Properties of Oblivious End-to-End Communication Protocols in Reliability Networks with Hypercubic and Mesh-like Topology

Martin Nehéz
Department of Computer Science and Engineering,
FEE CTU Prague, Karlovo náměstí 13,
121 35 Praha 2, Czech Republic
e-mail: nehez@fel.cvut.cz

January 16, 2007

Abstract

We design protocols for single-packet oblivious end-to-end communication problem according to induced shortest path requirement for several types of reliability networks modeled by random graphs. The most part of this paper contains an analysis of the protocols. Namely, the protocol for random k-dimensional generalized hypercubes GH uses packets with header size $O(k)$ bits and the protocol for n-node random tori T uses packets with header size $O(\log n)$ bits. Validity of the both protocols is satisfying whenever a necessary condition expressed in the form of the existence of induced shortest paths holds. We also claim the necessary conditions of fault-tolerant end-to-end communication protocols over the random graphs GH and T, respectively. These conditions are described in the terms of threshold functions and phase transition phenomena in random graphs.

1 Introduction

In recent years, massive exploitation of information technologies based on communication networks and multiprocessor architectures has caused the area of distributed computing and distributed systems to become of great interest by many researchers. One of the primary objects of study is the problem of reliable functioning of distributed systems. Many unreliable or defective models of distributed systems were examined: there are various types of faults in combination with several different models of their distribution. In this paper we adopt the model of interconnection networks with crash-fault communication links satisfying probabilistic distribution. This model is more realistic in practice than a so called worst-case model in which a bounded number of adverse distributed faults occur. The model used in our paper is as follows: we consider a point-to-point communication network modeled by a simple connected graph H in which the links fail independently with the failure probability $f = 1 - p$. It can be described as a generalization of the standard Erdös-Rényi random graph model. The fundamental problem, which arises in the reliability network model, is to guarantee such the operating conditions of the network which allow the faulty-free or near faulty-free communication among the nodes. Faulty-free communication is usually understand as the communication under the such conditions that the failure probability f is within the interval in which the random graph is connected (or "multiconnected") with high probability. Near faulty free communication usually means that the communication among the nodes is executed inside a connected subnetwork of the base network. Such a subnetwork can be mostly considered as a linear-sized connected component of the base graph. Networks with (near) faulty-free properties are also called robust networks and their graph-theoretic parameters are interesting for designing reliable routing strategies or other types of robust communication protocols.

Many various routing/communication algorithms and protocols were designed but their fault-tolerant (robust) characteristics are very different. We will consider a model which attains strong fault-tolerance, called oblivious end-to-end communication model in this paper. The end-to-end communication is the problem of sending a sequence of packets from a designated node called sender to a designated node called receiver through an unreliable communication network. This problem represents a fundamental task in distributed computing. Oblivious end-to-end communication means that transmitted packets contain headers with routing information and each routing decision is based solely on the content of the header. It means that nodes of the network do not keep routing information. The main advantage of such the model is its high fault-tolerance. We will consider an oblivious single-packet end-to-end communication problem in the network with randomly defective communication links in our paper. To optimizing the traffic in the network we examine a model of communication which uses induced shortest paths exclusively. We design a protocol for the mentioned communication model and investigate its properties.

1.1 Random Graph Model and Reliability Networks

We consider a point-to-point communication network modeled by a simple connected graph $H = (V, E)$, where V is the set of nodes (or processors) and E is a set of edges (or bidirectional communication links). The links of the network are either operational or inoperational (failure). Assume that a *random graph* G is obtained from a graph H with node set $\{1, \ldots, |V(H)|\}$ by independently removing each edge of H with a *failure probability* $1 - p$. For each spanning subgraph G of H it holds that $V(G) = V(H)$, $E(G) \subseteq E(H)$ and we will denote the set of all spanning subgraphs of H as 2^H. Let $\mathbb{G}(H, p) = (\Omega, \Pr)$ be a probability space of random graphs, such that $G \in \mathbb{G}(H, p)$ is a uniform labeled random spanning subgraph of H, where $\Omega = 2^H$ and

$$\Pr[G] \; = \; p^{|E(G)|}(1-p)^{|E(H)|-|E(G)|} \; .$$

This model, called the *reliability network model*, appears in [7] and [28] and is described in more details in [16]. Reliability network is a natural generalization of the binomial random graph model of Erdös-Rényi, denoted hereafter $\mathbb{G}(n, p)$, for which $H = K_n$ the complete graph on n nodes.

1.2 The Communication Model

We will examine the model of *oblivious single-packet induced shortest paths end-to-end communication*. We assume the point-to-point message-passing model of distributed system such that the communication network is represented by a reliability network model (a random graph) defined as before. Let us consider a *static* fault model in which links cannot change their status during the computation. The *end-to-end communication problem* (also called *reliable communication*) is the problem of sending a packet (or sequence of packets) from one specified processor S (*sender*) to another specified processor R (*receiver*) over an unreliable communication network. The network is assumed to be *asynchronous*. It means that the time for a packet to pass a link is finite but unbounded. We consider the *single-packet* end-to-end communication problem requiring the transmission of at least one packet from S to R. On the other hand, *stream-of-packet* end-to-end communication is the problem requiring the transmission of a sequence of packets from S to R. This problem is not considered in our paper.

We will investigate the properties of protocols for single-packet end-to-end communication problem. We consider *oblivious* (*memoryless*) protocols which take their routing decision at every node of the network, but the routing decision is based wholly on the content of the header of the packet. It means that the complete routing information is kept in the header of the packet. Each node of the networks "knows" only its self unique identification number and uniquely distinguished number of its ports (outgoing edges). The advantage of the oblivious protocol is their high tolerance to the failures occurred in the network, cf. [1]. In contrast, e.g. interval routing, in which the routing information in the form of intervals is kept by each node, has very poor robustness, cf. [30, 34].

We will measure the performance of the oblivious single-packet end-to-end protocol by the maximum size of the headers involved during the protocol execution. *Packet-header size* represents the amount of the routing information for the such distributed protocols. There are two requirements for each oblivious single-packet end-to-end communication protocol:

1. (*reception*) its execution allows to arrive at least one copy of the packet sent from its sender S to the receiver R,
2. (*termination*) after a finite time, no copy of the packets remains in the network.

Generally, end-to-end protocols can use arbitrary paths for transmitted packets. We will focus on protocols with induced shortest paths. Let H ge a graph of the network. For two nodes $u, v, \in V(H)$, let the notion $u - v$ *path* means the path of H with extremities u and v. We will denote a $u-v$ path as $P_{u,v}$. Let $dist_H(u, v)$ denotes the distance of nodes u and v in the graph H. Let $G \in \mathbb{G}(H, p)$ be a random graph and let u and v be as before. It holds $dist_G(u, v) \geq dist_H(u, v)$, since G is a subgraph of H. Each $u - v$ path of G

with the length $dist_H(u, v)$ will be called *shortest $u - v$ path induced by the graph H*, or shortly (H-)*induced shortest path*. The properties of induced shortest paths are investigated in more details in Section 4. The end-to-end communication protocol (over a reliability network) in which transmission of each packet from a sender S to the receiver R is executed along induced shortest paths is said to be *induced shortest path end-to-end communication protocol* (shortly *ISP end-to-end communication protocol*).

