1

Communication Complexity and Data Compression

Ulrich Tamm

***Abstract*— A result of Ahlswede and Cai for the 2-party communication complexity of set intersection is generalized to a multiparty model. There are relations to several areas as to the direct-sum conjecture and amortized complexity in computational complexity or interactive communication in information theory as well as to wireless sensor networks and even quantum communication. The aim of the paper is mostly to survey these different applications and to draw the attention of researchers in one area to the results and applications in other areas.**

***Index Terms*— prefix codes, communication complexity, amortized complexity, functions on direct sums**

I. Introduction

The notion of communication complexity was introduced by Yao in 1979 [26]. Since then it found many applications in Computer Science, e. g. [16]. The communication complexity of a function $f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ (where $\mathcal{X}$, $\mathcal{Y}$, and $\mathcal{Z}$ are finite sets), denoted as $C(f)$, is the number of bits that two persons, P_1 and P_2, have to exchange in order to compute the function value $f(x, y)$, when initially P_1 only knows $x \in \mathcal{X}$ and P_2 only knows $y \in \mathcal{Y}$. To this aim they follow a predetermined interactive protocol.

In his pioneering paper Yao [26] used an extra stop symbol to announce the end of a message in such an interactive protocol. Papadimitriou and Sipser [21] got rid of this extra symbol by allowing only prefix codes. Since then no possible message is the beginning of another one the end of a message is excatly determined. However, prefix codes may also serve to compress the amount of communication and a very remarkable compression was achieved by Ahlswede and Cai [1] (cf. also [2]) for set intersection.

To this aim they considered vector-valued functions f^n defined on the direct sums $\mathcal{X}^n, \mathcal{Y}^n$ of the sets from the domain of some basic function $f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$. For $x^n = (x_1, \ldots, x_n)$ f and $y^n = (y_1, \ldots, y_n)$ then

$$f^n(x^n, y^n) = \big(f(x_1, y_1), \ldots, f(x_n, y_n)\big)$$

The set intersection $si^n(x^n, y^n)$ then arises for the Boolean "and" as basic function si. Obviously, then si^n yields the intersection of the two sets represented by the binary strings x^n and y^n. In [1] it was shown that

$$c(si^n) = \lceil n \cdot \log_2(3) \rceil$$

Originally Ahlswede's research in this direction was motivated by studying the communication complexity of the Hamming distance [4] and similar functions defined on direct sums, e.g. [3] and [24]. He was mainly interested in a single–letter characterization basing the communication complexity of the direct sum function on the communication complexity of the basic function f. To this aim he and his coauthors demonstrated that several lower bounds behave multiplicatively. We shall sketch these bounds and also the protocol for set intersection in Section II.

A further line of research leading to direct sum methods in communication complexity goes back to the question if it is easier to solve communication problems simultaneously than separately, cf. [16], pp. 42 - 48. An obvious upper bound on the communication complexity $C(f^n)$ is obtained by evaluating each component $f(x_i, y_i)$ separately and communicating the result for component i using the optimal protocol for f. Can we do better by considering all components simultaneously? With the result of Ahlswede and Cai this is possible for set intersection, since $C(si) = 2$ but $C(si^n) = \lceil n \cdot \log_2 3 \rceil$.

The measure $\limsup_{n\to\infty} \frac{1}{n} C(f^n)$ is also called amortized communication complexity (see [12]). One of the main open problems in communication complexity is the question if there can exist a significant gap between the communication complexity and the amortized communication complexity of a function. We shall review the discussion in Section III.

This problem was recently extended to the "number in hand" model of multiparty communication complexity [11]. In Section IV we shall consider the set intersection function s^n for more than 2 sets and show that the results of Ahlswede and Cai [1] can be generalized to this case. Namely, for k parties involved in the communication

$$\lceil n \cdot log_2(k+1) \rceil \leq C(s^n) = \lceil n \cdot log_2(k+1) \rceil + k - 2$$

Interestingly, the"number in hand model" in the beginning was not so popular but later found an important application in streaming [5]. Especially, the compression for set intersection has an application in wireless sensor networks as discussed in [14], where even more general threshold functions were considered.

II. Communication Complexity of Set Intersection

For set intersection si^n the naive protocol, in which one person sends all the bits of his input and the other person returns the result, yields the upper bound $C(si^n) \leq 2n$ on the communication complexity.

