

STAC: Simultaneous Transmitting and Air Computing in Wireless Data Center Networks

Shengli Zhang
Faculty of Information Engineering
Shenzhen University
Shenzhen, China
Email: zsl@szu.edu.cn

Xiugang Wu
Department of Electrical Engineering
Stanford University
CA, US
Email: x23wu@standord.edu

Ayfer Ozgur
Department of Electrical Engineering
Stanford University
CA, US
Email: aozgur@standord.edu

Abstract—The data center network (DCN), wired or wireless, features large amounts of Many-to-One (M2O) sessions. Each M2O session is currently operated based on Point-to-Point (P2P) communications and Store-and-Forward (SAF) relays, and is generally followed by certain further computation at the destination. Different from this separate P2P/SAF-based-transmission and computation strategy, this paper proposes STAC, a novel physical layer scheme that achieves Simultaneous Transmission and Air Computation in wireless DCNs. In particular, STAC takes advantage of the superposition nature of electromagnetic (EM) waves, and allows multiple transmitters to transmit in the same time slot with appropriately chosen parameters, such that the received superimposed signal can be directly transformed to the needed summation at the receiver. Exploiting the static channel environment and compact space in DCN, we propose an enhanced Software Defined Network (SDN) architecture to enable STAC, where wired connections are established to provide the wireless transceivers external reference signals. Theoretical analysis and simulation show that with STAC used, both the bandwidth and energy efficiencies can be improved severalfold.

I. INTRODUCTION

A modern Data Center (DC) typically consists of a large dedicated cluster of commercial computers (work nodes) that are housed together to store/process big files in a parallel manner. The characteristic of parallel storing/processing requires frequent communications among the work nodes, which are accomplished through Data Center Networks (DCNs). Today, DCN is the principle bottleneck in large DCs [1]. Despite of its maturity in deployment and high bandwidth, the wired DCN has a few critical problems such as flexibility, cabling complexity, device cost, over subscription, etc. These problems highly limit the efficiency and scalability of the DCN and are being exacerbated provided that a huge amount of information needs to be stored/processed within the DC and exchanged through the DCN in today's big data age.

To address this issue, some works [2]–[7] studied the possibility of constructing wireless DCNs using high frequency electromagnetic (EM) waves. The 60 GHz techniques were suggested for realizing wireless DCN links with bandwidth comparable to wireline connections [2], [3], while the blockage and directivity problems associated with the EM waves can be significantly mitigated by utilizing the strategies of ceiling reflection and 3D beamforming [4]. Free-space optical DCN communications were also investigated, and were

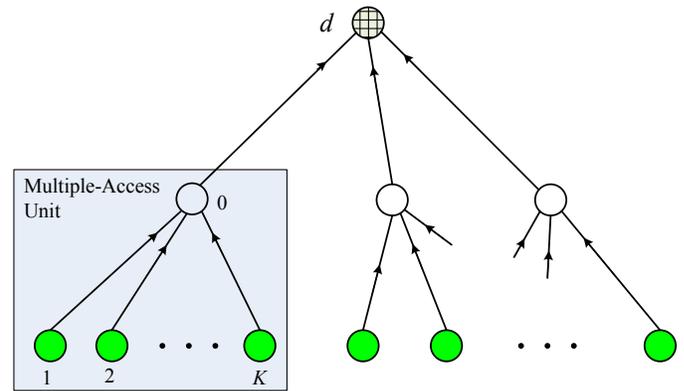


Fig. 1. Illustration of one M2O session. Source nodes (solid circles) transmit information to destination d via relay nodes (hollow circles).

shown to achieve some further improvements, including higher bandwidth and nearly perfect directivity [5]. On the other hand, from the structural perspective, the work [6] considered augmenting the wired DCN with added wireless flyways, and [7] demonstrated that a completely wireless DCN with a Cayley structure is feasible and performs even better than the wired DCN.