1.3 Related Work

Hop-count protocol [32] is a basic protocol for an oblivious end-to-end communication problem for unreliable networks with static link failures. It uses headers of size $O(\log n)$ in n-node networks. M. Adler and F. Fich have shown that the complete graph of n nodes requires headers of size $\Omega(\log n)$, cf. [1]. They also claimed that the oblivious end-to-end communication problem is closed under taking minors. P. Fraigniaud, C. Gavoille in [11] studied the lower bounds on the header size for 2-dimensional meshes and planar graphs. They have also shown that the lower bound on the header size depends on the tree-width of the network. Analysis of several versions of hop-count protocol is provided in [19]. For other results and several open questions see e.g. [10, 11].

Concept of a reliable communication over unreliable networks was studied in many works. We mention only two of them. Detailed analysis of bounded protocols for on-line sequence transmission including the model of probabilistic error assumption is presented by Y. Mansour and B. Schieber in [24]. The formal model of communication protocols over unreliable channels is described in [2]. There are also shown several properties of such a model in the latter paper. For an introduction to fault-tolerance and unreliable distributed systems we refer the corresponding sections in the monographs [6, 34], respectively.

Considering a bounded degree graph H, it is easy to show that a random graph of $\mathbb{G}(H, p)$ has not to be necessary connected for a constant probability p. In spite of this, several works about the random regular graphs are concerning with their near faulty free properties. In particular, A. Goerdt in [12, 13] has stated that $p = 1/(d - 1)$ is a threshold probability for the existence of linear-sized component in almost all random d-regular graphs. The extension of this result to the linear-sized subgraph which is an expander from the same author can be found in [14]. The threshold for a linear-sized component of the form $p = 1/k$ for a k-dimensional random hypercube is due to M. Ajtai, J. Komlos and E. Szemeredi, cf. [3] and the threshold for the same problem for 2-dimensional $n \times n$ random grid is $p = 1/2$, cf. [17]. Finally, the threshold values for the multiconnectivity properties for random regular graphs were claimed. The results are due to S. Nikoletseas, K. Palem, P. Spirakis, M. Yung, cf. [27, 28]. The notion of the threshold function plays a crucial role in the studying of the phase transition phenomena of random graphs when the graph process is between its subcritical and supercritical phase, cf. [4, 16, 22]. (Recently, the phenomenon of phase transition was observed also in the small world graphs, [25].) This concept is very similar to the physical view on the phase transition. Thus, the problem of percolation in physics is closely related. The end-to-end communication model over random graphs seems to be nearly identical to bond (edge) percolation. The result of H. Kesten [17] is one of the earliest that can prove this similarity. For the other works about the percolation see [18, 26, 33].

1.4 Our Results and Outline of Paper

We focus on the probability spaces of random k-dimensional generalized hypercubes $\mathbb{G}(GH_s^k, p)$ and random k-dimensional tori $\mathbb{G}(T_s^k, p)$ in our paper.

We will show that there is a protocol for oblivious single-packet induced shortest paths end-to-end communication for random k-dimensional generalized hypercubes $GH \in \mathbb{G}(GH_s^k, p)$ which uses packets with header size $O(k)$ bits and the similar protocol for n-node random tori $T \in \mathbb{G}(T_s^k, p)$ which uses packets with header size $O(k \log s) = O(\log n)$ bits.

We deal also with the validity of protocols with respect to the proposed communication model. We will show, in particular, that each end-to-end GH_s^k-induced shortest path communication protocol over a random generalized hypercube $GH \in \mathbb{G}(GH_s^k, p)$, with high probability, does not satisfy the reception requirement if $p < \widehat{p}_1 = 2/(k+1)$. The analogous value for random tori $T \in \mathbb{G}(T_s^k, p)$ is $\widehat{p}_2 = 1/k$.

The organization of the rest of this paper is as follows. Section 2 contains a description of necessary notions and definitions. The protocols for the mentioned communication problem are described in Section 3. Section 4 contains analysis of the necessary condition for a reliable communication with respect to our communication model. Concluding remarks are given and other related/open problems are discussed in Section 5.

2 Definitions and Background

Asymptotic notation (such as O, o, Ω, Θ) is used in the usual way in this paper.

2.1 Monotonicity and Threshold Functions

A family of subsets $Q \subseteq 2^H$ is said to be *increasing* if $G_1 \subseteq G_2$ and $G_1 \in Q$ imply that $G_2 \in Q$. A family of subsets Q' is *decreasing* if $2^H \setminus Q'$ is increasing. A family which is either increasing or decreasing is called *monotone*. A family of graphs from 2^H can be identified with a graph property.

Threshold functions are defined for monotone properties. In our paper it is sufficient to introduce them only for increasing properties. For other details see [16]. Let H be an n-node graph and let Q be an increasing property of graphs from 2^H. A function $\widehat{p} = \widehat{p}(n)$ is called *threshold function* for Q iff both of the following conditions hold:

$$\lim_{n\to\infty} \frac{p(n)}{\widehat{p}(n)} = 0 \;\Rightarrow\; \lim_{n\to\infty} \Pr[\; \mathbb{G}(H, p(n)) \textit{ has } Q \;] = 0,$$

$$\lim_{n\to\infty} \frac{p(n)}{\widehat{p}(n)} = 1 \;\Rightarrow\; \lim_{n\to\infty} \Pr[\; \mathbb{G}(H, p(n)) \textit{ has } Q \;] = 1.$$

The notion of the threshold function plays a crucial role in the studying of the phase transition phenomena of random graphs, cf. [4, 16, 22].

2.2 Cayley Graphs

Let $(\Gamma, \circ)$ be a finite group (with binary operation $\circ$) and let $A \in \Gamma$ be a set of its generators. Then $Cay(\Gamma, A) = (V, E)$ is said to be *the Cayley digraph of the group* Γ *generated by* A iff:

- $V = \Gamma$,
- $E = \{\; (u, v) \mid \exists a \in A, u \circ a = v \;\}$.

If A is without an identity element and closed under inverses, then $Cay(\Gamma, A)$ is a simple graph and it is called *Cayley graph*. Each Cayley graph is vertex-symmetric and this implies that it is also regular.