A lower bound is obtained via the rank of the function value matrices $M_z(f) = (a_{xy})_{x\in\mathcal{X}, y\in\mathcal{Y}}$ for all $z \in \mathcal{Z}$ defined by $a_{xy} = \begin{cases} 1 & \text{if } f(x,y) = z \\ 0 & \text{if } f(x,y) \neq z. \end{cases}$

Namely, for any function f

U. Tamm is with the Department of Economy, University of Applied Sciences, Bielefeld, Germany on leave from the German Language Department of Business Informatics, Marmara University, Istanbul, (email tamm@ieee.org)

$$C(f) \geq \lceil \log_2 r(f) \rceil, \text{ where } r(f) = \sum_{z \in \mathcal{Z}} \text{rank} M_z(f)$$

In [1] it was shown that the parameter $r(f)$ behaves multiplicatively, i.e., for any vector-valued function f^n with basic function f it holds

$$r(f^n) = r(f)^n$$

For the basic function si (the logical "and") then

$$r(si) = \text{rank} \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} + \text{rank} \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} = 2 + 1 = 3$$

The communication complexity of si hence is 2 bits, since the naive protocol requires 2 bits of communication and the lower bound is also $C(si) \lceil \geq \log_2(r(f)) \rceil = \lceil \log_2(3) \rceil = 2$.

However, for the vector-valued function, there is a gap between the upper and lower bound, since the naive protocol now requires $2n$ bits of communication, whereas the lower bound is

$$C(si^n) \geq \lceil \log_2 r(si^n) \rceil = \lceil \log_2(3^n) \rceil = \lceil n \cdot \log_2(3) \rceil$$

Via the use of prefix codes and Kraft's inequality it is possible to close this gap.

Theorem 1 ([1], [2]):

$$C(si^n) = \lceil \log_2(3) \rceil$$

Proof: The naive protocol is modified as follows. In knowledge of x^n the set of possible function values is reduced to the set $S(x^n) = \{y^n : y^n \subset x^n\}$. Hence, only $\lceil \log_2 S(x^n) \rceil$ bits have to be reserved for the transmission of $si^n(x^n, y^n)$ such that P_1 can assign longer messages to elements with few subsets. So, in contrast to the naive protocol, the messages $\{\phi_1(x^n) : x^n \in \{0,1\}^n\}$ are now of variable length. Since the prefix property has to be guaranteed, Kraft's inequality for prefix codes yields a condition, from which the upper bound can be derived. Specifically, we require that to each x^n there corresponds a message $\phi_1(x^n)$ of (variable) length $l(x^n)$ such that for all $x^n \in \{0,1\}^n$ the sum $l(x^n) + \lceil \log_2 S(x^n) \rceil$ takes a fixed value, L say. Kraft's inequality states that a prefix code exists, if $\sum_{x^n} 2^{-l(x^n)} \leq 1$. This is equivalent to $\sum_{x^n} 2^{\lceil \log_2 S(x^n) \rceil} \leq 2^L$. With the choice $L = \lceil \log_2(3^n) \rceil$ Kraft's inequality holds.

Thus, an efficient protocol was obtained via data compression, namely the appropriate encoding of the messages in the naive protocol. This can be interpreted as a compression of this protocol.

III. Amortized Communication Complexity and the Direct–Sum Conjecture

Direct sum methods in communication complexity are useful tools in separating complexity classes. Further applications are the comparison of lower bound techniques and the study of their power - e. g., how large can be the gap between the lower bound and the communication complexity [19]. The intuition is that small gaps for the basic function f become large for the vector-valued function f^n.

Karchmer, Raz, and Wigderson [17] asked how much better simultaneous computations are compared to the componentwise evaluation of the function f^n for basic Boolean functions $f : \{0,1\}^m \times \{0,1\}^m \to \{0,1\}$. They conjectured that the *amortized communication complexity*

$$\overline{C}(f) = \frac{1}{n} \limsup_{n \to \infty} C(f^n)$$

is close to $C(f)$ – the communication complexity of the basic function f.

As Karchmer, Raz, and Wigderson [17] point out, a proof of their direct sum conjecture would be a decisive step towards a separation of the complexity classes NC^1 and NC^2 - a long outstanding open problem in computer science. Actually, they considered the communication complexity of relations. For functions this concept was discussed in [12].

