ISSN 0032-9460, Problems of Information Transmission, 2018, Vol. 54, No. 2, pp. 191–198. © Pleiades Publishing, Inc., 2018.
Original Russian Text © B.Ya. Ryabko, 2018, published in Problemy Peredachi Informatsii, 2018, Vol. 54, No. 2, pp. 103–111.

COMMUNICATION NETWORK THEORY

Information-Theoretic Approach to Estimating the Capacity of Distributed Memory Systems[1]

B. Ya. Ryabko[a,b]
[a] *Institute of Computational Technologies, Siberian Branch of the Russian Academy of Sciences, Novosibirsk, Russia*
[b] *Novosibirsk State University, Novosibirsk, Russia*
e-mail: boris@ryabko.net

Received July 27, 2017; in final form, December 8, 2017

Abstract—Systems with cash memory (or more generally, with distributed memory) are very widely used in information technologies. Such are content delivery networks (CDN) of various types, which deliver digital movies, books, and similar content; peer-to-peer (P2P) networks, where millions of members exchange various information; and many other systems and devices of this kind. We introduce the notions of capacity and entropy efficiency for distributed memory systems, propose methods for estimating these quantities, and give an example of their application.

DOI: 10.1134/S0032946018020072

1. INTRODUCTION

Distributed memory systems appeared, apparently, with appearance of computers; in the very first models of the latter there were several memory types (or storage units): the so-called random access memory (RAM), magnetic storage, and magnetic tape. This diversity was due to the fact that different memory types differed in reading and recording rates, as well as in storage capacity, or space: the higher the rate, the less the space.

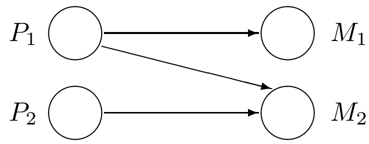
At present, there are a huge number of distributed memory systems, starting from cell phones, where several levels of the so-called cash memory are used, and ending with info-telecommunication networks of various purposes, among which we mention, as examples, CDN (content delivery network) and P2P (peer-to-peer) networks, used in particular for distribution of movies, books, and other content of this kind. In CDN networks there is usually a central server, where all available data are stored (for instance, all movies belonging to this company), and also "mirror," or local servers, storing the most popular (called-for) content.

When a user requests for a movie, first an inquiry is made whether it is present at the nearest local server, and only if this is not so, the request is sent to the central server. The purpose of such a structure is clear: data transfer from a closely located mirror server takes much less time than transmission from the central server. As a result, the average delivery time reduces.

As is seen from the above examples, distributed memory systems are used very extensively; numerous companies develop and exploit them. This, in turn, attracts many researchers to study questions of constructing and optimizing distributed memory systems. Among numerous works devoted to this subject, we note papers [1, 2] on distributed memory in networks.

It is important to note that, despite numerous works devoted to the analysis of distributed memory systems, many fundamental problems in this field remain unsolved. For example, there

[1] Supported in part by the Russian Foundation for Basic Research, project no. 15-07-01851.



Example of a distributed memory network: M_1 and M_2 are memory units; P_1 and P_2 are reading/recording devices (processors).

are no methods to estimate theoretically (analytically) the capacity and efficiency of concrete distributed memory systems. In particular, it is not possible to estimate theoretically (analytically) the impact of changes in memory size for various memory types on the network capacity, which, of course, substantially hinders the development and design processes. It should be noted that before the paper [3] appeared, the same situation was with computer designing: the impact of changing some parameters (in particular, characteristics of cash memory) was only estimated experimentally, which, of course, considerably complicated the process of developing new models.

In the present paper we propose a new approach to estimating the performance of content delivery networks with distributed memory (CDN, P2P, etc.). In the framework of this approach, we introduce the notions of capacity and entropy efficiency of a distributed memory network, which are close to classical information-theoretic notions of the channel capacity and channel transmission rate introduced by Shannon in his seminal paper [4]. Note also that in [3, 5] this approach was used to estimate computing capacity and efficiency of computers, which for the first time allowed to estimate these characteristics for real computers theoretically (analytically) [5] and even use them to predict changes in future models [6]. In particular, methods proposed there allow also to estimate the impact of different memory types on the computer capacity; therefore, we do not consider this problem here.

The notions and methods proposed in the present paper are actually carrying over the ideas and results of information theory to a new area, systems with distributed memory. Such an extension is, generally speaking, quite natural, since this area comprises in particular various data transmission systems, and information theory, after all, was designed to describe and analyze communication systems. However, to the best of the author's knowledge, such an approach was never used before, though various aspects of distributed memory network operation were studied in numerous works based on teletraffic theory, graph theory methods, and many other mathematical disciplines.