Let $(\mathbb{Z}_s, \oplus)$ denote the group of integers modulo $s \geq 1$ with the binary operation $\oplus$ and with the *zero (identity) element* 0. For a positive integer k, let $\mathbb{Z}_s^k$ denote a Cartesian product of k groups $(\mathbb{Z}_s, \oplus)$. Let ε_i denote i-th *unit* (k-dimensional) *vector* and let $-\varepsilon_i$ stands for its inverse. An example is, e.g. $\varepsilon_1 = (1, 0, \ldots, 0)$ and $-\varepsilon_1 = (-1, 0, \ldots, 0)$. We will use following Cayley graphs.

Generalized hypercube. For positive integers s and k, a *k-dimensional generalized hypercube* (also called *Hamming graph*) GH_s^k is defined as $Cay(\mathbb{Z}_s^k, A)$, where $A = \{\ c \cdot \varepsilon_i \mid 1 \leq i \leq k,\ c \in (\mathbb{Z}_s \setminus \{0\})\ \}$.
In particular, GH_2^k is called *k-dimensional hypercube* and it is denoted by H_k.

Tori. For positive integers s and k, a *k-dimensional torus* T_s^k is defined as $Cay(\mathbb{Z}_s^k, \{\ \pm\varepsilon_i \mid 1 \leq i \leq k\ \})$.

For other types of Cayley graphs and their properties see e.g. [5, 15, 21, 23]. Cayley graphs such as hypercubes or tori are often used as interconnection networks for parallel and distributed algorithms, [20, 34].

3 Oblivious ISP End-to-End Communication Protocols for Reliability Networks

Let S be a sender and R be a receiver in an n-node network. Assume an oblivious single-packet end-to-end communication problem. Standard hop-count protocol [32] which solves this problem is based on the idea of flooding algorithm [6] designed for e.g. the problem of broadcast in distributed systems. The main principle behind the hop-count protocol is that a node receiving a packet whose header involves the hop count $j < n - 1$ updates the hop count value by setting $j + 1$ instead of j and forwards a copy of the packet with the new header to each its neighbors. If the hop count of the packet is $j = n - 1$ then node removes such packet. The diameter of a network is at most $n - 1$ and thus, if S and R are connected then at least one copy of a packet sent from S eventually reaches R and the remaining packets are removed from the network.

Although the hop-count protocol has high tolerance to the failures possibly occurred in the networks its main disadvantage is generation of too many packets during the computation. To eliminate this disadvantage we design a protocol that uses only induced shortest paths (ISP). Our protocol solves the problem of oblivious ISP end-to-end communication. It is designed for random generalized hypercubes and random tori and it has different properties for each of these two types of network. We divide description of these two cases into separate subsections.

3.1 Protocol for Random Generalized Hypercubes

Recall the definition of a generalized hypercube GH_s^k. It holds that $GH_s^k = Cay(\mathbb{Z}_s^k, A)$ where $A = \{\ c \cdot \varepsilon_i \mid 1 \leq i \leq k,\ c \in (\mathbb{Z}_s \setminus \{0\})\ \}$ is the set of generators. Let topology of the network be given by a random generalized hypercube $GH \in \mathbb{G}(GH_s^k, p)$. Assume that each node (processor) of the network has an identification number $u \in \mathbb{Z}_s^k$ and it also knows the parameters s and k. Moreover, each its outgoing edge (*port*) is labeled according to the generator of the corresponding edge. (Such labeling of edges is identical to a Cayley sense of direction or Group-based structural information, cf. [8].) For a generator $a \in A$ we will denote the corresponding outgoing port as $e(a)$. Each packet is denoted by $\langle h, M \rangle$, where h is the header and M is the contents of the packet. The header of each packet is a k-dimensional vector of bits $h = (h_1, \ldots, h_k)$, where $h_i \in \{0, 1\}$ for $i = 1, \ldots, k$. Initially, h is a zero-vector and each *send*-statement causes that the number of ones in h is increased by one. Our protocol, called GH, starts from S that sends packets $\langle h, M \rangle$ to all its neighbors, where $h = \varepsilon_i$ iff the packet is transmitted along a port $e(c \cdot \varepsilon_i)$. It means that the header containing the one on its jth position is sent along the all ports that are associated to the generators with the nonzero value on jth position. Of course, packets are transmitted only via operational links (ports). The transmission of packets continues by the similar manner through "intermediate" processors. When an intermediate processor u receives a packet which header h contains r ones ($r < k$), it sends its copy (copies) with a new header h' which contains $r + 1$ ones. If it holds that $h' = h \oplus \varepsilon_i$ then the packets with headers h' are sent along the all operational ports $e(c \cdot \varepsilon_i)$ for $c = 1, \ldots, s - 1$. (Symbol "$\oplus$" stands for the addition of vectors modulo s.) The communication is terminated when a processor receives packet(s) in which the header is equal to the k-dimensional vector of ones, i.e. $h = (1, \ldots, 1) = \mathbb{1}$. The protocol for a receiver R is very simple: R is waiting for a packet(s). When it is delivered, the receiver switches to the state *success*. The formal description of the protocol is as follows.

Protocol GH: sender S

```
for c = 1 to s − 1 do
  for i = 1 to k do
    if port e(c · ε_i) is operational then
       h := ε_i ;
       send ⟨h, M⟩ via port e(c · ε_i)
```

The formal description of the protocol for the "intermediate" processor u follows. It holds: $u \neq S$ and $u \neq R$. Recall that $h = (h_1, \ldots, h_k)$, symbol $\oplus$ represents the extension of the operator addition modulo s to vectors and $\mathbb{1} = (1, \ldots, 1)$ of k dimensions.

Protocol GH: processor u

```
upon receiving ⟨h, M⟩
if h ≠ 𝟙 then
  for c = 1 to s − 1 do
    for i = 1 to k do
      if h_i = 0 and port e(c · ε_i) is operational
      then
         h' := h ⊕ ε_i ;
         send ⟨h', M⟩ via port e(c · ε_i)
```

Protocol GH: receiver R

```
upon receiving ⟨h, M⟩
result success
```

Before proving the correctness of the protocol GH we show an example of its execution.

Example 1 *Let* $k = 5$ *and* $s = 8$*, let* $S = (3, 0, 2, 6, 1)$ *and* $R = (4, 6, 2, 1, 3)$*. We describe an example of one possible communication from*

S to R. One transmission step (a send-receive sequence) is denoted by $\stackrel{h}{\longrightarrow}$, *where h is a header of a transmitted packet. A transmission step is an action between two processors denoted by* 5*-tuple identification numbers.*
$(3,0,2,6,1) \stackrel{(0,1,0,0,0)}{\longrightarrow} (3,6,2,6,1) \stackrel{(1,1,0,0,0)}{\longrightarrow} (4,6,2,6,1) \stackrel{(1,1,0,1,0)}{\longrightarrow} (4,6,2,1,1) \stackrel{(1,1,0,1,1)}{\longrightarrow} (4,6,2,1,3)$
Note that the communication can be viewed as a computation over the group $\mathbb{Z}_s^k$ *from the initial vector S to the resulting vector R with the sequence of operations chosen nondeterministically. Each operation corresponds to the one application of a some generator to a given vector.*

In order to prove the correctness of the protocol it is necessary to show that it satisfies the requirements of (1) reception and (2) termination. Moreover, it have to insure correct transmission of packets along induced shortest path(s) in random generalized hypercube. To prove this we will need the fact that the distance between two nodes u and v of GH_s^k is the number of coordinates in which differ vectors u and v.