In [6] the notion of closeness in the above conjecture was defined more formally. Namely the direct sum conjecture in [6] was stated as

$$C(f^n) = n \cdot (C(f) - O(1))$$

There is some evidence against it, since Naor, Orlitsky, and Shor [18] presented a partial function (not defined for every input - this concept is motivated from interactive communication [20]) with deterministic communication complexity $C(f) = \Theta(\log(m))$ but amortized complexity $O(1)$

Observe that the amortized communication complexity is just the limit for $n \longrightarrow \infty$ of the communication complexity of the vector–valued function divided by the number of components n. Hence, with Theorem 1 the function si^n can be evaluated much faster considering all n components simultaneously than by componentwise communication of the results for the basic function si, which would cost $2n$ bits. So the amortized communication complexity of the function si is $\frac{1}{n} \lim_{n \to \infty} C(si^n) = \log_2(3)$. Of course, the difference $C(si) - \overline{C}(si) = 2 - \log_2(3)$ is too small in order to disprove the direct sum conjecture. However, compression of a protocol leads to a significant improvement.

Much interesting for information theory is the application of protocol compression in the analysis of probabilistic and randomized protocols, e. g., [15]. Here methods as Slepian/Wolf coding [9], interactive communication [8] or common information and common randomness [13] come into play in the analysis of corresponding direct – sum theorems.

IV. Set Intersection for Multiparty Communication

The set intersection function s^n in $k > 2$ arguments has as basic function s the logical "and" of k binary inputs. Thus s is defined on $\{0,1\} \times \{0,1\} \times \ldots \times \{0,1\}$ via:

$$s(x_1, x_2, \ldots, x_k) = \begin{cases} 1 & , x_1 = x_2 = \ldots = x_k = 1 \\ 0 & , \text{ else} \end{cases}$$

The problem in generalizing Theorem 1 is that for more than 2 parties communicating the rank lower bound is not

applicable any more. Luckily, generalizing a result of [1] it can also be shown that the independence number behaves multiplicatively for vector-valued functions, i.e., $Ind(f^n) \geq Ind(f)^n$.

Here $Ind(f) = \sum_z ind_z(f)$, where $ind_z(f)$ is the minimal number of disjoint monochromatic rectangles needed to include all the values z in the function matrix. A monochromatic rectangle is a subset $A_1 \times \ldots \times A_k$ on which the function f^n takes the constant value z.

Theorem 2: ([14], [25])

$$\lceil n \cdot \log_2(k+1) \rceil \leq C(t^n) \leq \lceil n \cdot \log_2(k+1) \rceil + k - 2$$

Proof: The function tensor of the basic function s contains exactly one entry 1 namely for $x_1 = x_2 = \ldots = x_k = 1$, i.e., the all–1 vector of length k . All other entries are 0. The k neighbours of the all–1 vector, i.e. all $(x_1, \ldots, x_k)$ with exactly one $x_i = 0$ and all other $x_j = 1$ obviously must be contained in different monochromatic rectangles. Since also the all-1 vector must be contained in a separate monochromatic rectangle, the independence number $Ind(s) = k+1$ and hence $C(s^n) \geq \lceil n \log_2 Ind(s) \rceil = \lceil n \log_2(k+1) \rceil$.

A protocol that almost achieves this lower bound is again obtained by assigning an appropriate prefix code to the messages in the trivial protocol. As for the set intersection function si^n in two arguments, again Person 1 can assign longer messages to inputs with few 1s. The other persons then can determine the exact value following an optimal protocol for set intersection of $k-1$ sets. For $k = 2$ we already know that $\lceil n \cdot \log_2(3) \rceil$ bits are optimal. So, for $k = 3$, Person 1 transmits $l(x)$ bits, say for an input x. Since the total number of bits transmitted should be a fixed value L, say, $L = l(x) + f(x)$, where $f(x)$ is the number of bits the other persons should still transmit to agree on the result. In order to guarantee the existence of a prefix code, Kraft's inequality $\sum_x 2^{-l(x)} \leq 1$ must hold. This is equivalent to $\sum_x 2^{-(L-f(x))} \leq 1$ or $\sum_x 2^{f(x)} \leq 2^L$. Now if Person 1 has an input $x = x_1$ with exactly i many 1's then by the protocol for si we know already that $f(x) = \lceil i \cdot \log_2(3) \rceil$ bits are enough to determine the set intersection of the remaining two sets by persons 2 and 3. So, Kraft's inequality reduces to $\sum_i \binom{n}{i} 2^{\lceil i \log_2(3) \rceil} \leq 2^L$. This can be assured by the choice $L = \lceil n \log_2(4) \rceil + 1$. Analogously, for $k > 3$ we inductively obtain from Kraft's inequality $\sum_i \binom{n}{i} 2^{\lceil i \cdot \log_2(k) \rceil + k - 3} \leq 2^L$, which is fulfilled for $L = \lceil n \cdot \log_2(k+1) \rceil + k - 2$.