The paper is organized as follows. Section 2 is devoted to the description of systems and models in question. The main results of the paper are described in Section 3. Section 4 gives an example of using the proposed method. All proofs are postponed to the Appendix.

2. DESCRIPTION OF DISTRIBUTED MEMORY SYSTEMS AND CONSTRUCTION OF A MATHEMATICAL MODEL

We will represented distributed memory systems as directed graphs with vertices corresponding to "data storages" (in what follows, a "memory unit," or simply "memory") and reading/recording devices (see the figure). Below we will refer to them as processors, though they can be separate computers or even local networks. Note that in CDN networks they are sometimes called "points of presence"; in other networks, other terms are used.

An edge drawn from a processor P to memory M means that P has access to M. In the figure, the first processor has access to both M_1 and M_2, and the second, to M_2 only. Note that there exist networks where a vertex may function as both a processor and memory simultaneously, as in P2P networks. We will assume that a system contains n processors and m memory units.

Now we describe the content of the memory. Each memory unit stores files (i.e., binary words of the same size). Data are transmitted in the network in "batches" (i.e., parts of a certain fixed size), and we will assume that one batch contains one file. Note that if a movie or another content is of large enough size, it is transmitted as several batches, and it will be convenient to assume that it is also stored as several files corresponding to these batches.

Now let us discuss regimes of data usage. In CDN networks, two operation phases are usually distinguished, which somewhat conventionally correspond to daytime and nighttime: period when users only receive the requested data (say movies) and network channels are busy with transmitting these data, and another period when the intensity of requests for reading the content is rather low and data are moved in the network according to a certain algorithm (usually, frequently requested content is sent to mirror servers to replace rarely requested data). The goal of network designers is to achieve the maximum capacity in "daytime," i.e., in the period when the network serves users which receive data. In P2P networks, the situation is similar: each user is interested in minimizing the time required to receive the desired data, i.e., time required for reading data but not recording (or time required to receive batches but not send).

Now we introduce necessary notation: the set of all files stored in memory units will be denoted by F; let $\tau_{i,j}(f)$ be the time required for reading (receiving) a file $f \in F$ by a processor P_i from memory unit M_j; we will assume that this time is infinite if M_j does not contain f. For a network with m memory units and n processors and for a file $f \in F$, define

$$\tau_i(f) = \min_{j=1,\ldots,m} \tau_{i,j}(f), \qquad \tau_{\min} = \min_{i=1,\ldots,n} \tau_i(f), \qquad \tau_{\max} = \max_{i=1,\ldots,n} \tau_i(f); \tag{1}$$

we will consider systems for which

$$0 < \tau_{\min}, \qquad \tau_{\max} < \infty. \tag{2}$$

In many networks, each processor reads a required file from the memory unit for which the reading time is minimal. Hence, for a processor P_i and a sequence of files $f_1, f_2, \ldots, f_k$ we define the reading time as

$$\tau_i(f_1, f_2, \ldots, f_k) = \sum_{j=1}^{k} \tau_i(f_j). \tag{3}$$

Note that this is not the only way to choose a memory unit to read data from. For example, a memory unit can be chosen which will use the cheapest channel to transmit the data. It is important for us that for each file f and each processor i the quantity $\tau_i(f)$ is defined in some way and inequalities (2) are satisfied for it.

Thus, we consider systems with several processors and memory units (data storages), each of them containing files, i.e., words of the same length. Processors get requests for files, which are then received from memory units with lowest delivery time (see (1)). (Usually this is organized as follows: processors have lists of files stored in different memory devices). We assume that processors are maximally loaded, i.e., having read a file, a processor immediately starts reading the next.

3. CAPACITY AND ENTROPY EFFICIENCY

Now let us define our central notions, the capacity of a system and entropy efficiency. For that, let us first denote by F^∞ the set of all doubly infinite words in an alphabet F of the form $\ldots f_{-2} f_{-1} f_0 f_1 f_2 \ldots$; any its subset S will be called, by analogy with information theory, a source, not of symbols but of files. Define a translation operation on F^∞ as

$$\sigma(\ldots f_{-2} f_{-1} f_0 f_1 f_2 \ldots) = \ldots f_{-3} f_{-2} f_{-1} f_0 f_1 \ldots;$$

we say that a source S is stationary if

$$\sigma(S) = S. \tag{4}$$

We still assume that there are n processors in the system, the input of the ith is a message $f_1 f_2 \ldots$ generated by a source S_i, $i = 1, \ldots, n$, and for each processor i define the quantity

$$\mathcal{N}_i(t) = \{f_1 \ldots f_k : \tau_i(f_1 \ldots f_k) \geq t,\ \tau_i(f_1 \ldots f_{k-1}) < t,\ f_1 \ldots f_k \in S_i \cap F^k\}. \tag{5}$$

