Received November 5, 2021, accepted November 28, 2021. Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2021.3131635

Adaptive Dynamic Shortest Path Search Algorithm in Networks-on-Chip Based on Circulant Topologies

OLEG G. MONAKHOV[1], EMILIA A. MONAKHOVA[1], ALEKSANDR YU. ROMANOV[2], (Member, IEEE), ANDREJ M. SUKHOV[2], (Member, IEEE), AND EVGENII V. LEZHNEV[2], (Member, IEEE)

[1]Institute of Computational Mathematics and Mathematical Geophysics SB RAS, 630090 Novosibirsk, Russia
[2]HSE University, 101000 Moscow, Russia

Corresponding author: Aleksandr Yu. Romanov (a.romanov@hse.ru)

The research leading to the results obtained by Aleksandr Yu. Romanov, Andrej M. Sukhov and Evgenii V. Lezhnev has received funding from the Basic Research Program at the National Research University Higher School of Economics (HSE University). The work by Oleg G. Monakhov and Emilia A. Monakhova was carried out under state contract with ICMMG SB RAS (0251-2021-0005).

ABSTRACT A new pair routing algorithm for transmitting messages in multiprocessor systems and networks-on-chip based on circulant networks of arbitrary dimension is proposed. It allows using all reserve shortest paths in the presence of destructive factors (deadlocks, livelocks, starvation, failures) at the nodes and channels of the communication network. A distinctive feature of the proposed algorithm is the absence of using the routing tables with fixed shortest paths when message is transmitted. It becomes possible to determine the set of the shortest paths for routing due to the relative addresses of destination nodes based on a parametric description of the network. Estimates of the number of reserve shortest paths are obtained, and an effective algorithm for using these paths to prevent dynamic topology changes and network congestions is proposed. To reduce the required memory in networks-on-chip with a circulant topology, we proposed a version of the routing algorithm for two-dimensional optimal circulants. We experimentally found the minimum number of reference nodes (nodes containing mapping tables) for them and estimates of memory for mapping tables, as well as the average path length for the routing algorithm using the reference nodes.

INDEX TERMS Pair routing algorithm, reserve shortest path, circulant network, network-on-chip.

I. INTRODUCTION

Over the past 10 years, the performance of computing systems is no longer so dependent on their maximum clock frequency as on the number of computing cores that fit into a chip. The development of multiprocessor systems-on-chip (MPSoCs) has become a ubiquitous trend and has led to the fact that modern chips can accommodate tens, hundreds or even thousands of processor cores. So, the latest versions of the WSE2 chip from Cerebras can contain up to 850,000 computing cores [1], [2], and the project from Esperanto technologies promises 1088 energy-efficient ET-Minion 64-bit RISC-V each with a vector/tensor unit in ET-SoC-1 chip [3].

The operation of such large MPSoCs is not possible without a high-performance communication subsystem, the tasks of which are currently performed by the network-on-chip (NoC). NoCs represent many interconnected levels of abstraction [4], including the router level, the IP core interface level, the data transfer protocol level, the routing algorithm level, the network node connection topology level, etc. Effective implementation of each of these levels affects the performance of the entire network as a whole.

This work is devoted to one of the most important levels of NoCs implementation – routing algorithms. The fact is that the routing methods used in local computer networks are not applicable in NoCs primarily due to the inability to store large routing tables in routers [5]. As for routing methods from the field of wireless sensor networks [6]: although they have a number of advantages useful for NoCs, they cannot be used without adaptation to the NoC features.

It is very important that the routing algorithm implements the greedy forwarding principle, i.e. ensures the advancement of the package along the shortest path [7]. Although this is

The associate editor coordinating the review of this manuscript and approving it for publication was Gian Domenico Licciardo.