Wireless DCN is different from today's ubiquitous wireless networks, through traffic patterns to network structures. These differences can provide new challenges, as well as possibilities, to design more efficient wireless DCNs. One particular challenge in DC is the large amounts of Many-to-One (M2O) sessions, which brings some new problems for DCNs, especially with the Point-to-Point (P2P) communication and the Store-and-Forward (SAF) relay strategies. The M2O sessions arise from various DC applications, e.g., Google File System (GFS) [8] and MapReduce [9] framework. Due to the limited transmission range of high frequency EM waves, these M2O sessions are operated through multi-hopping over hierarchical multiple-access units as shown in Fig. 1, where each hop is based on the P2P communication and followed by the SAF relay to the next hop. Specifically, in the multiple-access unit as depicted, with Time Division Multiple-Access (TDMA), the source nodes $1, 2, \dots, K$ successively transmit their information digits to the relay node 0 in different time slots with P2P

strategy, and the relay stores all its received digits in the buffer before forwarding them to the destination d . Since the node 0's buffer and input/output bandwidth are shared by all the K source nodes, the transmission performance could be poor, especially when K is large. The nearer to the destination, the severer this problem will be, as the information that needs to be transmitted accumulates along the way. In fact, the problem of TCP throughput collapse caused by M2O transmissions in data center networks have been noted as incast problem [10].

A. A New Scheme: STAC

Rather than regarding the traffic of M2O feature as a nuisance, we propose a new physical layer scheme, dubbed STAC (Simultaneous Transmission and Air Computation), to take advantages of the *superposition* nature of EM waves and the M2O transmissions. Our STAC are based on two key observations on the distinguishing features of wireless DCNs.

Observation 1: One feature in DCs is that these M2O sessions are generally followed by certain further computations at the destination nodes. These computations normally satisfy the *commutative* and *associative* operational laws, with weighted summation being the typical case (e.g., in linear network coded storage [11] and MapReduce-based machine learning [12] applications). This opens up the possibility of dividing a whole computation task into several sub-tasks that can be conducted at the intermediate relay nodes, rather than demanding the final destination do all the jobs. In other words, instead of forwarding all the received digits, the relay could perform some intermediate computation and then forward only the output of the computation, thereby utilizing the bandwidth more efficiently¹. Considering that the bottleneck of the development of DCs lies in the DCN, not the compute capabilities of the work nodes, we believe that such Compute-and-Forward (CAF) relay strategy is preferable to the traditional SAF strategy for DCNs.

Observation 2: Another feature in DCs is that the static closed environment, where all the work nodes are closely placed in one relatively small rooms. As a result, the transceiver positions and the channel between them are time invariant. Moreover, with the indirect ceiling-reflection and the 60GHz techniques [4], the channel between transceivers are indirect Line of Sight (LoS) channel without multi-path effect. These two facts help to easy the cooperative transmissions among the nodes.

With respect to *Observation 1*, it suffices to illustrate STAC for a particular multiple-access unit as depicted in Fig. 1. Suppose that the receive node 0 is only interested in the weighted summation s_0 of the K source digits s_1, s_2, \dots, s_K ,

$$s_0 = \sum_{i=1}^K w_i s_i, \quad (1)$$

where w_1, w_2, \dots, w_K are the weight coefficients, and all the quantities here are assumed to be real integers throughout this

¹This can be regarded as a simple extension of the combiner operation from the source node to the relay nodes.

paper. In STAC, the K source nodes transmit their digits in the same time slot with appropriately chosen transmit powers, frequencies, phases and times, such that their information bearing EM waves arrive at node 0 in a desired *superimposed* form that can be transformed to s_0 directly. As will be shown, this new STAC scheme significantly improves the separate P2P/SAF-based-transmission and computation strategy, in terms of bandwidth and energy efficiencies. Additionally, in the general case when node 0 needs to fully recover the original K source digits, e.g., for performing some computation other than weighted summation, one can still apply STAC by properly designing a set of *pseudo* coefficients $\{w_1, w_2, \dots, w_K\}$ such that the original digits s_1, s_2, \dots, s_K can be extracted from the received s_0 .

To enable STAC, accurate channel state information (CSI) and perfect frequency/time synchronization among the transceivers are needed, both of which may be difficult to obtain in general wireless networks. Thanks to *Observation 2*, however, the CSI in a DC is nearly time-invariant and can be accurately estimated.