In other words, $\mathcal{N}_i(t)$ contains sequences $f_1 \ldots f_k$ generated by S_i that can be received by processor P_i in time at least t. The quantities

$$\mathcal{J}_i = \liminf_{t\to\infty} \frac{\log |\mathcal{N}_i(t)|}{t}, \qquad \mathcal{J} = \sum_{j=1}^{n} \mathcal{J}_j \tag{6}$$

will be called the processor capacity of P_i and the system capacity, respectively.

Let us briefly discuss the definition of the processor capacity. Informally, $|\mathcal{N}_i(t)|$ is the number of possible sequences of requests for reading files that a processor can process in time t. Recall that this quantity is analogous to the information-theoretic number of messages that can be transmitted in a given time over a noiseless channel, as was defined by Shannon (see [4]). In turn, the processor capacity $\mathcal{J}_i$ is completely analogous to the capacity of a noiseless channel, and both capacities are asymptotically equal to the argument of the exponent specifying the growth of the number of symbol chains generated by a source.

The above model does not take into account the fact that different files can be requested with different probabilities, which are not controlled by system designers (say the probability of requesting a movie depends on its popularity). To model such situations, we will consider a probability-theoretic system model. More precisely, we assume that a flow of requests for files is described by an n-dimensional stationary process $\mu_1, \mu_2, \ldots, \mu_n$ generating symbols of the alphabet F^n (here, as above, n is the number of processors). For a (one-dimensional) process λ, its kth-order entropy ($k \geq 0$) and limiting entropy are defined as

$$h_k(\lambda) = -\sum_{u\in F^{k+1}} \lambda(u)\log\lambda(u)/(k+1), \qquad h_\infty(\lambda) = \lim_{n\to\infty} h_n(\lambda). \tag{7}$$

It is known that for stationary λ the limit exists and

$$h_k(\lambda) \geq h_{k+1}(\lambda) \tag{8}$$

(see [7]). (Hereinafter, we use the same notation for a random process and its finite-dimensional distribution, since it is always clear from the context what we are speaking about.)

We estimate the efficiency of the ith processor and of the system in the whole by the entropy of the arrival flow of files per unit time:

$$\mathcal{H}(\mu_i) = h_\infty(\mu_i) \Big/ \left(\sum_{f\in F} \mu_i(f)\tau_i(f)\right), \qquad \mathcal{H} = \sum_{i=1}^{n} \mathcal{H}(\mu_i). \tag{9}$$

We call this quantity the entropy efficiency. Its information-theoretic meaning is clear: if a flow of requested files is subject to distribution μ, then the number of messages that can be transmitted in time t grows exponentially (as a function of t), and the entropy efficiency equals the argument of this exponent.

The main properties of the introduced quantities are as follows.

Theorem. *Let us be given a distributed memory system with n processors and m memory units.*

(i) *If the i-th processor is fed by a sequence of files generated by a stationary source (in the sense of Definition (4)), then for the capacity of this processor we have*

$$\mathcal{J}_i = \lim_{t\to\infty} \frac{\log |\mathcal{N}_i(t)|}{t} \tag{10}$$

(i.e., the limit inferior in (6) equals the "ordinary" limit) and

$$\mathcal{J}_i \le \log \mathcal{Z}_i, \tag{11}$$

where $\mathcal{Z}_i$ is a real solution of the equation

$$\sum_{f\in F} z^{-\tau_i(f)} = 1; \tag{12}$$

moreover, equality in (11) is attained for the source consisting of all possible sequences, i.e., $S_i = F^\infty$.

(ii) *If the i-th processor is fed by a sequence of files generated by a stationary source μ_i, then the entropy efficiency satisfies the inequality*

$$\mathcal{H}(\mu_i) \le \log \mathcal{Z}_i, \tag{13}$$

which holds with equality in the case where symbols $f_1 f_2 \ldots$ generated by μ_i are i.i.d. and

$$\mu_i(f) = \mathcal{Z}_i^{-\tau_i(f)}. \tag{14}$$

The proof of the theorem is based on well-known results from information theory and is given in the Appendix; here we note that the first claim is analogous to Shannon's theorem on the capacity of a noiseless channel (see [4]), and using equation (12) to estimate the capacity was also proposed by Shannon [4]. The second claim of the theorem is a direct analog of the relation between the topological entropy and Shannon entropy, well known in theory of dynamic systems: the former equals the maximum value of the latter (see [8, 9]).

