\mathcal{K}} \mid W_{\mathcal{K}})$, as implied by Lemma 2. Step (31) holds because conditioning does not increase entropy.

5.2 Proof of Theorem 2

We need the following lemmas.

Lemma 3. *For $\mathcal{K} \subseteq [K]$,*

$$L \times s_{\mathcal{K}} \leq H(Q_{[K]}) - H(Q_{[K]} \mid W_{\mathcal{K}}). \tag{34}$$

Proof.

$$\begin{aligned}
&L \times s_{\mathcal{K}} \\
&= L \times (r_{[K]} - r_{[K]\setminus\mathcal{K}}) \\
&= H(F) - H\left(\sum_{k\in[K]\setminus\mathcal{K}} \mathbf{V}_k W_k\right) && (35)\\
&= H(F) - H(F \mid W_{\mathcal{K}}) && (36)\\
&= I(F; W_{\mathcal{K}}) && (37)\\
&= I(Y; W_{\mathcal{K}}) && (38)\\
&\leq I(Q_{[K]}; W_{\mathcal{K}}) && (39)\\
&= H(Q_{[K]}) - H(Q_{[K]} \mid W_{\mathcal{K}}) && (40)
\end{aligned}$$

Step (39) follows from Holevo's bound. □

Lemma 4. *For $\{\mathcal{K}_1, \mathcal{K}_2, \cdots, \mathcal{K}_T\}$ a partition of $[K]$,*

$$\begin{aligned}
&L \times (s_{\mathcal{K}_1} + r_{\mathcal{K}_2} + \cdots + r_{\mathcal{K}_T}) \\
&\le H(Q_{[K]}) - H(Q_{\mathcal{K}_1} \mid W_{\mathcal{K}_1}) + H(Q_{\mathcal{K}_2} \mid W_{\mathcal{K}_2}) + \cdots + H(Q_{\mathcal{K}_T} \mid W_{\mathcal{K}_T}).
\end{aligned} \tag{41}$$

Proof. According to Lemma 3 and (30), we have

$$\begin{aligned}
&L \times (s_{\mathcal{K}_1} + r_{\mathcal{K}_2} + \cdots + r_{\mathcal{K}_T}) \\
&\le H(Q_{[K]}) - H(Q_{[K]} \mid W_{\mathcal{K}_1}) \\
&\quad + \sum_{i=2}^{T} \Big(H(Q_{\mathcal{K}_i} \mid Q_{[K]\setminus\mathcal{K}_i}, W_{[K]\setminus\mathcal{K}_i}) + H(Q_{\mathcal{K}_i} \mid W_{\mathcal{K}_i}) \Big) && (42) \\
&\le H(Q_{[K]}) - H(Q_{[K]} \mid W_{\mathcal{K}_1}) \\
&\quad + \sum_{i=2}^{T} \Big(H(Q_{\mathcal{K}_i} \mid Q_{\mathcal{K}_1\cup\cdots\cup\mathcal{K}_{i-1}\cup\mathcal{K}_{i+1}\cup\cdots\cup\mathcal{K}_T}, W_{\mathcal{K}_1}) + H(Q_{\mathcal{K}_i} \mid W_{\mathcal{K}_i}) \Big) && (43) \\
&\le H(Q_{[K]}) - H(Q_{\mathcal{K}_1} \mid W_{\mathcal{K}_1}) - H(Q_{\mathcal{K}_2\cup\mathcal{K}_3\cup\cdots\cup\mathcal{K}_T} \mid Q_{\mathcal{K}_1}, W_{\mathcal{K}_1}) \\
&\quad + \sum_{i=2}^{T} \Big(H(Q_{\mathcal{K}_i} \mid Q_{\mathcal{K}_1\cup\cdots\cup\mathcal{K}_{i-1}}, W_{\mathcal{K}_1}) + H(Q_{\mathcal{K}_i} \mid W_{\mathcal{K}_i}) \Big) && (44) \\
&= H(Q_{[K]}) - H(Q_{\mathcal{K}_1} \mid W_{\mathcal{K}_1}) + H(Q_{\mathcal{K}_2} \mid W_{\mathcal{K}_2}) + \cdots + H(Q_{\mathcal{K}_T} \mid W_{\mathcal{K}_T}) && (45)
\end{aligned}$$

where in Steps (43) and (44) we use the fact that conditioning does not increase entropy. □

We proceed as follows. First, by Lemma 4 and (30), we have

$$\begin{aligned}
&L \times (s_{\mathcal{K}_1} + r_{\mathcal{K}_1} + r_{\mathcal{K}_2} + \cdots + r_{\mathcal{K}_T}) \\
&\le H(Q_{[K]}) + H(Q_{\mathcal{K}_1} \mid Q_{\mathcal{K}_2\cup\cdots\cup\mathcal{K}_T}, W_{\mathcal{K}_2\cup\cdots\cup\mathcal{K}_T}) \\
&\quad + H(Q_{\mathcal{K}_2} \mid W_{\mathcal{K}_2}) + \cdots + H(Q_{\mathcal{K}_T} \mid W_{\mathcal{K}_T}) && (46) \\
&\le 2 \sum_{k\in[K]} \log \delta_k && (47) \\
&\implies (13) && (48)
\end{aligned}$$

Next, noting the symmetry in Lemma 4, we have

$$\begin{aligned}
&L \times (s_{\mathcal{K}_2} + r_{\mathcal{K}_1} + r_{\mathcal{K}_3} + \cdots + r_{\mathcal{K}_T}) \\
&\le H(Q_{[K]}) - H(Q_{\mathcal{K}_2} \mid W_{\mathcal{K}_2}) + H(Q_{\mathcal{K}_1} \mid W_{\mathcal{K}_1}) + H(Q_{\mathcal{K}_3} \mid W_{\mathcal{K}_3}) + \cdots + H(Q_{\mathcal{K}_T} \mid W_{\mathcal{K}_T}).
\end{aligned} \tag{49}$$

Adding (41) and (49), we obtain

$$\begin{aligned}
&L \times \big(s_{\mathcal{K}_1} + s_{\mathcal{K}_2}\big) + \big(r_{\mathcal{K}_1} + r_{\mathcal{K}_2}\big) + 2\big(r_{\mathcal{K}_3} + \cdots + r_{\mathcal{K}_T}\big) \\
&\le 2H(Q_{[K]}) + 2\big(H(Q_{\mathcal{K}_3} \mid W_{\mathcal{K}_3}) + \cdots + H(Q_{\mathcal{K}_T} \mid W_{\mathcal{K}_T})\big) && (50) \\
&\le 2 \sum_{k\in\mathcal{K}_1\cup\mathcal{K}_2} \log \delta_k + 4 \sum_{k\in\mathcal{K}_3\cup\cdots\cup\mathcal{K}_T} \log \delta_k && (51) \\
&\implies (14) && (52)
\end{aligned}$$

6 Proof of Theorem 3: Achievability

6.1 Standard form of the linear function

Given the LC-QMAC problem specified by $(\mathbb{F}_d, K = 3, \mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3)$, the function computed at Bob is by definition,

$$F = \mathbf{V}_1 W_1 + \mathbf{V}_2 W_2 + \mathbf{V}_3 W_3. \tag{53}$$

According to [34, Lemma 2], there exist $\mathbb{F}_d$ matrices (with full column ranks and m rows each)

$$\{U_{123}, U_{12}, U_{13}, U_{23}, U_{1(2,3)}, U_{2(1,3)}, U_{3(1,2)}, U_1, U_2, U_3\}$$

such that,

1. $\begin{bmatrix} U_{123} & U_{12} & U_{13} & U_{1(2,3)} & U_1 \end{bmatrix}$ form a basis for the column span of $\mathbf{V}_1$;
2. $\begin{bmatrix} U_{123} & U_{12} & U_{23} & U_{2(1,3)} & U_2 \end{bmatrix}$ form a basis for the column span of $\mathbf{V}_2$;
3. $\begin{bmatrix} U_{123} & U_{13} & U_{23} & U_{3(1,2)} & U_3 \end{bmatrix}$ form a basis for the column span of $\mathbf{V}_3$;
4. $\begin{bmatrix} U_{123} & U_{12} & U_{13} & U_{23} & U_{1(2,3)} & U_{2(1,3)} & U_1 & U_2 \end{bmatrix}$ form a basis for the column span of $[\mathbf{V}_1, \mathbf{V}_2]$;
5. $\begin{bmatrix} U_{123} & U_{12} & U_{13} & U_{23} & U_{1(2,3)} & U_{3(1,2)} & U_1 & U_3 \end{bmatrix}$ form a basis for the column span of $[\mathbf{V}_1, \mathbf{V}_3]$;
6. $\begin{bmatrix} U_{123} & U_{12} & U_{13} & U_{23} & U_{2(1,3)} & U_{3(1,2)} & U_2 & U_3 \end{bmatrix}$ form a basis for the column span of $[\mathbf{V}_2, \mathbf{V}_3]$;
7. $\begin{bmatrix} U_{123} & U_{12} & U_{13} & U_{23} & U_{2(1,3)} & U_{3(1,2)} & U_1 & U_2 & U_3 \end{bmatrix}$ form a basis for the column span of $[\mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3]$.
8. $U_{1(2,3)}, U_{2(1,3)}$ and $U_{3(1,2)}$ have the same size and $U_{1(2,3)} = U_{2(1,3)} + U_{3(1,2)}$.