To accomplish the synchronization, as another contribution, this paper novelly proposes to use wired connections among all the work nodes to provide the wireless transceivers external reference signals (e.g., a high quality external clock signal) [13], based on an enhanced Software Defined Network (SDN) architecture [14]. It should be pointed out that, the wired connections here are distinguished from the information transmission links in a wired DCN. The former are dedicated and solely responsible for control signals, not requiring the high bandwidth and random traffics as in the latter, and thus will not cause the aforementioned problems encountered by wired DCNs. We also remark that to build up such a wired control network in DCs is plausible considering that the work nodes are usually compactly piled up in a dedicated room of limited size. As a by-product, it will also reduce the DCN operation cost by eliminating the need of using individual oscillators at the transceivers.

II. MOTIVATING EXAMPLES

Two major DC applications are i) distributed file storage, e.g., GFS [8] and Hadoop Distributed File System (HDFS) [15], and ii) parallel big data processing, typically based on the MapReduce style framework [9]. We now present three detailed DC application examples mentioned in Section 1 that motivate our STAC scheme, where the first two correspond to GFS and MapReduce, respectively, and the last one shows the flexibility of STAC for general applications. Again, with task division, we can concentrate our discussions on the multiple-access unit depicted in Fig. 1.

Network Coded Storage. Due to the nonnegligible node failures in a DC [8], in distributed storage systems, a big file is usually divided into many fixed-length data blocks that are further protected by multiple replicas stored at different work nodes.

For storage efficiency, network code (or erasure code) can be applied [11], [16], [17], where each node stores the network

coded data blocks rather than their original forms. When a data block is lost due to the node failure, it can be reconstructed at a new node by performing the following algorithm digit-by-digit:

Algorithm 1 Network Coded Recovery

- 1: $s_0 = \sum_{i=1}^K w_i s_i$
 - 2: $s_0 \leftarrow s_0 \bmod 2^q$
-

where s_0 denotes a digit from the lost data block requiring recovery, s_1, s_2, \dots, s_K are digits from the data blocks stored at the other nodes, w_1, w_2, \dots, w_K are the network coding coefficients, and the modulo operation is due to the finite field size 2^q . Clearly, with STAC, we can achieve Step 1 of the algorithm directly.

MapReduce Based Data Processing. Popularized by Google, MapReduce is a dominant parallel big data processing tool in DCs. In MapReduce model, when the map nodes finish the processing, their outputs with the same key will be sent to a specified reduce node for the final computations. Such computations are also typically in the form of weighted summations [9], [18], e.g., for all machine learning algorithms fitting the statistical query model [12], scientific processes [19], [20], parallel K -means [21], prefix sum and brute-force sorting [22], documents similarity comparisons [23], etc. Again, our STAC scheme can be applied to achieve the simultaneous transmissions and computations efficiently.

General Case. In DCs, there are quite a few other applications, in which the additional task division does not applied. In such cases the receive node 0 needs the original source digits, one can appropriately design a set of pseudo coefficients $\{w_1, w_2, \dots, w_K\}$ such that the source digits s_1, s_2, \dots, s_K can be extracted from s_0 . In particular, suppose for each $i = 1, 2, \dots, K$, $0 \leq s_i \leq 2^q - 1$, then choosing $w_i = 2^{q(i-1)}$ yields

$$s_0 = \sum_{i=1}^K 2^{q(i-1)} s_i,$$

based on which all the source digits can be extracted with the following algorithm:

Algorithm 2 Source Digits Extraction

- 1: $i \leftarrow 1$
 - 2: **while** $i \leq K$ **do**
 - 3: $s_i \leftarrow s_0 \bmod 2^q$
 - 4: $s_0 \leftarrow (s_0 - s_i)/2^q$
 - 5: $i \leftarrow i + 1$
 - 6: **end while**
-

III. SYSTEM FRAMEWORK WITH STAC

A. A Basic STAC Unit

STAC is a general physical layer scheme that can be applied to wireless DCs with any structure, carrier frequency, etc.

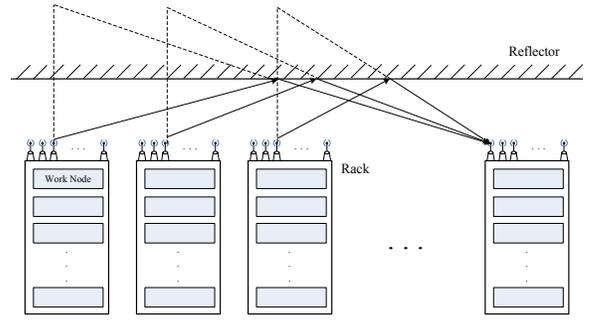


Fig. 2. A typical wireless DC layout.