4. EXAMPLE OF FINDING THE EFFICIENCY OF A CONTENT DELIVERY NETWORK

The example considered here is aimed at showing the possibilities of practical application of our method. For that, we consider a content delivery network (CDN) with characteristics given in Tables 1 and 2.

Each access point is linked with the main server, where all files are stored, and with one cash server, each of which stores copies of file parts. Transmission rates of channels connecting different access points with cash servers can be different, as is shown in Table 2.

Assume that requests for reading files are independent and identically distributed. Concerning the access probabilities we assume that they obey the so-called Zipf distribution, which well describes real-life processes of such kind. This means that if files are ordered with respect to their access probabilities, then the ith probability q_i is given by $q_i = (1/i) \Big/ \Big(\sum_{j=1}^{N} 1/j \Big)$, where N is the total number of files (see [10]).

Then we computed the entropy efficiency for the described system and for several its modifications to numerically estimate their impact. We assumed here that cash servers store the most called-for files, i.e., those with highest probabilities. Computations are presented in Tables 3 and 4.

Table 1. Parameters of the CDN network.

Parameter	Value
Number of memory units (cash servers)	100
Number of access points	1000
Storage capacity of a cash server	10 TB
Number of different files available in the network	5000
Content file size	10 GB

Table 2. Channel characteristics in the CDN model.

Channel function	Transmission rate	Number of channels
Access point (AP) – Cash server	10 Mbps	200 (20%)
	5 Mbps	600 (60%)
	2 Mbps	200 (20%)
AP – Main server	1 Mbps	1000 (100%)

Table 3. Impact of the cash server storage capacity on the entropy efficiency.

Storage capacity	Entropy efficiency, bps
10 TB	0.312561
11 TB	0.320883
12 TB	0.328918
13 TB	0.336713
14 TB	0.344305
20 TB	0.38695
30 TB	0.453011
40 TB	0.518798
50 TB	0.588478

Table 4. Impact of the channel transmission rate on the entropy efficiency.

10 Mbps	5 Mbps	2 Mbps	Entropy efficiency, bps
20%	60%	20%	0.312561
30%	50%	20%	0.322692
40%	40%	20%	0.332823
50%	30%	20%	0.342954
60%	20%	20%	0.353085

Hence the impact of network parameters on its efficiency is seen, which is of interest for designers of such systems. We emphasize that these estimates are obtained analytically, before implementing new variants and experimenting with them.

5. CONCLUSION

We have described a class of distributed memory systems, which comprises content delivery networks and many other structures, and proposed a method for theoretical (analytical) estimation of characteristics of such systems. This method is based on an approach well known in information theory: we show that the set of problems that a system can solve grows exponentially as a function of time, and hence the argument of this exponent is a reasonable characteristic of the capacity or efficiency. In our case, a problem is understood to be a sequence of files delivered by a system on users' requests (for comparison: in Shannon's classical work [4] the problem is to transmit a sequence of symbols; in [3] the problem is a sequence of commands executed by a computer).

The notions of processor capacity and entropy efficiency obtained in the present paper can be used in practice, which is illustrated by Section 4.

APPENDIX: PROOF OF THE THEOREM

We need a result of M. Fekete (cited from [11, p. 23]), which in the continuous case is formulated as follows.

Claim. *Let φ be a nonnegative function such that $\varphi(t_1) + \varphi(t_2) \geq \varphi(t_1 + t_2)$ for positive t_1 and t_2. Then $\lim\limits_{t\to\infty} \varphi(t)/t$ exists, and $\lim\limits_{t\to\infty} \varphi(t)/t = \liminf\limits_{t\to\infty} \varphi(t)/t$.*

Proof of equation (10). Let t_1 and t_2 be arbitrary positive numbers, and let $t_3 = t_1 + t_2$. We show that

$$|\mathcal{N}_i(t_1)||\mathcal{N}_i(t_2)| \geq |\mathcal{N}_i(t_3)|. \tag{15}$$

To this end, note that it follows from the stationarity and the definition of $\mathcal{N}_i(t)$ that, first, for any sequence $f_1^3 \dots f_{n_3}^3 \in \mathcal{N}_i(t_3)$ there exists a pair $f_1^1 \dots f_{n_1}^1 \in \mathcal{N}_i(t_1)$, $f_1^2 \dots f_{n_2}^2 \in \mathcal{N}_i(t_2)$ such that $f_1^3 \dots f_{n_3}^3$ is a prefix of the concatenation $f_1^1 \dots f_{n_1}^1 f_1^2 \dots f_{n_2}^2$, and second, for any pair $f_1^1 \dots f_{n_1}^1 \in \mathcal{N}_i(t_1)$, $f_1^2 \dots f_{n_2}^2 \in \mathcal{N}_i(t_2)$ there is at most one element of $\mathcal{N}_i(t_3)$ which is a prefix of the concatenation $f_1^1 \dots f_{n_1}^1 f_1^2 \dots f_{n_2}^2$. Hence, the number of different pairs of this type is not less than the number of elements in $\mathcal{N}_i(t_3)$, which implies (15).