Let n_* denote the number of columns of U_*, for $* \in \{1, 2, 3, 12, 13, 23, 123\}$. The number of columns for $U_{1(2,3)}$ (the same for $U_{2(1,3)}$ and $U_{3(1,2)}$) is denoted as n_o.

Recall that each $\mathbf{V}_k$ is an $m \times m_k$ matrix. Since we assume without loss of generality that $m_k \leq m$ and each $\mathbf{V}_k$ has full column rank, it follows that there exist invertible matrices R_1, R_2, R_3 such that

$$\mathbf{V}_1 = \begin{bmatrix} U_{123} & U_{12} & U_{13} & U_{1(2,3)} & U_1 \end{bmatrix} R_1, \tag{54}$$
$$\mathbf{V}_2 = \begin{bmatrix} U_{123} & U_{12} & U_{23} & U_{2(1,3)} & U_2 \end{bmatrix} R_2, \tag{55}$$
$$\mathbf{V}_3 = \begin{bmatrix} U_{123} & U_{13} & U_{23} & U_{3(1,2)} & U_3 \end{bmatrix} R_3. \tag{56}$$

Thus, (53) becomes

$$\begin{aligned} F = &\begin{bmatrix} U_{123} & U_{12} & U_{13} & U_{1(2,3)} & U_1 \end{bmatrix} (R_1 W_1) \\ &+ \begin{bmatrix} U_{123} & U_{12} & U_{23} & U_{2(1,3)} & U_2 \end{bmatrix} (R_2 W_2) \\ &+ \begin{bmatrix} U_{123} & U_{13} & U_{23} & U_{3(1,2)} & U_3 \end{bmatrix} (R_3 W_3). \end{aligned} \tag{57}$$

R_kW_k can be considered as the (m_k-dimensional) data available to Alice$_k$ for $k \in [3]$. It will be convenient to write R_kW_k as,

$$R_1W_1 = \begin{bmatrix} A_{123} \\ A_{12} \\ A_{13} \\ A_o \\ A_1 \end{bmatrix}, R_2W_2 = \begin{bmatrix} B_{123} \\ B_{12} \\ B_{23} \\ B_o \\ B_2 \end{bmatrix}, R_3W_3 = \begin{bmatrix} C_{123} \\ C_{13} \\ C_{23} \\ C_o \\ C_3 \end{bmatrix}, \tag{58}$$

where $A_{123}, A_{12}, A_{13}, \cdots, C_o, C_3$ are vectors with elements drawn in $\mathbb{F}_d$, with X_* being an n_*-length vector for $X \in \{A, B, C\}$ and $* \in \{o, 1, 2, 3, 12, 13, 23, 123\}$. Then, (57) becomes,

$$F = \underbrace{\begin{bmatrix} U_{123} & U_{12} & U_{13} & U_{23} & U_{2(1,3)} & U_{3(1,2)} & U_1 & U_2 & U_3 \end{bmatrix}}_{\mathbf{U}} \begin{bmatrix} A_{123} + B_{123} + C_{123} \\ A_{12} + B_{12} \\ A_{13} + C_{13} \\ B_{23} + C_{23} \\ A_o + B_o \\ A_o + C_o \\ A_1 \\ B_2 \\ C_3 \end{bmatrix} \tag{59}$$

by noting that $U_{1(2,3)} = U_{2(1,3)} + U_{3(1,2)}$. Since $\mathbf{U}$ is a basis (and thus has full column rank), computing F is equivalent to computing $\tilde{F}$, where,

$$\tilde{F} = \begin{bmatrix} A_{123} + B_{123} + C_{123} \\ A_{12} + B_{12} \\ A_{13} + C_{13} \\ B_{23} + C_{23} \\ A_o + B_o \\ A_o + C_o \\ A_1 \\ B_2 \\ C_3 \end{bmatrix}, \tag{60}$$

such that all A_* symbols come from Alice$_1$, B_* come from Alice$_2$, and C_* come from Alice$_3$. Let us refer to the form in (60) as the *standard* form of the linear computation for $K = 3$. We will refer to the elements of $\tilde{F}$ as demands, that the achievable scheme will need to satisfy. For example, the achievable scheme should satisfy n_{123} dimensions of Bob's demands along $A_{123} + B_{123} + C_{123}$.

Specifically, the standard form is composed of,

1. $A_{123} + B_{123} + C_{123}$, which is an n_{123}-dimensional 3-way sum and each term comes from a different Alice;

2. $A_{12} + B_{12}$, which is an n_{12}-dimensional 2-way sum of the inputs from Alice$_1$ and Alice$_2$; $A_{13} + C_{13}$, which is an n_{13}-dimensional 2-way sum of the inputs from Alice$_1$ and Alice$_3$; $B_{23} + C_{23}$, which is an n_{23}-dimensional 2-way sum of the inputs from Alice$_2$ and Alice$_3$, such that $A_{12}, A_{13}, B_{12}, B_{23}, C_{13}, C_{23}$ are different terms.

3. A_o+B_o, which is an n_o-dimensional 2-way sum of the inputs from Alice$_1$ and Alice$_2$; A_o+C_o, which is another n_o-dimensional 2-way sum of the inputs from Alice$_1$ and Alice$_3$, such that the same A_o appears in both $A_o + B_o$ and $A_o + C_o$. Since $A_o + B_o$ and $A_o + C_o$ always have the same dimension n_o, in the following they shall always be considered together.

4. A_1, an n_1-dimensional vector from Alice$_1$; B_2, an n_2-dimensional vector from Alice$_2$, and C_3, an n_3-dimensional vector from Alice$_3$.

With this form, we can evaluate Theorem 3 in terms of $\{n_* \mid * \in \{1,2,3,12,13,23,123,o\}\}$ as

$$\mathfrak{D}^* = \left\{ \begin{bmatrix} \Delta_1 \\ \Delta_2 \\ \Delta_3 \end{bmatrix} \in \mathbb{R}^3 \middle| \mathbf{A} \begin{bmatrix} \Delta_1 \\ \Delta_2 \\ \Delta_3 \end{bmatrix} \geq \mathbf{C} \begin{bmatrix} n_{123} \\ n_{12} \\ n_{13} \\ n_{23} \\ n_o \\ n_1 \\ n_2 \\ n_3 \end{bmatrix} \right\}, \tag{61}$$

where

$$\mathbf{A} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \\ 1 & 1 & 1 \\ 2 & 2 & 2 \\ 2 & 2 & 2 \\ 2 & 2 & 2 \\ 4 & 2 & 2 \\ 2 & 4 & 2 \\ 2 & 2 & 4 \end{bmatrix}, \mathbf{C} = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 2 & 1 & 1 & 1 \\ 3 & 2 & 2 & 2 & 3 & 2 & 1 & 1 \\ 3 & 2 & 2 & 2 & 3 & 1 & 2 & 1 \\ 3 & 2 & 2 & 2 & 3 & 1 & 1 & 2 \\ 4 & 3 & 3 & 2 & 4 & 2 & 2 & 2 \\ 4 & 3 & 2 & 3 & 4 & 2 & 2 & 2 \\ 4 & 2 & 3 & 3 & 4 & 2 & 2 & 2 \end{bmatrix}. \tag{62}$$

In the remainder of this section, the goal is to prove that (RHS of (61)) $\subseteq \mathfrak{D}^*$.

6.2 Building block protocols

Let us list the building block protocols that will be used to establish the achievable region. The first building block protocol is based on trivially treating qudits as classical dits (TQC). It is summarized as follows.

[TQC]: For any transmitter with $\mathbb{F}_d$ input x, by receiving one (encoded) qudit from that transmitter, the receiver can measure x with certainty. This protocol is suitable for satisfying Bob's demands along certain dimensions of A_1, B_2, C_3 in (60). Specifically, when applying TQC to satisfy certain dimensions of A_1, the protocol is referred to as **P1**. An amortized cost tuple $(1,0,0)$ is used for this protocol as with (Alice$_1$, Alice$_2$, Alice$_3$) sending $(1,0,0)$ qudit, one dimension of A_1 demand is satisfied for Bob. Similarly, **P2** with amortized cost tuple $(0,1,0)$ refers to TQC for satisfying B_2, and **P3** with amortized cost tuple $(0,0,1)$ refers to TQC for satisfying a C_3 demand.

The rest of the building block protocols **P4 – P17** are based on the N-sum box (Lemma 1). Note that to apply Lemma 1 one must first specify the transfer matrix $\mathbf{M} = [\mathbf{M}_x, \mathbf{M}_z]$ with full row rank N and $\mathbf{M}_x\mathbf{M}_z^\top = \mathbf{M}_x\mathbf{M}_z^\top$ (SSO property). For our purpose, we need the following two N-sum boxes.