For illustration, consider a typical layout of the wireless DC as shown in Fig. 2, where each rack contains multiple work nodes and has an antenna array mounted on its top to communicate with other racks (communications within a rack are accomplished with intra-rack connections) [7]. As in [4], ceiling-reflecting and 3D beamforming techniques are adopted to achieve an indirect LoS link between any two antenna arrays without causing interference to others.

Suppose K work nodes (in K different racks) need to transmit their digits s_1, s_2, \dots, s_K to node 0 for computing the weighted summation as in (1). The operating principle of STAC is illustrated in the following.

Each source node i maps its digit s_i to a baseband modulated complex symbol d_i , and then up converts the symbol d_i to a passband signal given by

$$\sqrt{P_i} e^{-j\theta_i} d_i(t) e^{-j f_c t},$$

where θ_i and $\sqrt{P_i}$ are the pre-equalizing phase and amplitude coefficients, respectively. Suppose each node i transmits at time t_i using 3D beamforming, then the received passband signal $y(t)$ can be expressed as

$$\sum_{i=1}^K h_i e^{j\theta'_i} \sqrt{P_i} e^{-j\theta_i} d_i(t - t_i - \tau_i) e^{-j f_c (t - t_i - \tau_i)} + n(t)$$

where $h_i e^{j\theta'_i}$ is the equivalent complex channel coefficient from node i to 0, τ_i is the propagation delay for node i , and $n(t)$ is a Gaussian noise of variance σ^2 for both the real and imaginary dimensions. With accurate CSI, one can set

$$\theta'_i = \theta_i \text{ and } t_i = t_0 - \tau_i, \quad (2)$$

such that the received signal simplifies to

$$y(t) = \sum_{i=1}^K h_i \sqrt{P_i} d_i(t - t_0) e^{-j f_c (t - t_0)} + n(t),$$

which, after down conversion and sampling at time $t = t_0$, yields the baseband symbol²

$$y = \sum_{i=1}^K h_i \sqrt{P_i} d_i + n. \quad (3)$$

²The h_i in (3) are real variables, so that the real and imaginary parts of symbol y can be separated. The sequel of this paper will only consider the real part for simplicity.

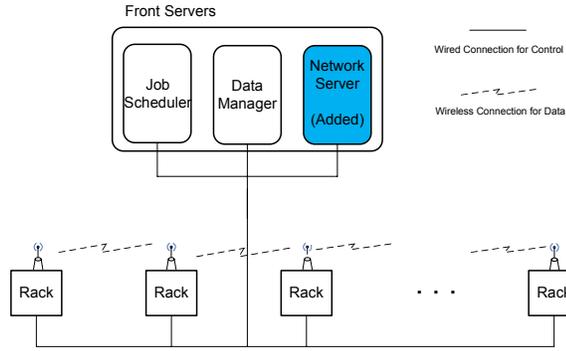


Fig. 3. An enhanced SDN architecture.

Clearly, if each node i sets

$$P_i = (w_i/h_i)^2, \quad (4)$$

then after eliminating the noise, node 0 can construct the desired digit s_0 as in (1) from the symbol y in (3).

With the above described principle, we can find that the time/frequency synchronization and pre-equalization, such as (2) and (4), are essential for our STAC. They can be realized based on an enhanced SDN architecture as shown in the next.

B. An Enhanced SDN Architecture

The DC generally works in a centralized control manner, where the front servers, including the job scheduler and data manager, manage all the work nodes. In current DCNs, control signals and data traffic share the same network. Here, we propose to use a dedicated low bandwidth wired control network with an added network server as shown in Fig. 3, based on an enhanced SDN architecture. As mentioned in Section 1, the feasibility of establishing the wired control network is endorsed by the limited DC size and the fixed node locations.

Our SDN architecture is an enhanced one in the sense that, it not only accomplishes networking control as in general SDNs, but also also provides the wireless transceivers the physical and upper layer configurations to enable STAC, including the synchronization information, the physical layer parameters such as powers, frequencies, phases and times, and the scheduling/routing information.