Lemma 1 *In every admissible execution of the protocol GH in generalized hypercube* GH_s^k*, at least one copy of a packet sent from the sender S is delivered to all other nodes of* GH_s^k*. Each packet delivered to R is transmitted along a shortest path.*

Proof. The proof proceeds by induction on k. The first part of the assertion follows from the definition of the generators: a set $A \subseteq B$ is a set of generators if any element of B can be obtained as a product of finitely many elements of A.

For $k = 1$ the set of edges of GH_s^1 corresponds exactly to the set of generators of the group $\mathbb{Z}_s^1$. Thus the sender S sends packets to all other nodes of the network. The transmission of packets passes exactly along the shortest paths, since the execution of the protocol $GH : u$ terminates without sending any packet.

Let $k > 1$. We will distinguish two cases.

1. A processor u receives a packet with header in which $h_k = 0$. It means that no copy of this packet has already been transmitted trough processors $v = (v_1, \ldots, v_k)$ which identification numbers differ in kth position. The generators for all such vertices are elements $c \cdot \varepsilon_k$ for $c = 1, \ldots, s-1$. Consequently, according to the statement of $GH : u$ "`for` $i = 1$ `to` k `do`",

by the assignment $i = k$, copies of the received packet will be sent towards all nodes with different kth position.

2. A processor u receives a packet with header in which $h_k = 1$. It means that copies of this packet have already been transmitted trough all the processors which identification numbers differ in kth position. Consequently, the condition "`if` $h_i = 0 \ldots$ `then`..." causes that no copy of the packet will be sent twice to the mentioned nodes. Hence, at lest one copy of a packet sent from S is delivered to all other nodes in the network.

The assertion that packets use only the shortest paths follows from the following property. The condition "`if` $h_i = 0 \ldots$ `then`..." and the assignment "$h' := h \oplus \varepsilon_i$" for a fixed i cause that no copy of the same packet is transmitted twice along the edge generated by the generator in ith dimension. Therefore, if S and D differ in l positions then the each packet addressed to R from S must passes through a path with the length l. ♢

Lemma 2 *Every admissible execution of the protocol GH in random generalized hypercube $GH \in \mathbb{G}(GH_s^k, p)$ is terminated and it uses packets with header size at most $O(k)$ bits.*

Proof. The termination of the protocol follows directly from the condition "`if` $h \neq \mathbb{1}$ `then`" of $GH : u$. When the equation $h = \mathbb{1}$ holds then no other *send*-statement is executed. Note that according to the assignment statement "$h' := h \oplus \varepsilon_i$" each receive-send action (i.e. once execution of the protocol $GH : u$) increases the number of ones in h by one. Hence, the protocol GH is terminating.

Each packet is a k-dimensional vector of bits. For the description of this format we can use at most $c \cdot k$ bits for any appropriate constant c. Hence, the total number of bits to encode each packet is at most $O(k)$. ♢

Lemma 3 *If there is a GH_s^k-induced shortest path between nodes S and R in $GH \in \mathbb{G}(GH_s^k, p)$ then every admissible execution of the protocol GH insures that at least one copy of a packet sent from S is delivered to R.*

Proof. The proof follows from the definition of the induced shortest path, Lemma 1 and properties of the protocol. Let $dist_{GH_s^k}(S, R) = l$. If there is at least one GH_s^k-induced shortest path between nodes S and R then its length is l. Let us denote one such path as $P_{S,R}$. It implies that all links along the path $P_{S,R}$ are operational. Consequently, there is a sequence σ of generators that correspond to the links along the path $P_{S,R}$. Each execution of the protocol GH "finds" all possibilities how to reach from S to R along induced shortest paths. (See Example 1.) It means that each execution of the protocol GH must "find" also the sequence of generators σ which corresponds to the path $P_{S,R}$. (For the details see the proof of Lemma 1.) The argument is that all links occurred in $P_{S,R}$ are operational, thus the condition "`if` port $e(c \cdot \varepsilon_i)$ is operational" falls *true* for all $c \cdot \varepsilon_i \in \sigma$. Hence, at least one packet from S is arrived in R. $\diamondsuit$

Note that the existence of a GH_s^k-induced shortest path between nodes S and R is the necessary condition for the correct reception of the packet(s) by R.

According to Lemma 1, 2 and 3 we have obtained the following main result.

Theorem 1 *There is a protocol such that if there is an GH_s^k-induced shortest path between nodes S and R then it solves the problem of oblivious single-packet induced shortest paths end-to-end communication problem over a random generalized hypercube $GH \in \mathbb{G}(GH_s^k, p)$ with packet-header size $O(k)$ bits.*

3.2 Protocol for Random Tori

The protocol for random tori, called T, is based on the same idea as the previous one. According to the definition of torus $T_s^k = Cay(\mathbb{Z}_s^k, \{ \pm\varepsilon_i \mid 1 \leq i \leq k \})$, we assume that each node (processor) of the network has its own identification number $u \in \mathbb{Z}_s^k$ and it also knows the parameters s and k. Similarly as in the previous case, the labels of the outgoing ports correspond to the generators of T_s^k. The main difference is in the format of the packet headers. Each header t of the protocol T is of the form of k-dimensional vector of integers $t = (t_1, \ldots, t_k)$, where $t_i \in \{-\lfloor s/2 \rfloor, \ldots, 0, \ldots, \lfloor s/2 \rfloor\}$ for $i = 1, \ldots, k$. Initially, t is a zero-vector and each *send*-statement causes that t will be changed by one exactly in the one position. More precisely, sender S sends a packet with the header $t = a$ via the operational port $e(a)$, for each generator $a = \pm\varepsilon_i$, where $i \in \{1, \ldots, k\}$. When an intermediate processor u receives a packet with the header $t = (t_1, \ldots, t_k)$ it first checks whether the absolute value of each t_i is lower than $\lfloor s/2 \rfloor$. This condition is necessary for the reason that the diameter (maximum of the distances between two nodes over all pairs of nodes) of a torus T_s^k is $k \cdot \lfloor s/2 \rfloor$ and for a cycle in one dimension it is $\lfloor s/2 \rfloor$. Consequently, for each $i \in \{1, \ldots, k\}$ for which the condition $|t_i| < \lfloor s/2 \rfloor$ falls *true* ($|.|$ denotes absolute value) the next execution of the protocol continues separately in two cases.