Box 1: A 2-sum box with transfer matrix $\mathbf{M}_1 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix}$.

Box 2: A 3-sum box with transfer matrix $\mathbf{M}_2 = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 & -1 \end{bmatrix}$.

It is readily verified that $\mathbf{M}_1, \mathbf{M}_2$ satisfy the SSO property. Using Box 1 with transfer matrix $\mathbf{M}_1$, we develop the following protocols.

[2-way-sums]: For 2 transmitters with $\mathbb{F}_d$ inputs (x_1, z_1), and (x_2, z_2), respectively, by receiving one qudit from each transmitter, the receiver can measure two sums $(x_1 + x_2, z_1 + z_2)$ with certainty. The negative sign can be handled by transmitter-side local operations. This protocol is suitable for satisfying demands along certain dimensions of $A_{12}+B_{12}$, referred to as **P4** with amortized cost tuple $(0.5, 0.5, 0)$, or $A_{13}+C_{13}$ (**P5** with amortized cost tuple $(0.5, 0, 0.5)$), or $B_{23} + C_{23}$ (**P6** with amortized cost tuple $(0, 0.5, 0.5)$). In addition, it is used to satisfy certain demands along the dimensions of $A_o + B_o$ or $A_o + C_o$. It will be sufficient to use 2-way-sums to satisfy demands for the *same* number of dimensions in $A_o + B_o$ as in $A_o + C_o$. Specifically, by letting (Alice$_1$, Alice$_2$, Alice$_3$) send $(1, 1, 0) + (1, 0, 1) = (2, 1, 1)$ qudits, we satisfy 2 demand dimensions in each of $A_o + B_o$ and $A_o + C_o$. The amortized cost tuple is $(1, 0.5, 0.5)$ per dimension of $A_o + B_o$ and $A_o + C_o$. Denote this protocol as **P7**. Then note that (A_o+B_o, A_o+C_o), $(A_o+B_o, -B_o+C_o)$ and (A_o+C_o, B_o-C_o) are computationally equivalent (invertible) expressions, i.e., any one of them suffices to compute all three of them. Therefore, alternatively, by sending $(1, 2, 1)$, or $(1, 1, 2)$ qudits, (Alice$_1$, Alice$_2$, Alice$_3$) can also satisfy 2 dimensions in both $A_o + B_o$ and $A_o + C_o$. This gives us another two protocols **P8** and **P9**, with respective amortized cost tuples $(0.5, 1, 0.5)$ and $(0.5, 0.5, 1)$.

[Superdense coding]: Setting $x_2 = z_2 = 0$ in 2-way-sums, by receiving one qudit from each of the two transmitters, the receiver can measure (x_1, z_1) with certainty. Note that this is exactly the superdense coding protocol, and the second transmitter only provides entangled qudits. This protocol is suitable for satisfying demands along certain dimensions of A_1 (referred to as **P10** if Alice$_2$ provides the entanglement, or **P11** if Alice$_3$ provides the entanglement). The amortized cost tuple for **P10** is $(0.5, 0.5, 0)$ per dimension of A_1, and for **P11** is $(0.5, 0, 0.5)$. Similarly we define **P12** with amortized cost tuple $(0.5, 0.5, 0)$, **P13** with amortized cost tuple $(0, 0.5, 0.5)$ as the protocols that use superdense coding to satisfy each dimension of B_2, and define **P14** with amortized cost tuple $(0.5, 0, 0.5)$, **P15** with amortized cost tuple $(0, 0.5, 0.5)$ as the protocols that use superdense coding to satisfy each dimension of C_3.

[3-way-sums]: For 3 transmitters with $\mathbb{F}_d$ inputs (u_1, v_1, w_1, x_1), (u_2, v_2, w_2, x_2) and (u_3, v_3, w_3, x_3), respectively, by applying 2-way-sums once to each pair of the three transmitters, with appropriate precoding at the transmitters, the receiver obtains $[(u_1 - v_1) + u_2, (w_1 - x_1) + w_2]$, $[v_2 + (v_3 - u_3), x_2 + (x_3 - w_3)]$ and $[v_1 + u_3, x_1 + w_3]$. From these, it is easy to verify that the receiver is able to obtain $[u_1 + u_2 + u_3, v_1 + v_2 + v_3, w_1 + w_2 + w_3, x_1 + x_2 + x_3]$. In the process, the receiver receives 6 qudits, 2 from each transmitter, and the output is 4 dimensions

of 3-way sums. This protocol is suitable for satisfying demands along certain dimensions of $A_{123}+B_{123}+C_{123}$. Note that the amortized cost tuple is $(0.5, 0.5, 0.5)$ per dimension of 3-way-sum. Denote this protocol as **P16**.

Using Box 2 with transfer matrix $\mathbf{M}_2$, we further develop the following protocols.

[3+2+2]: For 3 transmitters with with inputs (x_1, z_1), (x_2, z_2) and (x_3, z_3) respectively, by receiving one qudit from each transmitter, the receiver can measure three sums $(x_1+x_2+x_3, z_1+z_2, z_1+z_3)$ with certainty. Note that the same z_1 appears in both 2-way sums. This protocol is suitable for satisfying demands along certain dimensions of $(A_{123}+B_{123}+C_{123}, A_o+B_o, A_o+C_o)$. The amortized cost tuple is $(1, 1, 1)$ per dimension in each of $A_{123}+B_{123}+C_{123}$, A_o+B_o and A_o+C_o. Denote this protocol as **P17**.

[3+1+1]: Setting $z_1 = 0$ in **P5** allows the receiver to measure $(x_1+x_2+x_3, z_2, z_3)$ with certainty. This protocol is useful for satisfying demands along certain dimensions of $(A_{123}+B_{123}+C_{123}, A_1, B_2)$ (referred to as **P18**), or $(A_{123}+B_{123}+C_{123}, A_1, C_3)$ (referred to as **P19**), or $(A_{123}+B_{123}+C_{123}, B_2, C_3)$ (referred to as **P20**). The amortized cost tuple is $(1, 1, 1)$ for **P18–P20**.

6.3 Achievable region with auxiliary variables

Define $\mathfrak{D}_{\text{achi}}$ as follows,

$$\mathfrak{D}_{\text{achi}} = \left\{ \begin{bmatrix}\Delta_1\\ \Delta_2\\ \Delta_3\end{bmatrix} \in \mathbb{R}^3 \;\middle|\; \underbrace{\begin{bmatrix}\Delta_1\\ \Delta_2\\ \Delta_3\end{bmatrix} \geq \mathbf{D}\begin{bmatrix}\lambda_1\\ \lambda_2\\ \lambda_3\\ \vdots\\ \lambda_{20}\end{bmatrix}, \mathbf{E}\begin{bmatrix}\lambda_1\\ \lambda_2\\ \lambda_3\\ \vdots\\ \lambda_{20}\end{bmatrix} \geq \begin{bmatrix}n_{123}\\ n_{12}\\ n_{13}\\ n_{23}\\ n_o\\ n_1\\ n_2\\ n_3\end{bmatrix}}_{\textbf{Cond1}}, \underbrace{\begin{bmatrix}0\\0\\0\\ \vdots\\ 0\end{bmatrix} \leq \begin{bmatrix}\lambda_1\\ \lambda_2\\ \lambda_3\\ \vdots\\ \lambda_{20}\end{bmatrix} \in \mathbb{Q}^{20}}_{\textbf{Cond2}(\mathbb{Q})} \right\}, \tag{63}$$

where

$$\mathbf{D} = \begin{bmatrix} 1 & 0 & 0 & 0.5 & 0.5 & 0 & 1 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0 & 0.5 & 0 & 0.5 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0.5 & 0 & 0.5 & 0.5 & 1 & 0.5 & 0.5 & 0 & 0.5 & 0.5 & 0 & 0.5 & 0.5 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0.5 & 0.5 & 0.5 & 0.5 & 1 & 0 & 0.5 & 0 & 0.5 & 0.5 & 0.5 & 0.5 & 1 & 1 & 1 & 1 \end{bmatrix}, \tag{64}$$

$$\mathbf{E} = \begin{bmatrix} 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&1 \\ 0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&1&1&1&0&0&0&0&0&0&0&1&0&0&0 \\ 1&0&0&0&0&0&0&0&0&1&1&0&0&0&0&0&0&1&1&0 \\ 0&1&0&0&0&0&0&0&0&0&0&1&1&0&0&0&0&1&0&1 \\ 0&0&1&0&0&0&0&0&0&0&0&0&0&1&1&0&0&0&1&1 \end{bmatrix}. \tag{65}$$