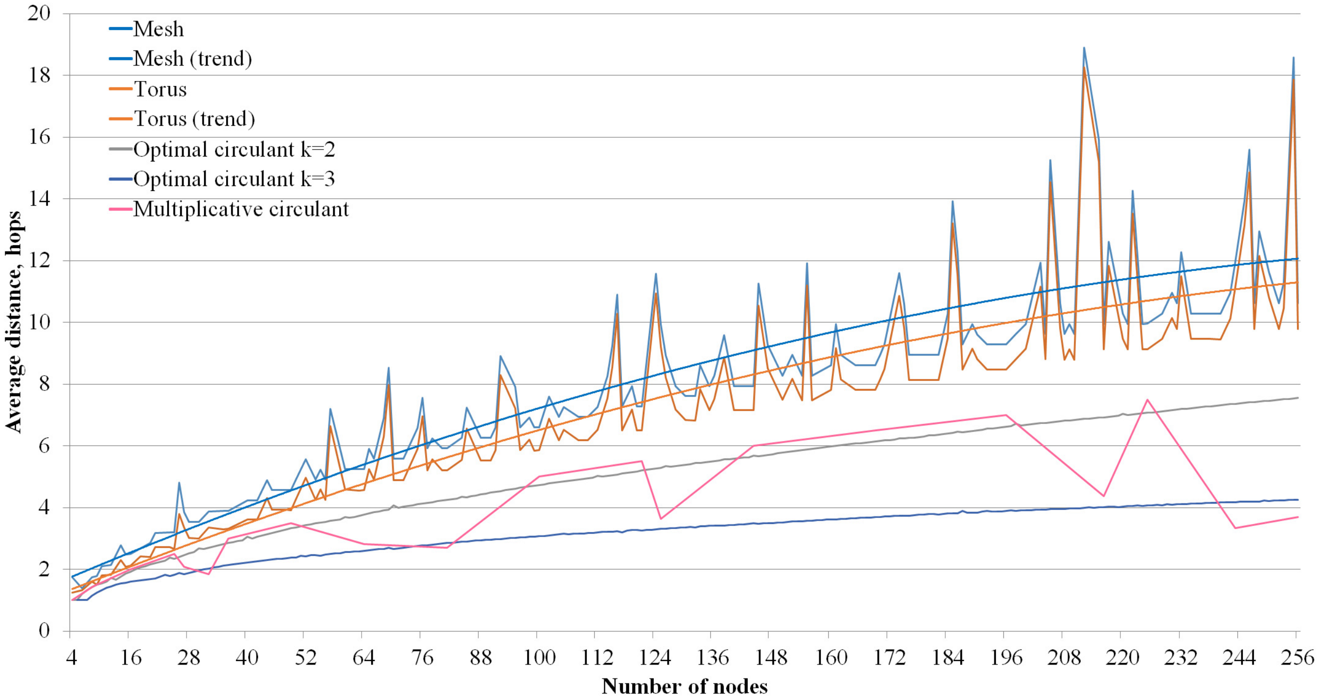

FIGURE 1. Dependence of the average distance between nodes on the number of nodes for mesh, torus and circulant topologies.

not always implemented [5], [8] (for example, for the sake of reducing hardware costs), it ultimately leads to an actual increase in the packet forwarding delays.

To solve the problem of finding the shortest path from a source vertex to all other vertices of an arbitrary connected graph (without edges of negative weight), Dijkstra's algorithm [9] (a greedy algorithm) and similar algorithms (for example, a wave algorithm in which the weights of all edges are equal to one unit) are often used. Let all N vertices of a connected undirected graph modeling a communication network be numbered from 0 to $N-1$. We have to find the shortest (i.e., using the least number of edges) path from a vertex to any other vertex $v, 0 \leq v < N$. The result of Dijkstra's algorithm in its classical version is the length of the shortest path or the path itself as a sequence of numbers of output directions (which are associated with certain edges of the graph) of all vertices belonging to one of the shortest paths to the vertex v. Dijkstra's algorithm has a time and memory complexity estimate of $O(N^2)$, which corresponds to the number of elements in the adjacency matrix, and in addition, it determines one fixed shortest path from any vertex of the graph to any other one.