1. If $t_i > 0$ then the packet with the header $t' = t + \varepsilon_i$ is sent via operational port $e(\varepsilon_i)$; the symbol "+" denotes the standard addition of vectors,

2. if $t_i < 0$ then the packet with the header $t' = t - \varepsilon_i$ is sent via operational port $e(-\varepsilon_i)$; binary operator "$-$" stands for the standard subtraction of vectors.

For $t_i = 0$ both of the cases are executed. It follows that each *send*-statement causes increasing/decreasing exactly one coordinate of the header t by one. It corresponds to the passing of a packet through exactly one operational link.

The receiver R executes the following protocol: upon receiving a packet it switches to the state *success*. The formal description of the protocol is listed below.

Protocol T: sender S

```
for i = 1 to k do
  if port e(ε_i) is operational then
     send ⟨ε_i, M⟩ via port e(ε_i)
  if port e(−ε_i) is operational then
     send ⟨−ε_i, M⟩ via port e(−ε_i)
```

For the intermediate node u it holds: $u \neq S$ and $u \neq R$. Binary operators "+", "$-$", respectively, stand for the standard addition and subtraction of the vectors.

Protocol T: processor u

```
upon receiving ⟨t, M⟩
for i = 1 to k do
  if |t_i| < ⌊s/2⌋ then
     if t_i ≥ 0 and port e(ε_i) is operational
     then
        t' := t + ε_i ;
        send ⟨t', M⟩ via port e(ε_i)
```

`if` $t_i \leq 0$ and port $e(-\varepsilon_i)$ is operational
`then`
  $t' := t - \varepsilon_i$ `;`
  `send` $\langle t', M\rangle$ `via port` $e(\varepsilon_i)$

Protocol T: receiver R

`upon receiving` $\langle t, M\rangle$
`result` *success*

To better understanding of the protocol we list the following example.

Example 2 *Let $k = 4$ and $s = 5$, let $S = (3, 0, 2, 4)$ and $R = (4, 3, 2, 1)$. We describe an example of one possible communication from S to R. The symbols such as $\xrightarrow{h}$ are used by the same manner as in the previous example.*
$(3, 0, 2, 4) \xrightarrow{(0,0,0,1)} (3, 0, 2, 0) \xrightarrow{(0,0,0,2)} (3, 0, 2, 1) \xrightarrow{(0,-1,0,2)} (3, 4, 2, 1) \xrightarrow{(0,-2,0,2)} (3, 3, 2, 1) \xrightarrow{(1,-2,0,2)} (4, 3, 2, 1)$
Note that the absolute value of each header coordinate must be lower than or equal to $\lfloor 5/2 \rfloor = 2$.

Before proving the correctness of the protocol we note that the distance of two nodes $u = (u_1, \ldots, u_k)$ and $v = (v_1, \ldots, v_k)$ of T_s^k is given by:

$$dist_{T_s^k}(u, v) = \sum_{i=1}^{k} l_i,$$

where for each $i \in \{1, \ldots, k\}$ it holds:

$$l_i = \begin{cases} |u_i - v_i| & \text{if } |u_i - v_i| \leq s/2, \\ s - |u_i - v_i| & \text{if } |u_i - v_i| > s/2. \end{cases}$$

As before, the correctness proof of the protocol involves the parts about its reception and termination validity as well as about the correct transmission of packets along induced shortest path(s) insurance.

Lemma 4 *In every admissible execution of the protocol T in torus T_s^k, at least one copy of a packet sent from the sender S is delivered to all other nodes of T_s^k. Each packet delivered to R is transmitted along a shortest path.*

Proof. At first we prove the part that at least one copy of a packet is delivered to all nodes except S. It is sufficient to restrict our attention only on one dimension, say jth. Let j be a fixed number such that $1 \leq j \leq k$. Recall the definition of the generators. We have two generators for jth dimension, ε_j and $-\varepsilon_j$. Note that ports $e(\varepsilon_j)$ and $e(-\varepsilon_j)$ are opposite. It means that if a packet was received from the port $e(\varepsilon_j)$ then the consecutive action which sends this packet via the same port $e(\varepsilon_j)$ is redundant. It yields that considering each dimension, packets are passed exclusively in one of the two possible directions. In spite of this, jth position in the packet header always indicates the direction in which the copies of the packet are passed. If $t_j > 0$ then the direction follows the direction of the generator ε_j. The opposite direction is active for $t_j < 0$. The header with $t_j = 0$ indicates that such packet is not still passed along jth dimension. This is the only situation when such packet is sent via the both ports $e(\varepsilon_j)$ and $e(-\varepsilon_j)$. (See the corresponding statements in the protocol.) Each *send*-statement via the port $e(\varepsilon)$ or $e(-\varepsilon)$, respectively, causes incresing/decreasing of the value t_j by one. This process continues until $|t_j| = \lfloor s/2 \rfloor$. Recall that this value represents the maximal distance between two nodes in one dimension. Note that for s even, the nodes that are at the distance $l_j = s/2$ from the sender S receive the packets from both opposite directions. (For odd s, it is only from the one direction.) Such the way the packets from S are delivered to all nodes that differ in jth coordinate. Moreover, no copy of the same packet is transmitted twice along the same edge. A generalization of this analysis to all coordinates completes the proof.

In order to prove the second part of the lemma it is sufficient to reflect the fact that for a given header t, the addition of all $|t_i|$ for $i = 1, \ldots, k$ expresses the distance between S and the node that received the packet with header t. The rest follows from the previous analysis and from the definition of the distance in T_s^k. Alternatively, this assertion can be also proved by induction on the distance between S and R. $\diamondsuit$

Lemma 5 *Every admissible execution of the protocol T in random torus $T \in \mathbb{G}(T_s^k, p)$ is terminated and it uses packets with header size at most $O(k \log s)$ bits.*

Proof. The "termination condition" is: $|t_i| = \lfloor s/2 \rfloor$ must hold together for each $i = 1, \ldots, k$. Let us denote this condition by $\mathcal{C}$. It is easy to see that the protocol $T : u$ does not send any message whenever $\mathcal{C}$ is true. Initially, each header is a zero-vector and each *send*-statement increases/decreases exactly one coordinate of t by one. The increment-operation is executed exclusively if $t_j \geq 0$ (for each j) and the decrement one exclusively if $t_j \leq 0$. It yields that after a finite number of *send*-statements each header will satisfy the condition $\mathcal{C}$. Consequently, the protocol T satisfies the termination requirement.