Let us first establish that $\mathfrak{D}_{\text{achi}} \subseteq \mathfrak{D}^*$. This is argued as follows. $(\lambda_1, \lambda_2, \lambda_3, \cdots, \lambda_{20})$ are the amortized amounts of usage of the corresponding building block protocols (**P1**–**P20**). Since the batch size L can be chosen to be any positive integer, λ_i are allowed to be any non-negative rationals. Therefore, as long as there exist such non-negative $\lambda_{[20]}$ that satisfy the condition in (63), a feasible coding scheme can be constructed from the combination of the aforementioned building block protocols. Denote $\overline{\mathfrak{D}_{\text{achi}}}$ as the closure of $\mathfrak{D}_{\text{achi}}$ in $\mathbb{R}^3$. It then follows that $\overline{\mathfrak{D}_{\text{achi}}} \subseteq \mathfrak{D}^*$ as $\mathfrak{D}^*$ is closed by definition. To obtain $\overline{\mathfrak{D}_{\text{achi}}}$, let

$$\mathfrak{D} \triangleq \left\{(\Delta_{[3]}, \lambda_{[20]}) \in \mathbb{R}^{23} \mid \mathbf{Cond1}, \mathbf{Cond2}(\mathbb{Q})\right\}, \tag{66}$$

where the conditions $\mathbf{Cond1}$ and $\mathbf{Cond2}(\mathbb{Q})$ have appeared in (63). It is readily seen that the closure of $\mathfrak{D}$ in $\mathbb{R}^{23}$ is equal to

$$\overline{\mathfrak{D}} = \left\{(\Delta_{[3]}, \lambda_{[20]}) \in \mathbb{R}^{23} \mid \mathbf{Cond1}, \mathbf{Cond2}(\mathbb{R})\right\} \tag{67}$$

where $\mathbf{Cond2}(\mathbb{R})$ is the condition $\mathbf{Cond2}(\mathbb{Q})$ with $\mathbb{Q}$ replaced by $\mathbb{R}$.

Let ϕ be the mapping from $\mathbb{R}^{23}$ to $\mathbb{R}^3$ such that

$$\phi(\Delta_{[3]}, \lambda_{[20]}) = \Delta_{[3]}. \tag{68}$$

It follows that $\mathfrak{D}_{\text{achi}} = \phi(\mathfrak{D})$. Since ϕ is continuous, $\phi(\overline{\mathfrak{D}}) \subseteq \overline{\mathfrak{D}_{\text{achi}}}$ [35, Ex. 9.7]. On the other hand, $\mathfrak{D}_{\text{achi}} \subseteq \phi(\overline{\mathfrak{D}})$, and thus $\overline{\mathfrak{D}_{\text{achi}}} \subseteq \overline{\phi(\overline{\mathfrak{D}})} = \phi(\overline{\mathfrak{D}})$, where the last step is because $\phi(\overline{\mathfrak{D}})$ is a 3-dimensional polyhedron and thus closed. We conclude that

$$\overline{\mathfrak{D}_{\text{achi}}} = \phi(\overline{\mathfrak{D}}) = \left\{\Delta_{[3]} \in \mathbb{R}^3 \mid \mathbf{Cond1}, \mathbf{Cond2}(\mathbb{R})\right\}. \tag{69}$$

6.4 Eliminating the auxiliaries

Recall that our goal is to show that (RHS of (61)) $\subseteq \mathfrak{D}^*$. Since $\overline{\mathfrak{D}_{\text{achi}}} \subseteq \mathfrak{D}^*$, it suffices to show that (RHS of (61)) $\subseteq \overline{\mathfrak{D}_{\text{achi}}}$. This is done by Fourier-Motzkin elimination. We also show this explicitly in Appendix A.

7 Conclusion

The information theoretic optimality of the N-sum box protocol for all 3 user $\mathbb{F}_q$ linear computations in the LC-QMAC setting, as established in this work, is both promising and intriguing. In particular, it motivates a natural follow up question – does this optimality hold for any number of users? Since the N-sum box is constrained primarily by the SSO condition, generalized optimality results could shed light on the information theoretic significance of this condition. Based on this work, one would expect that generalizations beyond 3 users might require both new converse bounds, as well as larger subspace-decompositions. In addition to the results of this work, the previously established optimality of N-sum box protocols in the Σ-QMAC [20] under generalized entanglement distribution patterns, and in the Σ-QEMAC [21] (where the quantum channels are subject to erasures) bodes well for future efforts towards these generalizations.

A Eliminating auxiliaries

Our goal is to show that, given non-negative $(\Delta_1, \Delta_2, \Delta_3)$ and $(n_{123}, n_{12}, \cdots, n_3)$ that satisfy

$$\mathbf{A} \begin{bmatrix} \Delta_1 \\ \Delta_2 \\ \Delta_3 \end{bmatrix} \geq \mathbf{C} \begin{bmatrix} n_{123} \\ n_{12} \\ n_{13} \\ n_{23} \\ n_o \\ n_1 \\ n_2 \\ n_3 \end{bmatrix}, \tag{70}$$

there exist non-negative $(\lambda_1, \cdots, \lambda_{20})$ such that

$$\begin{bmatrix} \Delta_1 \\ \Delta_2 \\ \Delta_3 \end{bmatrix} \geq \mathbf{D} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \vdots \\ \lambda_{20} \end{bmatrix}, \tag{71}$$

$$\mathbf{E} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \vdots \\ \lambda_{20} \end{bmatrix} \geq \begin{bmatrix} n_{123} \\ n_{12} \\ n_{13} \\ n_{23} \\ n_o \\ n_1 \\ n_2 \\ n_3 \end{bmatrix}. \tag{72}$$

Let us use analogy for intuition. First note that all variables considered are non-negative reals. Let $\Delta_1, \Delta_2, \Delta_3$ be the amounts of three non-exchangeable currencies, namely Currency$_1$, Currency$_2$ and Currency$_3$, say corresponding to 3 different countries, that are available to an importer of goods, subject to the constraint (70). Let **P1**–**P20** represent 20 different goods, and $\lambda_1, \lambda_2, \cdots, \lambda_{20}$ be the amounts of these goods to be imported, respectively. $\mathbf{D}$ specifies the prices of the 20 goods sold by the three countries. Specifically, the $(i,j)^{th}$ entry of $\mathbf{D}$, i.e., $D_{i,j}$ is the cost in terms of Currency$_i$ to import a unit of $\mathbf{P}j$. Condition (71) says that the total amount of any type of currency spent cannot exceed the amount of that type of currency given to the importer. Further constraints are specified by $\mathbf{E}$: each row in (72) places a demand on the amounts of goods that need to be imported. There are 8 rows in $\mathbf{E}$. Let us refer to the 8 requirements as **R1** – **R8**. For example, the first row of (72) corresponds to **R1**, and with $\mathbf{E}$ as defined in (65), this constraint says that the total amount of **P16** to **P20** imported has to be at least n_{123}. We will show that as long as the importer is given the amount of currencies $(\Delta_1, \Delta_2, \Delta_3)$ that satisfy (70), then there is always a strategy (described as follows) to satisfy all the constraints.

The strategy is divided into the following main steps.

1. Satisfy **R2**, **R3**, **R4** by importing n_{12} units of **P4**, n_{13} units of **P5** and n_{23} units of **P6**. This incurs a cost $0.5(n_{12}+n_{13}, n_{12}+n_{23}, n_{13}+n_{23})$ in terms of (Currency$_1$, Currency$_2$, Currency$_3$), and the feasibility (availability of sufficient currency) is guaranteed by (70).

2. Import $\min\{n_{123}, n_o\} \triangleq \tilde{n}$ unit of **P17**, which incurs a cost $(\tilde{n}, \tilde{n}, \tilde{n})$. The feasibility is guaranteed by (70). Note that after this step, either **R1** or **R5** is satisfied: if $\tilde{n} = n_{123}$, then **R1** is satisfied; if $\tilde{n} = n_o$, then **R5** is satisfied.

3. Case I: If $\tilde{n} = n_{123}$, then import appropriate amount of **P7**, **P8**, **P9** to satisfy **R5**, and import appropriate amount of **P1–P3**, **P10–P15** to satisfy **R6–R8**. Case II: if $\tilde{n} = n_o < n_{123}$, then import appropriate amount of **P1–P3**, **P10–P16**, **P18–P20** to satisfy **R1** and **R6–R8**.

While in the first two steps we specify exact amount of imported goods and the currencies spent, the third step is more complicated and needs further analysis, since it involves further optimizations that are not so straightforward. In the following we analyze the third step.

Recall that the initial currency amounts given to the importer are $(\Delta_1, \Delta_2, \Delta_3)$. Thus, after the first two steps, there remains

$$(\Delta_1, \Delta_2, \Delta_3) - 0.5(n_{12} + n_{13}, n_{12} + n_{23}, n_{13} + n_{23}) - (\tilde{n}, \tilde{n}, \tilde{n}) \tag{73}$$

$$\triangleq (\Delta_1', \Delta_2', \Delta_3') \tag{74}$$

currency for the importer to allocate.