Applying Fekete's result to $\varphi(t) = \log|\mathcal{N}_t|$, we obtain from (15) the proof of equation (10). △

Proof of equations (11) and (12). The proof is contained in Shannon's seminal paper [4].

Proof of claim (ii) of the theorem. As follows from (8), for given probabilities $\pi(f)$, $f \in F$, the entropy is maximal for a process π generating i.i.d. symbols. From this and definitions (7) and (9), we conclude that the maximum value of the entropy efficiency is given by

$$\mathcal{H}(\pi) = \frac{-\sum\limits_{f\in F} \pi(f)\log\pi(f)}{\sum\limits_{f\in F} \pi(f)\tau_i(f)} \tag{16}$$

for some probability distribution $\pi(f)$ on F. To complete the proof, we apply the well-known Kullback–Leibler inequality

$$\sum_{f\in F} p(f)\log\frac{p(f)}{q(f)} \geq 0,$$

where $p(\cdot)$ and $q(\cdot)$ are arbitrary probability distributions on a finite set F (see [7]). Hence it is seen that

$$\frac{-\sum\limits_{f\in F} p(f)\log p(f)}{\sum\limits_{f\in F} p(f)\log(1/q(f))} \leq 1.$$

In particular, this is true for the distribution $q(f) = Z_i^{-\tau_i(f)}$ for any distribution $p(f)$. The last inequality yields

$$\frac{-\sum\limits_{f\in F} p(f)\log p(f)}{\sum\limits_{f\in F} p(f)\tau_i(f)} \leq \log Z_i,$$

which coincides with (13). It is easily seen that for $p(f) = Z_i^{-\tau_i(f)}$ this inequality turns into equality, which proves equation (14). △

The author is grateful to P.A. Pristavka for his assistance in preparing the example in Section 4.

REFERENCES

1. Jeon, S.-W., Hong, S.-N., Ji, M., Caire, G., and Molisch, A.F., Wireless Multihop Device-to-Device Caching Networks, *IEEE Trans. Inform. Theory*, 2017, vol. 63, no. 3, pp. 1662–1676.

2. Hachem, J., Karamchandani, N., and Diggavi, S.N., Coded Caching for Multi-Level Popularity and Access, *IEEE Trans. Inform. Theory*, 2017, vol. 63, no. 5, pp. 3108–3141.

3. Ryabko, B., An Information-Theoretic Approach to Estimate the Capacity of Processing Units, *Perform. Eval.*, 2012, vol. 69, no. 6, pp. 267–273.

4. Shannon, C.E., A Mathematical Theory of Communication, *Bell Syst. Tech. J.*, 1948, vol. 27, no. 3, pp. 379–423; no. 4, pp. 623–656.

5. Ryabko, B. and Rakitskiy, A., An Analytic Method for Estimating the Computation Capacity of Computing Devices, *J. Circuits Syst. Comput.*, 2017, vol. 26, no. 5, p. 1750086.

6. Ryabko, B. and Rakitskiy, A., Application of the Computer Capacity to the Analysis of Processors Evolution, `arXiv:1705.07730 [cs.PF]`, 2017.

7. Cover, T.M. and Thomas, J.A., *Elements of Information Theory*, New York: Wiley, 1991.

8. Dinaburg, E.I., On the Relations among Various Entropy Characteristics of Dynamical Systems, *Izv. Akad. Nauk SSSR, Ser. Mat.*, 1971, vol. 35, no. 2, pp. 324–366 [*Math. USSR Izv.* (Engl. Transl.), 1971, vol. 5, no. 2, pp. 337–378].

9. Goodman, T.N.T., Relating Topological Entropy and Measure Entropy, *Bull. London Math. Soc.*, 1971, vol. 3, pp. 176–180.

10. Breslau, L., Cao, P., Fan, L., Phillips, G., and Shenker, S., Web Caching and Zipf-like Distributions: Evidence and Implications, in *Proc. 18th Annual Conf. on Computer Communications (INFOCOM'99), New York, USA, Mar. 21–25, 1999*, vol. 1, pp. 126–134.

11. Krichevsky, R., *Universal Compression and Retrieval*, Dordrecht: Kluwer, 1994.