Each header is a k-dimensional vector of integers. It is necessary to encode at most $s + 1$ integers in each coordinate. This can be done by at most $\lceil \log_2(s+1) \rceil$ bits. The expression can be estimated by the upper bound $\log_2 s + c$ for an appropriate constant c. It yields that the requirements for the size of the packet header are at most $O(k \log s)$ bits. $\diamondsuit$

Lemma 6 *If there is a T_s^k-induced shortest path between nodes S and R in $T \in \mathbb{G}(T_s^k, p)$ then every admissible execution of the protocol T insures that at least one copy of a packet sent from S is delivered to R.*

Proof. The proof is the same as the proof of Lemma 3. $\diamondsuit$

As before, the existence of a T_s^k-induced shortest path between nodes S and R is the necessary condition for the correct reception of the packet(s) by R.

The following main result is the direct consequence of Lemma 4, 5 and 6.

Theorem 2 *There is a protocol such that if there is an T_s^k-induced shortest path between nodes S and R then it solves the problem of oblivious single-packet induced shortest paths end-to-end communication problem over a random torus $T \in \mathbb{G}(T_s^k, p)$ with packet-header size $O(k \log s)$ bits.*

4 Necessary Conditions for ISP End-to-End Communication Protocols

We deal with the necessary condition of the existence of induced shortest paths (ISP) between nodes S and R in more details in this section. We employ the observation that ISP end-to-end communication problem is very resembling the directed percolation, cf. [18, 26], especially to the directed bond percolation. This similarity suggests to investigate the validity of ISP properties in the terms of threshold functions and phase transition phenomena in random graphs. As we note before, the notion of the threshold function is well-defined only for monotone properties. In spite of this we first show that the property of ISP existence is monotone, rather increasing.

4.1 Monotonicity of ISP Properties

Let $u, v \in V(H)$ be arbitrary but fixed nodes and let l be a positive integer. Let $R^l_{u,v}$ denote the property "graph $G \in 2^H$ contains at least one path $P_{u,v}$ of the length l such that $P_{u,v}$ is the shortest path of G".

Lemma 7 *The property $R^l_{u,v}$ is not increasing.*

Proof. For $u, v \in V(H)$, $u \neq v$, let $P_{u,v}$ denote a $u - v$ path of H such that it is not a shortest path in H. Put l the length of $P_{u,v}$ and let G_1 be $P_{u,v}$. Consequently, the property $R^l_{u,v}$ holds for G_1. Let $G_2 \in 2^H$ be a new graph such that $G_1 \subseteq G_2$ and in addition, G_2 contains a $u - v$ path $P^*_{u,v}$ which length is equal to $dist_H(u, v)$. It implies that $P^*_{u,v}$ is the shortest path in H and also in G_2. From the construction of $P_{u,v}$ follows that $l > dist_H(u, v)$ and thus, $P_{u,v}$ is not the shortest path in G_2. It yields that the property $R^l_{u,v}$ does not hold for the graph G_2. $\diamondsuit$

Observe that each spanning subgraph G of H involves at most the same arrangement of edges as H however, G may contain also less edges. Hence, the following simple property holds.

Lemma 8 *If $G \in 2^H$ contains a path $P_{u,v}$ of the length $l = dist_H(u, v)$ then $P_{u,v}$ is the shortest path in G.*

However the inverse implication does not hold and that is the reason for the fact that $R^l_{u,v}-$like properties are not increasing. In spite of this, we will restrict at the shortest paths with the length of $dist_H(u, v)$, exclusively.

Let $Q_{u,v}$ denote the following property: "graph $G \in 2^H$ contains at least one path $P_{u,v}$ of the length $dist_H(u, v)$". Note that from Lemma 8 follows that $P_{u,v}$ is the shortest path also in G.

Lemma 9 *The property $Q_{u,v}$ is increasing.*

Proof. Let $G_1 \in 2^H$ be a spanning subgraph of H satisfying the property $Q_{u,v}$. Let G_2 be a spanning subgraph of H such that $G_1 \subseteq G_2$. Because of Lemma 8, any addition of edges to G_1 does not affect the property $Q_{u,v}$. Hence $Q_{u,v}$ holds also for the graph G_2. $\diamondsuit$

Note that if for $G \in 2^H$ holds the property $Q_{u,v}$ then each $u - v$ path of G with the length $dist_H(u, v)$ is the shortest $u - v$ path induced by the graph H.

4.2 Random Generalized Hypercubes

We will use the following property.

Lemma 10 *Let $u, v \in V(GH_s^k)$ be distinct nodes with distance l. The number of all different $u - v$ paths with the length l in GH_s^k is $l!$.*

Proof. Recall the definition of the generalized hypercube. If the distance of $u = (u_1, \ldots, u_k)$ and $v = (v_1, \ldots, v_k)$ is l then vectors u and v differ in l positions. Thus there are exactly $l!$ possibilities how to chose the shortest path $P_{u,v}$. $\diamondsuit$

The following upper bound on factorial is derived from the Stirling's formula, cf. [9].

Lemma 11

$$n! \leq \left(\frac{n+1}{2}\right)^n$$

The main theorem of this section is the following one.

Theorem 3 *Let $k \geq 2$ and let sender S and receiver R be fixed nodes of GH_s^k with distance $l \geq k$ (in GH_s^k). Let:*

$$\widehat{p}_1 = \frac{2}{l+1} \ .$$

If $p < \widehat{p}_1$ then the probability that an arbitrary end-to-end GH_s^k-induced shortest path communication protocol over a random generalized hypercube $GH \in \mathbb{G}(GH_s^k, p)$ satisfies the reception requirement is approaching 0 as $l \to \infty$.

Proof. Let $dist_{GH_s^k}(S, R)$ be l. The argument is based on the Lemma 10. We assume that k is an increasing sequence $(k_i)_{i=1}^{\infty}$ and consequently the number of nodes of GH_s^k is increasing. We will consider GH_s^k as a finite but very large graph. Supposing l is related to k ($l \leq k$), l can be also considered as an increasing sequence $(l_i)_{i=1}^{\infty}$ Let $X_{S,R}$ be a random variable on $\mathbb{G}(GH_s^k, p)$ associated to the property $Q_{S,R}$. This random variable can be split into indicator (0-1) random variables ξ_P as follows:

- $\xi_P(GH) = 1$, if GH contains one GH_s^k-induced shortest (S, R)-path P,
- $\xi_P(GH) = 0$, otherwise.