Case I: In this case $\tilde{n} = n_{123}$. Define $n_o' \triangleq n_o - n_{123} \geq 0$. After the first two steps, the importer still needs to fulfill **R5–R8**. For **R5**, since $\tilde{n} = n_{123}$ out of n_o is satisfied by importing **P17**, there remains another n_o' to be fulfilled. The first four rows of (70) imply,

$$\begin{cases} \Delta_1' \geq \frac{1}{2}(n_o' + n_1) \\ \Delta_2' \geq \frac{1}{2}(n_o' + n_2) \\ \Delta_3' \geq \frac{1}{2}(n_o' + n_3) \\ \Delta_1' + \Delta_2' + \Delta_3' \geq 2n_o' + n_1 + n_2 + n_3 \end{cases} \tag{75}$$

Therefore, there exist non-negative $(a_i, b_i)_{i\in[3]}$ such that

$$\Delta_i' = \frac{1}{2}(n_o' + n_i) + a_i + b_i, \ \ \forall i \in [3], \tag{76}$$

and

$$a_1 + a_2 + a_3 = \frac{n_o'}{2}, \ \ b_1 + b_2 + b_3 = \frac{n_1 + n_2 + n_3}{2}. \tag{77}$$

The strategy then finds

$$\begin{bmatrix} 1 & 0.5 & 0.5 \\ 0.5 & 1 & 0.5 \\ 0.5 & 0.5 & 1 \end{bmatrix} \begin{bmatrix} \lambda_7 \\ \lambda_8 \\ \lambda_9 \end{bmatrix} = \begin{bmatrix} \frac{n_o'}{2} + a_1 \\ \frac{n_o'}{2} + a_2 \\ \frac{n_o'}{2} + a_3 \end{bmatrix} \tag{78}$$

$$\implies \begin{bmatrix} \lambda_7 \\ \lambda_8 \\ \lambda_9 \end{bmatrix} = \begin{bmatrix} \frac{n_o'}{4} + \frac{3a_1 - a_2 - a_3}{2} \\ \frac{n_o'}{4} + \frac{3a_2 - a_1 - a_3}{2} \\ \frac{n_o'}{4} + \frac{3a_3 - a_1 - a_2}{2} \end{bmatrix} \geq \begin{bmatrix} \frac{n_o'}{4} - \frac{a_1 + a_2 + a_3}{2} \\ \frac{n_o'}{4} - \frac{a_1 + a_2 + a_3}{2} \\ \frac{n_o'}{4} - \frac{a_1 + a_2 + a_3}{2} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{79}$$

With this choice of $(\lambda_7, \lambda_8, \lambda_9)$, the remaining n_o' part in **R5** is satisfied, as $\lambda_7 + \lambda_8 + \lambda_9 = n_o'$. Now, only **R6 – R8** remain to be fulfilled, and the remaining currency amounts are,

$$\left(\frac{n_1}{2} + b_1, \frac{n_2}{2} + b_2, \frac{n_3}{2} + b_3\right) \triangleq (\Delta_1'', \Delta_2'', \Delta_3''). \tag{80}$$

The importer will then import **P1–P3**, **P10–P15** to satisfy **R6–R8**. We claim that this is feasible as long as the following conditions hold,

$$\begin{cases} \Delta_i'' \geq \frac{n_i}{2}, \forall i \in [3], \\ \Delta_1'' + \Delta_2'' + \Delta_3'' \geq n_1 + n_2 + n_3 \end{cases} \tag{81}$$

Intuitively, this claim (formalized in Lemma 5) follows from the fact that **P1–P3** are from TQC and **P10–P15** are from superdense coding. This condition (81) is satisfied because b_1, b_2, b_3 are non-negative and because $b_1 + b_2 + b_3 = \frac{n_1+n_2+n_3}{2}$. Therefore, **R5–R8** are satisfied. This completes the proof for Case I.

The claim is formalized in the following lemma, which will be useful again in the sequel.

Lemma 5. *Say the remaining demands to be satisfied are n_1, n_2, n_3 corresponding to* **R6**, **R7**, **R8**, *respectively. If the remaining currencies $(\Delta_1, \Delta_2, \Delta_3)$ satisfy $\Delta_i \geq \frac{n_i}{2}, \forall i \in [3]$ and $\Delta_1+\Delta_2+\Delta_3 \geq n_1+n_2+n_3$, then there exist non-negative $(\lambda_i)_{i\in\{1,2,3,10,\cdots,15\}}$ such that*

$$\begin{cases} \Delta_1 \geq \lambda_1 + 0.5(\lambda_{10} + \lambda_{11} + \lambda_{12} + \lambda_{14}) \\ \Delta_2 \geq \lambda_2 + 0.5(\lambda_{10} + \lambda_{12} + \lambda_{13} + \lambda_{15}) \\ \Delta_3 \geq \lambda_3 + 0.5(\lambda_{11} + \lambda_{13} + \lambda_{14} + \lambda_{15}) \\ \lambda_1 + \lambda_{10} + \lambda_{11} \geq n_1 \\ \lambda_2 + \lambda_{12} + \lambda_{13} \geq n_2 \\ \lambda_3 + \lambda_{14} + \lambda_{15} \geq n_3 \end{cases} \tag{82}$$

Note that the first three conditions in (82) imply that the available currency is sufficient for the amounts corresponding to $(\lambda_i)_{i\in\{1,2,3,10,\cdots,15\}}$, with the remaining λ_i set to zero. The last three conditions in (82) imply that **R6**, **R7**, **R8** are satisfied.

Proof. It suffices to show the existence of $(\lambda_i)_{i\in\{1,2,3,10,\cdots,15\}}$ for

$$(\Delta_1, \Delta_2, \Delta_3) \in \left\{ \left(\frac{n_1}{2}, \frac{n_2}{2}, \frac{n_1+n_2}{2} + n_3\right), \left(\frac{n_1}{2}, \frac{n_1+n_3}{2} + n_2, \frac{n_3}{2}\right), \left(\frac{n_2+n_3}{2} + n_1, \frac{n_2}{2}, \frac{n_3}{2}\right) \right\}$$

because other cases can be reduced to a convex combination of these 3 points. Due to symmetry it suffices to consider the first case, i.e., $(\Delta_1, \Delta_2, \Delta_3) = (\frac{n_1}{2}, \frac{n_2}{2}, \frac{n_1+n_2}{2} + n_3)$. The solution for $(\lambda_i)_{i\in\{1,2,3,10,\cdots,15\}}$ in this case is listed explicitly as,

$$\lambda_3 = n_3, \lambda_{11} = n_1, \lambda_{13} = n_2, \quad \lambda_i = 0, \forall i \in \{1, 2, 10, 12, 14, 15\}. \tag{83}$$

The proof of the lemma is thus complete. □

Case II: In this case $\tilde{n} = n_o$ and $n_{123} > n_o$. Define $n'_{123} \triangleq n_{123} - n_o > 0$. After the first two steps, the importer still needs to fulfill **R1**, **R6–R8**. For **R1**, since $\tilde{n} = n_o$ out of the n_{123} constraint is already satisfied by importing **P17**, there only remains another n'_{123} to be fulfilled. To this end, we will show that it suffices to import certain amounts of **P1–P3**, **P10–P16**, **P18–P20**. Starting with

(70), we note that the remaining currency amounts $(\Delta_1', \Delta_2', \Delta_3')$ after the first two steps satisfy

$$\begin{cases} \Delta_1' \geq \frac{1}{2}(n_{123}' + n_1) \\ \Delta_2' \geq \frac{1}{2}(n_{123}' + n_2) \\ \Delta_3' \geq \frac{1}{2}(n_{123}' + n_3) \\ \Delta_1' + \Delta_2' + \Delta_3' \geq n_{123}' + \frac{n_1+n_2+n_3}{2} + \Gamma \\ \Delta_1' + \frac{\Delta_2'}{2} + \frac{\Delta_3'}{2} \geq n_{123}' + \frac{n_1+n_2+n_3}{2} \\ \Delta_2' + \frac{\Delta_1'}{2} + \frac{\Delta_3'}{2} \geq n_{123}' + \frac{n_1+n_2+n_3}{2} \\ \Delta_3' + \frac{\Delta_1'}{2} + \frac{\Delta_2'}{2} \geq n_{123}' + \frac{n_1+n_2+n_3}{2} \end{cases} \tag{84}$$

where we define,

$$\Gamma \triangleq \max\left\{\frac{n_{123}' + n_1}{2}, \frac{n_{123}' + n_2}{2}, \frac{n_{123}' + n_3}{2}, \frac{n_1 + n_2 + n_3}{2}\right\}. \tag{85}$$

$$\begin{cases} \Delta_1' \geq \lambda_1 + \frac{\lambda_{10}+\lambda_{11}+\lambda_{12}+\lambda_{14}+\lambda_{16}}{2} + \lambda_{18} + \lambda_{19} + \lambda_{20} \\ \Delta_2' \geq \lambda_2 + \frac{\lambda_{10}+\lambda_{12}+\lambda_{13}+\lambda_{15}+\lambda_{16}}{2} + \lambda_{18} + \lambda_{19} + \lambda_{20} \\ \Delta_3' \geq \lambda_3 + \frac{\lambda_{11}+\lambda_{13}+\lambda_{14}+\lambda_{15}+\lambda_{16}}{2} + \lambda_{18} + \lambda_{19} + \lambda_{20} \\ \lambda_{16} + \lambda_{18} + \lambda_{19} + \lambda_{20} \geq n_{123}' \ \textbf{R1} \\ \lambda_1 + \lambda_{10} + \lambda_{11} + \lambda_{18} + \lambda_{19} \geq n_1 \ \textbf{R6} \\ \lambda_2 + \lambda_{12} + \lambda_{13} + \lambda_{18} + \lambda_{20} \geq n_2 \ \textbf{R7} \\ \lambda_3 + \lambda_{14} + \lambda_{15} + \lambda_{19} + \lambda_{20} \geq n_3 \ \textbf{R8} \end{cases} \tag{86}$$

It suffices to show the existence of $(\lambda_i)_{i\in\{1,2,3,10,\cdots,16,18,19,20\}}$ for the corner points of $(\Delta_1', \Delta_2', \Delta_3')$ in the region induced by (84). Further by symmetry, it suffices to consider the following 7 subcases (II.1 – II.7).