The research in [25] was motivated by a recent extension of the direct–sum conjecture to the "number in hand" model of multiparty communication complexity [11]. Yao's model of communication complexity can be generalized to several multiparty models depending on the information accessible to each person. Most well studied is the "number on the forehead" model in which each person knows all inputs but her own, for instance [7] or [22]. The "number in hand" model, in which each person knows just her own input, was not so popular in the beginning but later found an important application in streaming [5].

So it was natural to extend set interesection and other functions defined over the binary alphabet to more than 2 arguments. The compression for set intersection also was helpful in the study of several more functions. Whereas the direct – sum conjecture states that a significant reduction of communication cannot be expected by increasing the length n of the inputs, set intersection demonstrates that the amount of communication can significantly be reduced, when the number k of communicators is increased.

This has an application in wireless sensor networks. Observe that the naive protocol would require $n \cdot k$ bits of communication among the k communicators, whereas the compressed protocol above requires only $n \cdot \log_2(k+1)$ bits. Kowshik and Kumar in [14] were interested in this reduction of communication and derived a more general result for threshold functions, where set intersection occurs as a special case. Thus, instead of calculating the basic function at every time instance separately, the sensor network can save energy by collecting information about n time instances and then follow the compressed protocol.

The problem with "number in hand" is that a generalization of the lower bound techniques is rather difficult. The most powerful lower bound in two-party communication complexity is the rank lower bound. But the rank of a matrix is generalized by a tensor rank (3 and higher dimensional matrices), which is not so easy to determine. Besides, the matrix rank is multiplicative under the tensor product (very important for functions on direct sums). This is no longer the case for higher dimensional tensors - a serious problem also in quantum information theory, cf. [10]. Over small alphabets a lower bound can be derived via the independence number as in the proof of Theorem 2. However, for functions defined over larger alphabets, this number is not easy to determine.

V. Concluding Remarks

A communication protocol was presented for the computation of the intersection of k subsets of an n–elementary set represented by their characteristic vectors as binary strings of length n. Via prefix coding communication can be reduced from $n \cdot k$ bits to $n \cdot \log_2(k+1)$ bits. As a consequence the amortized complexity of the logical "and" of k binary inputs is $\log_2(k+1)$, which is much better than its communication complexity k.

The underlying multiparty communication model here was "number in hand". As an application, computation of the logical "and" can be carried out more efficiently in collocated wireless sensor networks [14].

Message compression in protocols also is a useful tool in probalistic communication complexity for functions defined on direct sums. For deterministic communication the direct – sum conjecture relating the communication complexity and the amortized communication complexity of a function would yield an important separation result in computational complexity [17].

REFERENCES

[1] R. Ahlswede and N. Cai, "On Communication complexity of vector-valued functions", *IEEE Trans. Inform. Theory*, vol. 40 (6), pp. 2062 - 2067, 1994.

[2] R. Ahlswede, N. Cai, and U. Tamm, "Communication complexity in lattices", *Appl. Math. Letters*, vol. 6 (6), pp. 53–58, 1993.

[3] R. Ahlswede, N. Cai, and Z. Zhang, "A general 4–word–inequality with consequences for 2–way communication complexity", *Advances in Applied Mathematics*, vol. 10, pp. 75–94, 1989.

[4] R. Ahlswede, A. El Gamal, and K.F. Pang, "A two–family extremal problem in Hamming space", *Discr. Math.* vol. 49, pp. 1–5, 1984.

[5] N. Alon, M. Matias, and M. Szegedy, "The space complexity of approximating the frequency moments", *J. Comput. Syst. Sci.*, vol. 58 (1), pp. 137–147, 1999.