If the length of each path P is l, then it holds:

$$\Pr[\xi_P = 1] = p^l$$

for all P as above. Thus,

$$X_{S,R} = \sum \xi_P \ ,$$

where the summation ranges over all shortest $S - R$ paths P induced by the graph GH_s^k. Let us express the expectation $E(X_{S,R})$. From Lemma 10 and the linearity of the expectation follows the following equation:

$$E(X_{S,R}) = E\left(\sum \xi_P\right) = l!\ p^l \ .$$

Using Lemma 11 we have:

$$E(X_{S,R}) \leq \left(\frac{l+1}{2}\right)^l \left(\frac{2}{l+1}\right)^l = 1$$

for $p = \widehat{p}_1 = 2/(l+1)$. The Markov's inequality, cf. [4, 9] yields:

$$\Pr[X_{S,R} > 1] \leq E(X_{S,R}) < 1$$

for each $p < \widehat{p}_1$. More precisely, let $p = \delta \cdot \widehat{p}_1$, where $0 < \delta < 1$. According to the Markov's inequality we obtain that the probability that there is a shortest path $P_{S,R}$ induced by the graph GH_s^k in $\mathbb{G}(GH_s^k, p)$ is at most δ^l. Note that $\delta^l \to 0$ as $l \to \infty$. Let us consider an arbitrary end-to-end ISP communication protocol $\mathcal{P}$. Consequently, the probability that a packet sent from S will be delivered to R according to $\mathcal{P}$ is approaching 0 as $l \to \infty$. Hence, the reception requirement does not hold for an arbitrary protocol $\mathcal{P}$. $\diamondsuit$

Remark 1 *In order to prove that $\widehat{p}_1$ is the threshold function it is necessary to use the second moment method, cf. [4]. It suggests to show that for the variance $Var(X_{S,R})$ it holds: $Var(X_{S,R}) = o(E^2(X_{S,R}))$ and consequently, $X_{S,R} \sim E(X_{S,R})$. To enumerate the variance $Var(X_{S,R})$ we will use the equation from [4, 9]: $Var(X_{S,R}) = E(X^2_{S,R}) - E^2(X_{S,R})$. The estimation*

$$E(X^2_{S,R}) \sim E(X_{S,R}) \int_1^l l^x p^x \left(\frac{x}{e}\right)^x dx$$

yields the result:

$$Var(X_{S,R}) \sim E(X_{S,R}) \cdot (l^{l/c} - c^l) \ .$$

However, this approximation is too weak for our purpose. Thus the question whether $\widehat{p}_1$ is the threshold function still remains open.

In particular, for $l = k$ we obtain the following consequence.

Corollary 1 *Let $k \geq 2$ and let S and R be fixed nodes of GH^k_s with distance k (in GH^k_s). Put:*

$$\widehat{p}_1 = \frac{2}{k+1} \ .$$

If $p < \widehat{p}_1$ then the probability that an arbitrary end-to-end GH^k_s-induced shortest path communication protocol over a random generalized hypercube $GH \in \mathbb{G}(GH^k_s, p)$ satisfies the reception requirement is approaching 0 as $k \to \infty$.

4.3 Random Tori

Assume two nodes u, v of T^k_s. Recall the definition of the distance between u and v in T^k_s. Let us consider u, v such that $l_i < s/2$ for each $i \in \{1, \ldots, k\}$. Let the length of any shortest path $P_{u,v}$ be l and let vectors u, v differ in k positions. There are k ways for choosing the next step from the node u to its neighbor along the path $P_{u,v}$. This gives the following property.

Lemma 12 *Assume the previous notation. Let $u, v \in V(T^k_s)$ be distinct nodes with distance l. The number of all different $u - v$ paths with the length l in T^k_s is:*

$$\binom{l}{l_1, \ldots, l_k}.$$

Note that despite the fact that l is large enough, it may fall the case that S and R differ in only one coordinate, say j. This situation yields that $l = l_j$ and there is only one shortest path $P_{S,R}$. Hence, for the random variable $X_{S,R}$ on the probability space $\mathbb{G}(T_s^k, p)$ it holds that $E(X_{S,R}) = p^l \to 0$ as $l \to \infty$. In order to eliminate such cases we will concentrate our interest only on the pairs of nodes with "many" shortest $u - v$ paths.

Definition 1 *Two nodes $u, v \in V(T_s^k)$ are* diagonal *if for all their coordinate differences $l_1, \ldots, l_k$ holds: $l_1 = l_2 = \ldots = l_k$.*

Similarly as in the previous subsection, we claim the necessary condition for the reception requirement of end-to-end ISP protocols.

Theorem 4 *Let $k \geq 2$ be a fixed integer and let $k \ll s$. Let S and R be fixed diagonal nodes of T_s^k with distance l (in T_s^k). Let:*

$$\widehat{p}_2 = k^{-1} \ .$$

If $p < \widehat{p}_2$ then the probability that an arbitrary end-to-end T_s^k-induced shortest path communication protocol over a random torus $T \in \mathbb{G}(T_s^k, p)$ satisfies the reception requirement is approaching 0 as $l \to \infty$.

Proof. The proof is similar to the proof of Theorem 3. We advert mostly the main differences. We assume s to be an increasing sequence $(s_i)_{i=1}^{\infty}$, whereas k be a constant. Let $Q_{S,R}^D$ denote the following property: "graph $T \in 2^{T_s^k}$ contains at least one path $P_{S,R}$ of the length $dist_{T_s^k}(S, R) = l$, under the condition that S and R are diagonal". Let $X_{S,R}^D$ be a random variable on $\mathbb{G}(T_s^k, p)$ associated to the property $Q_{S,R}^D$. Consequently, $X_{S,R}^D$ counts the number of paths $P_{S,R}$ in random torus $T \in \mathbb{G}(T_s^k, p)$. Let $E(X_{S,R}^D)$ denote the expectation of $X_{S,R}^D$. By the same manner as in the proof of the Theorem 3 and using Lemma 12 we obtain the following equation:

$$E(X_{S,R}^D) = \frac{l!}{l_1! \ldots l_k!} \, p^l = \frac{l! \cdot p^l}{[(l/k)!]^k} \ ,$$

since $l_1 = \ldots = l_k = l/k$. According to Lemma 11 it holds:

$$E(X_{S,R}^D) \sim \left(\frac{l+1}{l+k} \right)^l k^l p^l \leq (kp)^l \ .$$

The same estimation can be also claimed from the Stirling's formula, since $k \ll l$. Hence, the Markov's inequality yields the inequality:

$$\Pr[X_{S,R}^D > 1] \le E(X_{S,R}^D) < 1$$

for each $p < \widehat{p}_2 = 1/k$. Moreover, for $p = \delta \cdot \widehat{p}_2$ and $0 < \delta < 1$ we have:

$$\Pr[X_{S,R}^D > 1] \le \delta^l \ ,$$

and $\delta^l \to 0$ as $l \to \infty$. The rest is the same as in the proof of the Theorem 3. $\diamondsuit$

The question whether $\widehat{p}_2$ is the threshold function remains open but the value $\widehat{p}_2$ for $k = 2$ matches the threshold stated for square bond percolation from H. Kesten [17].

5 Conclusion

Motivated by the concept of robustness communication in the reliability networks, we have designed the protocols for end-to-end induced shortest path communication problems. We have also stated the necessary conditions for a faulty-free communication with respect to the defined communication model. The question whether the values $\widehat{p}_1$ and $\widehat{p}_2$ are the threshold functions remains open. (See thereinbefore.)