II.1: In this case we consider

$$\begin{bmatrix} \Delta_1' \\ \Delta_2' \\ \Delta_3' \end{bmatrix} = \begin{bmatrix} \frac{n_{123}'+n_1}{2} \\ \frac{n_{123}'+n_2}{2} \\ \frac{n_{123}'+n_3}{2} \end{bmatrix} \tag{87}$$

which corresponds to the first three inequalities in (84) being tight. It can be verified that (84) then implies

$$n_1 = n_2 = n_3 = 0, \tag{88}$$

by noting the non-negativity of $n_i, \forall i \in [3]$. This means that **R5**–**R8** do not require anything. Therefore, for this subcase, the importer only needs to import n_{123}' amount of **P16** to satisfy **R1**. The feasibility is guaranteed since $\Delta_i' \geq \frac{n_{123}'}{2}$ for $i \in [3]$.

II.2: In this case we consider

$$\begin{bmatrix} \Delta_1' \\ \Delta_2' \\ \Delta_3' \end{bmatrix} = \begin{bmatrix} \frac{n_{123}'+n_1}{2} \\ \frac{n_{123}'+n_2}{2} \\ \frac{n_3}{2} + \Gamma \end{bmatrix} \tag{89}$$

which corresponds to the $1^{st}, 2^{nd}, 4^{th}$ inequalities in (84) being tight. It can be verified that (84) then implies

$$\begin{cases} n_1 \geq \min\{n_2 + n_3, n'_{123}\} \\ n_2 \geq \min\{n_1 + n_3, n'_{123}\} \end{cases}, \tag{90}$$

and we consider the following subcases.
II.2.a: $n'_{123} \geq \max\{n_1 + n_3, n_2 + n_3\}$. (90) implies $n_1 = n_2, n_3 = 0$. Import n_1 amount of **P18**, and $n'_{123} - n_1$ amount of **P16**. This is feasible as $\Delta'_i \geq \frac{n'_{123}+n_1}{2}, \forall i \in [3]$.
II.2.b: $n_1 + n_3 \geq n'_{123} \geq n_2 + n_3$. (90) implies $n_1 \geq n_2 + n_3, n_2 \geq n'_{123}$. This in turn implies that $n_1 = n_2 = n_3 = n'_{123} = 0$. No further consideration is required for this subcase.
II.2.c: $n_1 + n_3 \geq n_2 + n_3 \geq n'_{123}$. (90) implies $n_1 \geq n'_{123}, n_2 \geq n'_{123}$. Import n'_{123} **P18** and **R1** is satisfied. The remaining currency amounts are

$$\begin{bmatrix} \Delta''_1 \\ \Delta''_2 \\ \Delta''_3 \end{bmatrix} = \begin{bmatrix} \frac{n_1 - n'_{123}}{2} \\ \frac{n_2 - n'_{123}}{2} \\ \frac{n_3}{2} + \Gamma - n'_{123} \end{bmatrix} \tag{91}$$

and the remaining demands for **R6**–**R8** are

$$\begin{bmatrix} n'_1 \\ n'_2 \\ n'_3 \end{bmatrix} = \begin{bmatrix} n_1 - n'_{123} \\ n_2 - n'_{123} \\ n_3 \end{bmatrix}. \tag{92}$$

Lemma 5 then implies that the remaining demands of **R6**–**R8** can be satisfied with the remaining currencies.
II.3: In this case we consider

$$\begin{bmatrix} \Delta'_1 \\ \Delta'_2 \\ \Delta'_3 \end{bmatrix} = \begin{bmatrix} \frac{n'_{123}+n_1}{2} \\ \frac{n'_{123}+n_2}{2} \\ \frac{n'_{123}}{2} + \frac{n_2}{2} + n_3 \end{bmatrix} \tag{93}$$

which corresponds to the $1^{st}, 2^{nd}, 5^{th}$ inequalities in (84) being tight. It can be verified that (84) then implies

$$n_1 \leq \min\{n'_{123}, n_2\}. \tag{94}$$

Import n_1 amount of **P18** and $n'_{123} - n_1$ amount of **P16**. **R1** is then satisfied. The remaining currency amounts are

$$\begin{bmatrix} \Delta''_1 \\ \Delta''_2 \\ \Delta''_3 \end{bmatrix} = \begin{bmatrix} 0 \\ \frac{n_2 - n_1}{2} \\ \frac{n_2 - n_1}{2} + n_3 \end{bmatrix} \tag{95}$$

and the remaining demands for **R6**–**R8** are

$$\begin{bmatrix} n'_1 \\ n'_2 \\ n'_3 \end{bmatrix} = \begin{bmatrix} 0 \\ n_2 - n_1 \\ n_3 \end{bmatrix}. \tag{96}$$

Lemma 5 then implies that the remaining demands of **R6**–**R8** can be satisfied with the remaining currencies.

II.4: In this case we consider

$$\begin{bmatrix}\Delta_1'\\ \Delta_2'\\ \Delta_3'\end{bmatrix} = \begin{bmatrix}\frac{n_{123}'+n_1}{2}\\ \frac{n_{123}'+n_2}{2}\\ \frac{n_{123}'}{2}+\frac{n_1+n_2}{4}+\frac{n_3}{2}\end{bmatrix} \tag{97}$$

which corresponds to the $1^{st}, 2^{nd}, 7^{th}$ inequalities in (84) being tight. It can be verified that (84) then implies

$$n_{123}' \geq n_1 = n_2,\ n_3 = 0. \tag{98}$$

R8 requires nothing. Import n_1 amount of **P18**, and $n_{123}' - n_1$ amount of **P16**. The feasibility can be verified and this satisfies **R1**, **R6** and **R7**.

II.5: In this case we consider

$$\begin{bmatrix}\Delta_1'\\ \Delta_2'\\ \Delta_3'\end{bmatrix} = \begin{bmatrix}\frac{n_{123}'+n_1}{2}\\ \frac{n_1+n_2+n_3}{2}+n_{123}'-\Gamma\\ 2\Gamma-\frac{n_1}{2}-\frac{n_{123}'}{2}\end{bmatrix} \tag{99}$$

which corresponds to the $1^{st}, 4^{th}, 6^{th}$ inequalities in (84) being tight. Note that Γ is a maximum of 4 terms, and we further consider subcases according to the value of Γ as follows.

II.5.a: $\Gamma = \frac{n_{123}'+n_1}{2}$. This condition together with (84) implies

$$\min\{n_1, n_{123}'\} \geq n_2 + n_3. \tag{100}$$

Import n_2 amount of **P18**, n_3 amount of **P19**, and $n_{123}' - (n_2+n_3)$ amount of **P16**. **R1**, **R7** and **R8** are then satisfied. The remaining currencies are

$$\begin{bmatrix}\Delta_1''\\ \Delta_2''\\ \Delta_3''\end{bmatrix} = \begin{bmatrix}\frac{n_1-n_2-n_3}{2}\\ 0\\ \frac{n_1-n_2-n_3}{2}\end{bmatrix} \tag{101}$$

and the remaining demand for **R6** is

$$n_3' = n_1 - n_2 - n_3. \tag{102}$$

Lemma 5 then implies that the remaining demand of **R8** can be satisfied with the remaining currency amounts.

II.5.b: $\Gamma = \frac{n_{123}'+n_2}{2}$. This condition together with (84) implies

$$n_3 = 0,\ \ n_{123}' \geq n_1 = n_2. \tag{103}$$

R8 requires nothing. Import n_2 amount of **P18** and $n_{123}' - n_2$ amount of **P16**. The feasibility can be verified, and this satisfies **R1**, **R6** and **R7**.