Synchronization with External Reference Signals. External reference signals are provided to all the transceivers for synchronization. These include a high quality external clock signal, with which individual crystal oscillators at the transceivers are no longer needed and the operation cost can be thereby reduced. These reference signals can also help calibrate the wireless transceivers, e.g., reduce the errors induced from the device hardware differences [13].

Physical Layer Parameters. The network server maintains a connection information table that stores important physical layer parameters for each connection, such as the transmission delay τ , channel coefficient $he^{-j\theta}$ and the steering vectors required for 3D beamforming. When a transceiver fails (or a

new one comes in), it informs the network server through the control network to remove (add) it from (to) the connection information table.

Scheduling/Routing. Also maintained by the network server is a table storing the scheduling/routing information. When a current task finishes or a new one needs to start, the job scheduler informs the network server to update the scheduling/routing information table, and then the network server will do the corresponding coordinations among all the work nodes involved.

IV. PHYSICAL LAYER ISSUES

A. Modulation-Demodulation Mapping

The modulation for STAC is the same as that for P2P channels. However, their demodulation mappings are subtly different: STAC demodulation maps a superimposed symbol, which may even not belong to the transmit symbol sets, to the summation of the digits, whereas the P2P channel demodulation maps a particular symbol from the transmit symbol set to the corresponding digit.

STAC Modulation. Specifically, writing node i 's digit s_i into the bit sequence form yields

$$[s_i(1), s_i(2), \dots, s_i(l), \dots, s_i(L)]$$

where $s_i(l)$ is the l -th bit, L is the sequence length, and

$$s_i = \sum_{l=0}^{L-1} 2^l s_i(l).$$

For modulation, assume BPSK (Binary Phase Shift Keying)³ without error correction coding throughout this paper. At node i , each bit $s_i(l)$ is modulated to a symbol $d_i(l) \in \{-1, +1\}$ as $d_i(l) = 1 - 2 \times s_i(l)$.

STAC Demodulation. After the l -th transmission and the removal of noise with signal detection, the received superimposed symbol can be written as

$$y(l) = \sum_{i=1}^K h_i \sqrt{P_i} d_i(l) \quad (5)$$

By setting the transmit power⁴ $P_i = (w_i/h_i)^2$, one has

$$y(l) = \sum_{i=1}^K w_i d_i(l), \quad (6)$$

which, through the operation

$$\frac{1}{2} \left(\sum_{i=1}^K w_i - y(l) \right),$$

³STAC also applies with other modulations such as QPSK, QAM, OOK, OFDM, etc. This paper only considers the simplest BPSK due to the same reason mentioned in Footnote 1.

⁴With the unit power of d_i in BPSK, the transmit power $P_i |d_i|^2$ simply equals P_i .

yields the summation $\sum_{i=1}^K w_i s_i(l)$. Finally, the desired digit can be constructed as

$$\sum_{l=0}^{L-1} 2^l \sum_{i=1}^K w_i s_i(l) = \sum_{i=1}^K w_i \sum_{l=0}^{L-1} 2^l s_i(l) = \sum_{i=1}^K w_i s_i.$$

B. Signal Detection

We now present a simple signal detection scheme for removing the noise in (3) to obtain (5), and analyze its corresponding SER (Symbol Error Rate). It suffices to consider only one of the L transmissions, and hence the index l as in the last subsection will be omitted.

Specifically, view the symbol $\sum_{i=1}^K w_i d_i$ in (6) as a point of a non-standard PAM (Pulse Amplitude Modulation) constellation that results from the weighted superposition of the transmit BPSK constellations and hence may have unequal distance between different adjacent constellation points. A simple detection scheme is to quantize the y in (3) to its nearest constellation point. Let π be a permutation on $\{1, 2, \dots, K\}$ such that $w_{\pi(j_1)} \leq w_{\pi(j_2)}, \forall j_1 \leq j_2$. We have the following theorem regarding the SER with such detection.

Theorem 1: The SER with the nearest point detection is upper bounded by

$$\text{SER}_{\text{STAC}} \leq (1 - 1/2^K) \text{erfc}(1/\sqrt{2}\sigma) \quad (7)$$

where $\text{erfc}(x) = \frac{2}{\sqrt{\pi}} \int_x^\infty e^{-t^2} dt$ is the complementary error function, σ^2 is the variance of the noise, and the equality in (7) holds when the distance between any two adjacent constellation points is equal to 2, e.g., when $w_{\pi(j)} = 2^{j-1}$ or 1, $\forall j = 1, \dots, K$.