Our results together with the works of M. Ajtai, et al. [3], H. Kesten [17], M. Nehéz and D. Bernát [31] represents also a contribution to the detailed investigation of phase transition behavior of random H-type graphs in their critical phase. The future works may concern e.g. the generalization to other topologies and also a more accurate analysis of the end-to-end communication protocols.

Acknowledgement

This research has been supported by MŠMT under research program MSM 6840770014.

References

[1] M. Adler, F. Fich: *The Complexity of End-to-End Communication in Memoryless Networks*, In Proc. 18th ACM Symposium on Principles of Distributed Computing, PODC'99, 239–248.

[2] Y. Afek, H. Attiya, A. D. Fekete, M. Fischer, N. Lynch, Y. Mansour, D. Wang, L. Zuck: *Reliable communication over unreliable channels*, Journal of the ACM, **41** (1994), No. 6, pp. 1267–1297.

[3] M. Ajtai, J. Komlos, E. Szemeredi: *Largest random Component of a k-Cube*, Combinatorica, **2** (1), 1982, 1–7.

[4] N. Alon, P. Erdös, J. Spencer: *The Probabilistic Method,* John Wiley & Sons, New York, 1992.

[5] F. Annexstein, M. Baumslag, A. Rosenberg: *Group Action Graphs and Parallel Architectures*, SIAM J. Computing, **19** (1990), No. 3, 544–569.

[6] H. Attiya, J. Welch: *Distributed Computing: Fundamentals, Simulations and Advanced Topics*, McGraw-Hill, London, 1998.

[7] B. Bollobás: *Random Graphs*, Academic Press, New York, 1985.

[8] S. Dobrev: *Yet Another Look at Structural Information*, In Proc. 5th Int. Colloquium on Structural Information & Communication Complexity SIROCCO'98, L. Gargano and D. Peleg (Eds.), Carleton Scientific, 1998, pp. 114–128.

[9] W. Feller: *An Introduction to Probability Theory and Its Applications*, John Wiley & Sons, New York, 3rd Edition, 1968.

[10] F. Fich: *End-to-End Communication*, OPODIS 1998, 37–44.

[11] P. Fraigniaud, C. Gavoille: *Lower Bounds for Oblivious Single-Packet End-to-End Communication* In Proc. 17th Int. Symposium on Distributed Computing, DISC 2003, LNCS 2848, Springer-Verlag, Berlin Heidelberg, 211-223.

[12] A. Goerdt: *The Giant Component Threshold for Random Regular Graphs with Edge Faults*, In Proc. 22nd Int. Symposium on Math. Foundations of Comp. Sci., MFCS'97, LNCS 1295, Springer-Verlag, Berlin Heidelberg, 279–288.

[13] A. Goerdt: *The Giant Component Threshold for Random Regular Graphs with Edge Faults*, Theor. Comp. Sci. **259** (2001), 307–321 (full version of [12]).

[14] A. Goerdt: *Random regular graphs with edge faults: Expansion through cores*, Theor. Comp. Sci. **264** (2001), 91–145.

[15] M.-C. Heydemann, B. Ducourthial: *Cayley graphs and interconnection networks*, In Graph Summetry, G. Hahn and G. Sabidussi (Eds.), vol. 497 of NATO ASI C, Kluwer Academic Publishers, 1997, 167–226.

[16] S. Janson, T. Luczak, A. Rucinski: *Random Graphs*, John Wiley & Sons, New York, 2000.

[17] H. Kesten: *The Critical Probability of Bond Percolation on the Square Lattice Equals* 1/2, Communication in Mathematical Physics, **74** (1980), 41–59.

[18] W. Kinzel: *Directed Percolation*, In Percolation Structures and Processes, G. Deutscher, R. Zallen, J. Adler (Eds.), Bristol, England: Adam Hilger, 1983, 425–445.

[19] R. Ladner, A. LaMarca, E. Tempero: *Performance of Counting Protocols for Reliable End-to-End Sequence Transmission*, Journal of Computer and Systems Science, **56** (1998), No. 1, 96–111.

[20] F. T. Leighton: *Introduction to Parallel Algorithms and Architectures: Arrays, Trees, Hypercubes*, Morgan Kaufmann Publishers, San Mateo, California, 1992.

[21] L. Lin: *Balanced Generalized Hypercubes: Complexity and Cost/Performance Analysis*, Int. J. of High Speed Computing, **10** (1999), No. 4, 379–397.

[22] T. Luczak: *Phase transition phenomena in random discrete strctures*, In Proc. on Europ. Conf. of Comb., Graph Theory and Applications, EUROCOMB 2003, ITI Charles University, Praha, 2003, 16.

[23] M. Mačaj: *Vertex transitive graphs and 2-transitive groups I: Injective mappings*, Manuscript, submitted to publication, 2002.

[24] Y. Mansour, B. Schieber: *The intractability of bounded protocols for non-FIFO channels*, Journal of the ACM, **39** (1992), No. 4, 783–799.

[25] M. Markošová: *Small World Networks* (in Slovak), In Proc. Congnition and Artificial Life III, elfa, Košice, 2003, 227–232.

[26] M. Markošová, J. Antala: *Directed site percolation on the square lattice - Pascal like triangle approach*, unpublished manuscript, 2004.

[27] S. Nikoletseas, K. Palem, P. Spirakis, M. Yung: *Short Vertex Disjoint Paths and Multiconnectivity in Random Graphs: Reliable Network Computing*, In Proc. 21st Int. Colloquium on Automata, Languages and Programming, ICALP'94, LNCS 820, Springer-Verlag, Berlin Heidelberg, 1994, 508–515.

[28] S. Nikoletseas, K. Palem, P. Spirakis, M. Yung: *Connectivity Properties in Random Regular Graphs with Edge faults*, Int. J. Foundation Comp. Sci., **11** (2), 2000, 247–262 (full version of [27]).

[29] M. Nehéz: *On Geometrical Properties of Random Tori and Random Graph Models*, Journal of Electrical Engineering, **51** (2000), No. 12/s, 59–62.

[30] M. Nehéz: *The Compactness Lower Bound of Shortest-path Interval Routing on $n \times n$ Tori with Random Faulty Links*, Tech. Report, KAM-DIMATIA Series, Charles University, Praha, **582**, 2002, 1–19.

[31] M. Nehéz, D. Bernát: *On Communication Protocols in Unreliable Mesh Networks and their Relation to Phase Transitions*, In Proc. 17th ISCA Int. Conf. on Parallel and Distributed Computing Systems PDCS 2004, ISCA Cary 2004, 235–240.

[32] J. Postel: *Internet protocol*, Network Working Group RFC 791, 1981.

[33] D. Stauffer, A. Aharony: *Introduction to Percolation Theory*, 2nd ed. London: Taylor & Francis, 1992.

[34] G. Tel: *Introduction to Distributed Algorithms*, Cambridge University Press, 2nd Edition, 2000.