II.5.c: $\Gamma = \frac{n_{123}'+n_3}{2}$. This condition together with (84) implies

$$n_2 = 0,\ \ \min\{n_{123}', n_3\} \geq n_1. \tag{104}$$

R7 requires nothing. Import n_1 amount of **P19**, $n'_{123} - n_1$ amount of **P16**, and $n_3 - n_1$ amount of **P3**. The feasibility can be verified and this satisfies **R1**, **R6** and **R8**.
II.5.d: $\Gamma = \frac{n_1+n_2+n_3}{2}$. This condition together with (84) implies

$$n_1 \geq n'_{123} \geq n_2, n_2 + n_3 \geq n'_{123}. \tag{105}$$

Import n_2 amount of **P18**, and $n'_{123} - n_2$ amount of **P19**. **R1** and **R7** are then satisfied. The remaining currencies are

$$\begin{bmatrix} \Delta''_1 \\ \Delta''_2 \\ \Delta''_3 \end{bmatrix} = \begin{bmatrix} \frac{n_1 - n'_{123}}{2} \\ 0 \\ \frac{n_1 - 3n'_{123}}{2} + n_2 + n_3 \end{bmatrix} \tag{106}$$

and the remaining demands for **R6** and **R8** are

$$n'_1 = n_1 - n'_{123},\ n'_3 = n_3 - n'_{123} + n_2. \tag{107}$$

Lemma 5 then implies that the remaining demands of **R6** and **R8** can be satisfied with the remaining currencies.
II.6: In this case we consider

$$\begin{bmatrix} \Delta'_1 \\ \Delta'_2 \\ \Delta'_3 \end{bmatrix} = \begin{bmatrix} \frac{n_1+n_2+n_3}{2} + n'_{123} - \Gamma \\ \frac{n_1+n_2+n_3}{2} + n'_{123} - \Gamma \\ 3\Gamma - \frac{n_1+n_2+n_3}{2} - n'_{123} \end{bmatrix} \tag{108}$$

which corresponds to the $4^{th}, 5^{th}, 6^{th}$ inequalities in (84) being tight. We further consider subcases according to the value of Γ as follows.
II.6.a: $\Gamma = \frac{n'_{123}+n_1}{2}$. This condition together with (84) implies

$$n'_{123} \geq n_1 = n_2 + n_3. \tag{109}$$

Import n_2 amount of **P18**, n_3 amount of **P19**, and $n'_{123} - n_1$ amount of **P16**. The feasibility can be verified, and this satisfies **R1** and **R6 – R8**.
II.6.b: $\Gamma = \frac{n'_{123}+n_2}{2}$. By symmetry, this case is the same as II.6.a.
II.6.c: $\Gamma = \frac{n'_{123}+n_3}{2}$. This condition together with (84) implies

$$\min\{n'_{123}, n_3\} \geq n_1 + n_2. \tag{110}$$

Import n_1 amount of **P19**, n_2 amount of **P20**, $n'_{123} - (n_1 + n_2)$ amount of **P16** and $n_3 - n_1 - n_2$ amount of **P3**. The feasibility can be verified and this satisfies **R1** and **R6 – R8**.
II.6.d: $\Gamma = \frac{n_1+n_2+n_3}{2}$. This condition together with (84) implies

$$\min\left\{n_1 + n_2, n_1 + n_3, n_2 + n_3, \frac{n_1 + n_2 + n_3}{2}\right\} \geq n'_{123} \geq \max\{n_1, n_2\}. \tag{111}$$

Import $n_1 + n_2 - n'_{123}$ amount of **P18**, $n'_{123} - n_2$ amount of **P19**, $n'_{123} - n_1$ amount of **P20**, and $n_1 + n_2 + n_3 - 2n_{123}$ amount of **P3**. The feasibility can be verified and this satisfies **R1** and **R6–R8**.
II.7: In this case we consider

$$\begin{bmatrix} \Delta'_1 \\ \Delta'_2 \\ \Delta'_3 \end{bmatrix} = \begin{bmatrix} \frac{n'_{123}}{2} + \frac{n_1+n_2+n_3}{4} \\ \frac{n'_{123}}{2} + \frac{n_1+n_2+n_3}{4} \\ \frac{n'_{123}}{2} + \frac{n_1+n_2+n_3}{4} \end{bmatrix} \tag{112}$$

which corresponds to the $5^{th}, 6^{th}, 7^{th}$ inequalities in (84) being tight. It can be verified that (84) then implies

$$n_i + n_j \geq n_k, \text{ for distinct } i,j,k \in [3], \;\; n'_{123} \geq \frac{n_1+n_2+n_3}{2}. \tag{113}$$

Import $\frac{n_1+n_2-n_3}{2}$ amount of **P18**, $\frac{n_1+n_3-n_2}{2}$ amount of **P19**, $\frac{n_2+n_3-n_1}{2}$ amount of **P20**, and $n'_{123} - \frac{n_1+n_2+n_3}{2}$ amount of **P16**. The feasibility can be verified and this satisfies **R1** and **R6–R8**. □

B Necessity of 3-way Entanglement for Toy Example 1

Recall that the setting in Toy Example 1 contains Alice$_1$, Alice$_2$, Alice$_3$, who have data streams $(A,B),(C,D),(E,F)$, respectively, all symbols in $\mathbb{F}_d$ with $d=3$, and a receiver (Bob) who wishes to compute,

$$f(A,B,C,D,E,F) = \begin{bmatrix} A+C+E \\ B+2D \\ B+2F \end{bmatrix}.$$

Suppose instead of all possible quantum coding schemes as specified in the problem formulation, we now only allow the transmitters to use *pairwise* entanglement throughout all the stages. Specifically, Alice$_1$ and Alice$_2$ share a bipartite quantum system $Q_1 = Q_{1,1}Q_{1,2}$ such that $Q_{1,1}$ is accessible at Alice$_1$ and $Q_{1,2}$ is accessible at Alice$_2$. Similarly, Alice$_1$ and Alice$_3$ share another quantum system Q_2 such that $Q_{2,k}$ is accessible at Alice$_k$ for $k \in \{1,3\}$; Alice$_2$ and Alice$_3$ share another quantum system Q_3 such that $Q_{3,k}$ is accessible at Alice$_k$ for $k \in \{2,3\}$. Q_1, Q_2 and Q_3 are assumed to be independent in the preparation stage, kept unentangled in the encoding stage, and measured separately in the decoding stage, whereas the subsystems $Q_{i,j}$ and $Q_{i,k}$ are allowed to be entangled for distinct $j,k \in \{1,2,3\}$. Let $\delta_i, i \in [3]$ denote the dimension of Q_i in the encoding stage. According to [20], one can lower bound the total download cost $\sum_{i\in[3]} \log_d \delta_i/L$ by the classical (unentangled) total download cost of a hypothetical problem, where there are $\binom{3}{2} = 3$ transmitters, denoted as Alice$'_1$, Alice$'_2$, Alice$'_3$, who know $(A,B,C,D),(A,B,E,F),(C,D,E,F)$, and the same receiver (Bob) who computes the same function f. This is because any output measured from Q_i can be sent directly through a same dimension classical system from Alice$'_i$ in the hypothetical setting for $i \in [3]$. In the hypothetical setting, let $X_i, i \in [3]$ be a δ'_i-dimensional (classical) system sent from Alice$'_i$. We want to obtain a lower bound for $\sum_{i\in[3]} \log_d \delta'_i/L$.

Without loss of generality, assuming that each of the data streams $A, B, \cdots, F$ is uniformly distributed in $\mathbb{F}_d^L$, we have

$$\begin{aligned}
\sum_{i\in[3]} \log_d \delta'_i &\geq H(X_1,X_2,X_3) && (114)\\
&= H(X_1,X_2,X_3,\left[\begin{smallmatrix} B+2D \\ B+2F \end{smallmatrix}\right]) && (115)\\
&= H(\left[\begin{smallmatrix} B+2D \\ B+2F \end{smallmatrix}\right]) + H(X_1,X_2,X_3 \mid \left[\begin{smallmatrix} B+2D \\ B+2F \end{smallmatrix}\right]) && (116)\\
&= 2L + H(X_1,X_2,X_3 \mid \left[\begin{smallmatrix} B+2D \\ B+2F \end{smallmatrix}\right]) && (117)\\
&\geq 2L + \frac{1}{2}\sum_{i=1}^{3} H(X_{[3]\setminus\{i\}} \mid X_i, \left[\begin{smallmatrix} B+2D \\ B+2F \end{smallmatrix}\right]) && (118)\\
&= 2L + \frac{1}{2}\sum_{i=1}^{3} H(X_{[3]\setminus\{i\}}, A+C+E \mid X_i, \left[\begin{smallmatrix} B+2D \\ B+2F \end{smallmatrix}\right]) && (119)
\end{aligned}$$

$$\geq 2L + \frac{1}{2}(3L) \tag{120}$$

$$= 3.5L \tag{121}$$

$$\implies \sum_{i\in[3]} \log_d \delta_i'/L \geq 3.5 \tag{122}$$

Step (115) holds because $(B+2D, B+2F)$ is determined by (X_1, X_2, X_3). Step (118) follows from submodularity of classical entropy, i.e., the general property that $2H(Z_1, Z_2, Z_3 \mid Z_4) \geq H(Z_1, Z_2 \mid Z_3, Z_4) + H(Z_2, Z_3 \mid Z_1, Z_4) + H(Z_3, Z_1 \mid Z_2, Z_4)$ for any classical random variables Z_1, Z_2, Z_3, Z_4. Step (119) holds because $A + C + E$ is determined by (X_1, X_2, X_3). To see Step (120), note that (X_1, A, B, C, D, F) is independent of E, so that the first term in the sum, i.e., $H(X_2, X_3, A+C+E \mid X_1, \left[\begin{smallmatrix} B+2D \\ B+2F \end{smallmatrix}\right]) \geq H(X_2, X_3, A + C + E \mid A, B, C, D, F, X_1, \left[\begin{smallmatrix} B+2D \\ B+2F \end{smallmatrix}\right]) \geq H(E) = L$, and similar reasoning applies to each of the three terms in the sum, so that their sum is lower bounded by $3L$. Therefore, the total download cost for the hypothetical problem is at least 3.5. We conclude that with only 2-way entanglement, the total download cost for Toy Example 1 is at least 3.5.