Proof Sketch: Since w_i are all real integers, the largest SER is attained when the distance between any two adjacent constellation points is 2, which includes the case of $w_{\pi(j)} = 2^{j-1}$ or 1, $\forall j = 1, \dots, K$.

C. Performance of STAC

The performance of STAC is a tradeoff among SER, energy efficiency and bandwidth efficiency, and is clearly dependent of the weight coefficients. The *air computation* essence of STAC and its advantage over the separate strategy can be best illustrated in the ideal case of $w_1 = w_2 = \dots = w_K = 1$, where we will show that for fixed energy efficiency, STAC achieves better SER and significantly improved bandwidth efficiency.

On the other hand, to show that STAC uniformly outperforms the separate strategy, we will consider the pseudo coefficients case as mentioned in Section 2, i.e., $w_{\pi(j)} = 2^{j-1}, \forall j$. The argument here is that by applying STAC with the pseudo coefficients, one can recover the original K source digits, based on which summation with any weight coefficients can be computed. We will show that in this case, STAC achieves better energy efficiency for fixed SER and bandwidth efficiency.

1) *The Ideal Case:* Suppose $w_1 = w_2 = \dots = w_K = 1$, which is the ideal case for STAC. The SER of STAC is given in Theorem 1, i.e.,

$$\text{SER}_{\text{STAC}} = (1 - 1/2^K) \text{erfc}\left(\frac{1}{\sqrt{2}\sigma}\right).$$

Note that in this case, the resultant receive PAM constellation has only $K + 1$, instead of 2^K , points, where the decrease of the constellation size is due to the “air computation”. Or equivalently, viewed from the energy perspective, this advantage is reflected by the fact that the needed transmit power now attains the minimum $P_i = 1/h_i^2$ for each node i .

For the separate strategy, assume each node i transmits with the same power $P_i = 1/h_i^2$ as in STAC. The SER for node i is a standard result, given by $\frac{1}{2} \text{erfc}\left(\frac{1}{\sqrt{2}\sigma}\right)$. Combining all the detected K symbols, the receiver computes $\sum_{i=1}^K w_i d_i$, and the resultant SER_{SEP} is characterized in the following theorem.

Theorem 2: The SER with the separate strategy is given lower bounded by

$$\text{SER}_{\text{SEP}} \geq \frac{1}{2} - \frac{1}{2} \left(1 - \text{erfc}\left(\frac{1}{\sqrt{2}\sigma}\right)\right)^K$$

where the equality achieves when $w_1 = w_2 = \dots = w_K = 1$.

Proof Sketch: The theorem can be proved by noting that the number of erroneous symbols is a binomial random variable with parameters (K, p) , and the computation result is wrong if and only if there are odd number of erroneous symbols when $w_1 = w_2 = \dots = w_K = 1$.

Theorem 3: $\text{SER}_{\text{SEP}} > \text{SER}_{\text{STAC}}$ for any $K \geq 2$.

Proof Sketch: Use mathematical induction.

Therefore, STAC achieves a better SER and simultaneously improves the bandwidth efficiency by a factor of K . Especially, note that as $K \rightarrow \infty$, $\text{SER}_{\text{STAC}} \rightarrow \text{erfc}(1/\sqrt{2}\sigma)$ whereas $\text{SER}_{\text{SEP}} \rightarrow 1/2$.

To achieve the same bandwidth efficiency, suppose each node transmits K bits in one symbol for separated transmission. Then each node need to increase its transmit power at least by a factor 2^K , resulting an SER more than SER_{SEP} . In other words, STAC can improve the energy efficiency by a factor more than 2^K in the ideal case.