[6] A. Ambainis, H. Buhrmann, W. Gasarch, B. Kalyanasundaram, and L. Torenvliet, "The communication complexity of enumeration, elimination, and selection", *J. Comput. System Sci.*, vol. 63, 148 – 185, 2001.

[7] L. Babai, P. Frankl, and J. Simon, "Complexity classes in communication complexity theory", *Proc. IEEE FOCS*, pp. 337-347, 1986.

[8] B. Barak, M. Braverman, X. Chen, and A. Rao, "How to compress interactive communication", *Proc. ACM STOC*, 2010.

[9] M. Braverman and A. Rao, "Information equals amortized communication", *Proc. FOCS*, pp. 748 – 757, 2011.

[10] L. Chen, E. Chitambar, R. Duan, Z. Ji, and A. Winter, A, "Tensor rank and stochastic entaglement catalysis for multipartite pure states", *Physical Review Letters*, vol. 105, 2010.

[11] J. Draisma, E. Kushilevitz, and E. Weinreb, E, "Partition arguments in multiparty communication complexity", *Theoretical Computer Science*, vol. 412, pp. 2611–2622, 2011.

[12] T. Feder, E. Kushilevitz, M. Naor, and N. Nisan, "Amortized communication complexity", *SIAM J. Comp.*, vol. 24 (4), pp. 736 - 750, 1995.

[13] P. Harsha, R. Jain, D.A.McAllester, and J. Radhakrischnan, " The communication complexity of correlation", *IEEE Trans. Inform. Theory*, vol. 56, pp. 438 – 449, 2009.

[14] H. Kowshik, and P.R. Kumar, "Optimal strategies for computing symmetric Boolean functions in collocated networks", *Proc. IEEE Inf. Theory Workshop, Cairo*, 2010.

[15] R. Jain, J. Radhakrishnan, and P. Sen, "A direct sum theorem in communication complexity via message compression", pp. 300 – 315 in J.C.M. Baeten, J.K. Lenstra, J. Parrow, and G.J. Woeginger (eds.), *ICALP*, Springer Lecture Notes in Computer Science, vol. 2719, 2003.

[16] E. Kushilevitz and N. Nisan, *Communication Complexity*, Cambridge University Press, 1997.

[17] M. Karchmer, R. Raz, and A. Wigderson, "Super-logarithmic depth lower bounds via direct sum methods in communication complexity", *Proc. 6th IEEE Structure in Complexity Theory*, 1991, 299 - 304

[18] M. Naor, A. Orlitsky, and P. Shor, "Three results on interactive communication", *IEEE Trans. Inform. Theory*, vol. 39 (5), pp. 1608 - 1615, 1993.

[19] N. Nisan and A. Wigderson, "On rank vs. communication complexity", *Combinatorica*, vol. 15 (4), pp. 557-566, 1995.

[20] A. Orlitsky, "Worst case interactive communication I: two messages are almost optimal", *IEEE Trans. Inform. Theory*, vol. 36, pp. 1111-1126, 1990.

[21] C. Papadimitriou and M. Sipser, "Communication complexity", *Proc. ACM STOC*, pp. 260 – 269, 1982.

[22] A.A. Sherstov, "The multiparty communication complexity of set disjointness", *Electronic Colloquium on Computational Complexity*, Report No. 145, 2011.

[23] U. Tamm, "Deterministic communication complexity of set intersection", *Discr. Appl. Math.* vol. 61, pp. 271 - 283, 1995.

[24] U. Tamm, "Communication complexity of functions on direct sums", pp. 589–602 in: I. Althöfer, N. Cai, G. Dueck, L. Khachatrian, M. Pinsker, A. Sárközy, I. Wegener, and Z. Zhang (eds.), *Numbers, Information and Complexity*, Kluwer, 2000.

[25] U. Tamm, "Multiparty communication complexity of vector-valued and sum-type functions", pp. 451 – 462 in: H. Aydinian, F. Cicalese and C. Deppe eds., *Information Theory, Combinatorics, and Search Theory*, Springer Lecture Notes in Computer Science, 2013.

[26] A.C. Yao, "Some complexity questions related to distributive computing", *Proc. ACM STOC*, pp. 209–213, 1979.