References

[1] S. Song and M. Hayashi, "Capacity of quantum private information retrieval with multiple servers," *IEEE Transactions on Information Theory*, vol. 67, no. 1, pp. 452–463, 2020.

[2] ——, "Capacity of quantum symmetric private information retrieval with collusion of all but one of servers," *IEEE Journal on Selected Areas in Information Theory*, vol. 2, no. 1, pp. 380–390, 2021.

[3] M. Allaix, S. Song, L. Holzbaur, T. Pllaha, M. Hayashi, and C. Hollanti, "On the capacity of quantum private information retrieval from MDS-coded and colluding servers," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 3, pp. 885–898, 2022.

[4] V. Giovannetti, S. Lloyd, and L. Maccone, "Quantum-enhanced measurements: Beating the standard quantum limit," *Science*, vol. 306, no. 5700, pp. 1330–1336, 2004. [Online]. Available: https://www.science.org/doi/abs/10.1126/science.1104149

[5] Z. Zhang and Q. Zhuang, "Distributed quantum sensing," *Quantum Science and Technology*, vol. 6, no. 4, p. 043001, jul 2021. [Online]. Available: https://dx.doi.org/10.1088/2058-9565/abd4c3

[6] J. Rubio, P. A. Knott, T. J. Proctor, and J. A. Dunningham, "Quantum sensing networks for the estimation of linear functions," *Journal of Physics A: Mathematical and Theoretical*, vol. 53, no. 34, p. 344001, aug 2020. [Online]. Available: https://dx.doi.org/10.1088/1751-8121/ab9d46

[7] Q. Zhuang and Z. Zhang, "Physical-layer supervised learning assisted by an entangled sensor network," *Phys. Rev. X*, vol. 9, p. 041023, Oct 2019. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevX.9.041023

[8] Y. Xia, W. Li, Q. Zhuang, and Z. Zhang, "Quantum-enhanced data classification with a variational entangled sensor network," *Phys. Rev. X*, vol. 11, p. 021047, Jun 2021. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevX.11.021047

[9] A. Kawachi and H. Nishimura, "Communication Complexity of Private Simultaneous Quantum Messages Protocols," in *2nd Conference on Information-Theoretic Cryptography (ITC 2021)*, ser. Leibniz International Proceedings in Informatics (LIPIcs), S. Tessaro, Ed., vol. 199. Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021, pp. 20:1–20:19. [Online]. Available: https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.ITC.2021.20

[10] R. B. Christensen and P. Popovski, "Private product computation using quantum entanglement," *IEEE Transactions on Quantum Engineering*, vol. 4, September 2023.

[11] A. Winter, "The capacity of the quantum multiple-access channel," *IEEE Transactions on Information Theory*, vol. 47, no. 7, pp. 3059–3065, 2001.

[12] C. Bennett, P. Shor, J. Smolin, and A. Thapliyal, "Entanglement-assisted capacity of a quantum channel and the reverse shannon theorem," *IEEE Transactions on Information Theory*, vol. 48, no. 10, pp. 2637–2655, 2002.

[13] M.-H. Hsieh, I. Devetak, and A. Winter, "Entanglement-assisted capacity of quantum multiple-access channels," *IEEE Transactions on Information Theory*, vol. 54, no. 7, pp. 3078–3090, 2008.

[14] J. Yard, P. Hayden, and I. Devetak, "Capacity theorems for quantum multiple-access channels: Classical-quantum and quantum-quantum capacity regions," *IEEE Transactions on Information Theory*, vol. 54, no. 7, pp. 3091–3113, 2008.

[15] H. Shi, M. Hsieh, and S. Guha, et al., "Entanglement-assisted capacity regions and protocol designs for quantum multiple-access channels," *npj Quantum Inf*, vol. 74, no. 7, 2021.

[16] M. A. Sohail, T. A. Atif, and S. S. Pradhan, "Unified approach for computing sum of sources over CQ-MAC," in *2022 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2022, pp. 1868–1873.

[17] M. A. Sohail, T. A. Atif, A. Padakandla, and S. S. Pradhan, "Computing sum of sources over a classical-quantum MAC," *IEEE Transactions on Information Theory*, vol. 68, no. 12, pp. 7913–7934, 2022.

[18] M. Hayashi and Á. Vázquez-Castro, "Computation-aided classical-quantum multiple access to boost network communication speeds," *Physical Review Applied*, vol. 16, no. 5, p. 054021, 2021.

[19] E. Y. Zhu, Q. Zhuang, M.-H. Hsieh, and P. W. Shor, "Superadditivity in trade-off capacities of quantum channels," *IEEE Transactions on Information Theory*, vol. 65, no. 6, pp. 3973–3989, 2019.

[20] Y. Yao and S. A. Jafar, "The capacity of classical summation over a quantum MAC with arbitrarily distributed inputs and entanglements," *IEEE Transactions on Information Theory (Early Access)*, 2024.

[21] ——, "Capacity of summation over a symmetric quantum erasure MAC with replicated inputs," 2023. [Online]. Available: https://arxiv.org/abs/2311.08386

[22] M. Mamindlapally and A. Winter, "Singleton bounds for entanglement-assisted classical and quantum error correcting codes," *IEEE Transactions on Information Theory*, vol. 69, no. 9, pp. 5857–5868, 2023.

[23] S. Jafar, "Interference alignment: A new look at signal dimensions in a communication network," in *Foundations and Trends in Communication and Information Theory*, 2011, pp. 1–136.

[24] R. Appuswamy and M. Franceschetti, "Computing linear functions by linear coding over networks," *IEEE Transactions on Information Theory*, vol. 60, no. 1, pp. 422–431, Jan. 2014.

[25] C. Huang, Z. Tan, S. Yang, and X. Guang, "Comments on cut-set bounds on network function computation," *IEEE Transactions on Information Theory*, vol. 64, no. 9, pp. 6454–6459, 2018.

[26] A. Ramamoorthy and M. Langberg, "Communicating the sum of sources over a network," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 4, pp. 655–665, 2013.

[27] Y. Yao and S. A. Jafar, "On the generic capacity of K-user symmetric linear computation broadcast," *IEEE Transactions on Information Theory*, vol. 70, no. 5, pp. 3693–3717, 2024.

[28] J. Korner and K. Marton, "How to encode the modulo-two sum of binary sources (corresp.)," *IEEE Transactions on Information Theory*, vol. 25, no. 2, pp. 219–221, 1979.

[29] M. Allaix, Y. Lu, Y. Yao, T. Pllaha, C. Hollanti, and S. Jafar, "N-sum box: An abstraction for linear computation over many-to-one quantum networks," 2023. [Online]. Available: https://arxiv.org/abs/2304.07561

[30] A. S. Holevo, "Bounds for the quantity of information transmitted by a quantum communication channel," *Problemy Peredachi Informatsii*, vol. 9, no. 3, pp. 3–11, 1973.

[31] ——, "The capacity of the quantum channel with general signal states," *IEEE Transactions on Information Theory*, vol. 44, no. 1, pp. 269–273, 1998.

[32] R. Appuswamy, M. Franceschetti, N. Karamchandani, and K. Zeger, "Network coding for computing: Cut-set bounds," *IEEE Transactions on Information Theory*, vol. 57, no. 2, pp. 1015–1030, Feb. 2011.

[33] A. Peres and D. R. Terno, "Quantum information and relativity theory," *Rev. Mod. Phys.*, vol. 76, pp. 93–123, Jan 2004. [Online]. Available: https://link.aps.org/doi/10.1103/RevModPhys.76.93

[34] Y. Yao and S. A. Jafar, "The capacity of 3 user linear computation broadcast," *IEEE Transactions on Information Theory*, vol. 70, no. 6, pp. 4414–4438, 2024.

[35] W. A. Sutherland, *Introduction to metric and topological spaces*. Oxford University Press, 2009.