2) *Pseudo Coefficients Case:* Consider a set of pseudo coefficients $w_{\pi(j)} = 2^{j-1}, \forall j$. To minimize the total transmit power $\sum_{i=1}^K (w_i/h_i)^2$ with STAC, we allocate these coefficients among the K nodes such that $h_{\pi(j_1)} \geq h_{\pi(j_2)}, \forall j_1 \leq j_2$. Assuming STAC is completed within unit time, the total transmit energy E_{STAC} is given by

$$E_{\text{STAC}} = \sum_{j=1}^K (2^{(j-1)}/h_{\pi(j)})^2. \quad (8)$$

We now calculate the total energy needed E_{SEP} for the separate strategy assuming that each node transmits 1 bit to the receiver within $1/K$ time to maintain the same bandwidth efficiency as STAC. For the separate strategy to achieve the similar SER as STAC, the distance between any adjacent

receive constellation points also needs to be 2, in which case node i 's transmit power is given by

$$P_i = \sum_{j=1}^K (2^{(j-1)}/h_i)^2.$$

Therefore, the total energy needed is

$$E_{\text{SEP}} = \frac{1}{K} \sum_{i=1}^K \sum_{j=1}^K (2^{(j-1)}/h_i)^2 \quad (9)$$

where the factor $1/K$ accounts for the transmission time of each node.

Theorem 4: $E_{\text{SEP}} \geq E_{\text{STAC}}$, where the equality holds only when h_i are the same for all i .

Proof Sketch: The proof utilizes the important fact that $w_{\pi(j_1)} \leq w_{\pi(j_2)}$ and $h_{\pi(j_1)} \geq h_{\pi(j_2)}$, $\forall j_1 \leq j_2$.

From Theorem 4, it can be concluded that STAC performs uniformly better than the separate strategy for any set of weight coefficients. This is because even requiring STAC to fully recover the original K source digits leads to better energy efficiency than the separate strategy, for fixed SER and bandwidth efficiency.

3) *Discussion:* The above analyzes two extreme cases of the weight coefficients. In general, depending on the specific weight coefficients, one has the freedom of dividing the K nodes into M groups ($1 \leq M \leq K$), and letting each group transmit using STAC separately, to achieve a tradeoff between the bandwidth efficiency and energy efficiency.

V. CONCLUSION

The wireless DCN differs from general wireless networks in that it has large amounts of M2O sessions, which are normally followed by further computations at the destinations, with weighted summation being the typical case. Recognizing this, we have proposed a novel physical layer scheme STAC that achieves simultaneous transmissions and computations over the air, and an enhanced SDN architecture to enable it. It is demonstrated that with STAC used, both the bandwidth and energy efficiencies can be significantly improved.

REFERENCES

- [1] Mohammad Al-Fares, Alexander Loukissas, and Amin Vahdat. A scalable, commodity data center network architecture. In *Proceedings of the ACM SIGCOMM 2008 Conference on Data Communication*, SIGCOMM '08, pages 63–74, New York, NY, USA, 2008. ACM.
- [2] S. Kandula, J. Padhye, and P. Bahl. Flyways to de-congest data center networks. In *Proceedings of the ACM HotNets 2009 Conference*. ACM, 2009.
- [3] K. Ramachandran, R. Kokku, R. Mahindra, and S. Rangarajan. 60ghz data-center networking: wireless \rightarrow worryless. *Tech. Rep., NEC Laboratories America, Inc.*, July 2008.
- [4] X. Zhou, Z. Zhang, Y. Zhu, Y. Li, S. Kumar, A. Vahdat, B. Y. Zhao, and H. Zheng. Mirror mirror on the ceiling: Flexible wireless links for data centers. In *Proceedings of the ACM SIGCOMM 2012 Conference*, SIGCOMM '12, pages 443–454, New York, NY, USA, 2012. ACM.
- [5] N. Hamedazimi, Z. Qazi, H. Gupta, V. Sekar, S. R. Das, J. P. Longtin, H. Shah, and A. Tanwer. Firefly: A reconfigurable wireless data center fabric using free-space optics. In *Proceedings of the ACM SIGCOMM 2014 Conference*, SIGCOMM '14, pages 319–330, New York, NY, USA, 2014. ACM.

- [6] D. Halperin, S. Kandula, J. Padhye, P. Bahl, and D. Wetherall. Augmenting data center networks with multi-gigabit wireless links. In *Proceedings of the ACM SIGCOMM 2011 Conference*, SIGCOMM '11, pages 38–49, New York, NY, USA, 2011. ACM.
- [7] J.-Y. Shin, E. G. Sirer, H. Weatherspoon, and D. Kirovski. On the feasibility of completely wireless datacenters. *IEEE/ACM Trans. Netw.*, 21(5):1666–1679, October 2013.
- [8] Sanjay Ghemawat, Howard Gobioff, and Shun-Tak Leung. The google file system. In *Proceedings of the Nineteenth ACM Symposium on Operating Systems Principles*, SOSP '03, pages 29–43, New York, NY, USA, 2003. ACM.
- [9] Jeffrey Dean and Sanjay Ghemawat. MapReduce: Simplified data processing on large clusters. *Commun. ACM*, 51(1):107–113, January 2008.
- [10] David Nagle, Denis Serenyi, and Abbie Matthews. The panasas activescale storage cluster: Delivering scalable high bandwidth storage. In *Proceedings of the 2004 ACM/IEEE Conference on Supercomputing*, SC '04, pages 53–, Washington, DC, USA, 2004. IEEE Computer Society.
- [11] A.G. Dimakis, P.B. Godfrey, Y. Wu, M.J. Wainwright, and K. Ramchandran. Network coding for distributed storage systems. *IEEE Trans. Inf. Theory*, 56(9):4539–4551, 2010.
- [12] C.-T. Chu, S. K. Kim, Y.A. Lin, Y. Yu, G. Bradski, A. Ng, and K. Olukotun. MapReduce for machine learning on multicore. In *Proc. Neural Information Processing Systems Conference (NIPS)*, April 2006.
- [13] Clayton Shepard, Hang Yu, Narendra Anand, Erran Li, Thomas Marzetta, Richard Yang, and Lin Zhong. Argos: Practical many-antenna base stations. In *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking*, Mobicom '12, pages 53–64, New York, NY, USA, 2012. ACM.
- [14] Nick McKeown, Tom Anderson, Hari Balakrishnan, Guru Parulkar, Larry Peterson, Jennifer Rexford, Scott Shenker, and Jonathan Turner. Openflow: Enabling innovation in campus networks. *SIGCOMM Comput. Commun. Rev.*, 38(2):69–74, March 2008.
- [15] K. Shvachko, Hairong Kuang, S. Radia, and R. Chansler. The hadoop distributed file system. In *Mass Storage Systems and Technologies (MSST), 2010 IEEE 26th Symposium on*, pages 1–10, May 2010.
- [16] Y. Wu. Existence and construction of capacity-achieving network codes for distributed storage. *IEEE Journal on Selected Areas in Communications*, 28(2):277–288, February 2010.
- [17] D.S. Papailiopoulos, Jianqiang Luo, A.G. Dimakis, Cheng Huang, and Jin Li. Simple regenerating codes: Network coding for cloud storage. In *Proceedings of the IEEE INFOCOM 2012*, pages 2801–2805, March 2012.
- [18] C. Ranger, R. Raghuraman, A. Penmetsa, G. Bradski, and C. Kozyrakis. Evaluating MapReduce for multi-core and multiprocessor systems. In *Proceedings of 13th IEEE International Symposium on High Performance Computer Architecture*, 2007. HPCA 2007., pages 13–24, Feb 2007.
- [19] Sangwon Seo, E.J. Yoon, Jaehong Kim, Seongwook Jin, Jin-Soo Kim, and Seungryoul Maeng. Hama: An efficient matrix computation with the MapReduce framework. In *Proceedings of IEEE Second International Conference on Cloud Computing Technology and Science (CloudCom), 2010*, pages 721–726, Nov 2010.
- [20] Chao Liu, Hungchih Yang, Jinliang Fan, Li-Wei He, and Yi-Min Wang. Distributed nonnegative matrix factorization for web-scale dyadic data analysis on Mapreduce. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, pages 681–690, New York, NY, USA, 2010. ACM.
- [21] Weizhong Zhao, Huifang Ma, and Qing He. Parallel k-means clustering based on mapreduce. In *Cloud Computing*, pages 674–679. Springer, 2009.
- [22] M. T. Goodrich, N. Sitchinava, and Q. Zhang. Sorting, searching, and simulation in the mapreduce framework. In *Proceedings of the ISAAC*, pages 374–383. Springer, December 2011.
- [23] Tamer Elsayed, Jimmy Lin, and Douglas W. Oard. Pairwise document similarity in large collections with MapReduce. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers*, HLT-Short '08, pages 265–268, Stroudsburg, PA, USA, 2008. Association for Computational Linguistics.