In the context of the impact of destructive factors (deadlock, livelock, starvation, failure) [10]–[12] on the communication network, it becomes important to use the reserve shortest paths of the communication network, which provide the opportunity to bypass the blocked regions of the network. To solve the problem of the set of the shortest path search and developing a fault-tolerant routing algorithm, we propose a modification of the breadth-first search (BFS) algorithm [13]. The adaptive dynamic pair exchange algorithm based on it can be used for any subclasses of regular graphs with a parametric description [14]–[19] used as models of communication networks of multiprocessor systems and NoCs. The problem of effective organization of a communication subsystem in NoCs with hundreds of processor cores is becoming urgent. In this paper, we study the applicability of the new routing algorithm for the class of circulant graphs (networks) [15], [20]–[22] considered as a NoC topology. We use circulant graphs in NoC topologies because classical regular topologies such as mesh, hypercube, and torus do not always meet modern requirements for NoCs with an increase in the number of nodes. It makes it possible to improve characteristics of communication subsystems of NoCs such as the diameter, average distance between the nodes, and reliability in comparison with standard mesh and torus NoC topologies [5].

Fig. 1 shows the changes in the average distance between nodes depending on the number of nodes for mesh, torus and various classes of circulant topologies. They clearly demonstrate the advantages of circulant topologies over classical regular ones. The curves for the diameters of the topologies look similar and are, therefore, not shown here. For circulants, the change in characteristics occurs smoothly, regardless of the number of nodes, which cannot be said about mesh and torus topologies. Although they are classified as regular topologies [4], strictly speaking, they are conditionally regular and have the best topological characteristics only for a square form, when the number of nodes is a power of a natural number. Moreover, for a number of cases (for example, when the number of nodes is a prime number), mesh and torus topologies generally degenerate into linear and ring topologies with unsatisfactory characteristics [4], as a result of which they are not shown at all on the graphs. The presence of outliers on mesh and torus plots corresponds to cases where the shape of the topology is very far from square; for example, for $N = 226 = 2 \times 113$.

Thus, circulants look like a real alternative to the classical topologies used in NoCs, while the presence of different classes of circulants provides the developer with a field for choosing the appropriate topology option for a specific task. For example, Fig. 1 shows a curve of average distance for the class of multiplicative circulants [10], which are not available for every number of nodes, but have a number of useful properties; the choice between optimal circulants with different dimensions can be determined by a trade-off between hardware costs and required bandwidth; ring circulants [5] (not shown in Fig. 1 because their curves, with the exception of some points, almost coincide with those for optimal circulants) can be chosen to simplify routers and routing algorithm, etc.

The idea of using circulant graphs as a topology for a communication subsystem in NoCs appeared recently and became possible due to the significant progress in field-programmable gate array (FPGA) and application-specific integrated circuit (ASIC) chips, as well as the evolution of computer-aided design software (CADs). Currently, there is no need to bind to rigid planar floor planning tiles of NoC area for NoC design, as this is done by automated means. Thus, it becomes possible to move the design of NoC communication subsystem to a higher level of abstraction and use new topological solutions, including circulant graphs [5], [8].

Let us define circulant networks. Let $k, N \in \aleph$, $S = \{s_1, s_2, \ldots, s_k\}$ be the set of integers, where $1 \leq s_1 < \ldots < s_k < N$. An undirected circulant graph $C(N; S)$ has a set of vertices $V = Z_N = \{0, 1, \ldots, N-1\}$ and a set of edges $E = \{(v, v \pm s_i (mod\ N)) | v \in V, i = \overline{1, k}\}$. The numbers $s_1, s_2, \ldots, s_k$ are called generators of the graph, N is its order, k is its dimension, the degree of a vertex of the graph is $2k$ if $s_k \neq N/2$, or $2k - 1$ otherwise. The diameter of the graph (which estimates the maximum structural delay in the network) is $d = max_{u,v \in V} D(u, v)$, where $D(u, v)$ is the length of the shortest path between the vertices u and v. The average distance in the graph is also an important criterion for the functioning of communication networks.

It should be noted that circulant graphs of dimensions 2 and 3 are used as topologies of multiprocessor systems [15], [20], [21], and also as promising topologies of NoCs [5], [8].

Modern chips already can have more than 1000 cores [1]–[3]. For the most part, they are still connected by mesh topology, but at the same time (due to the large number of nodes), the distance between nodes is too large, which is to be corrected by global ring connections (for example, over a wireless channel [23], [24]). This state of the problem opens up opportunities for applying circulant topologies with dimensions $k \geq 3$ in NoCs. Furthermore, the development of new efficient routing algorithms in circulant networks, which use the shortest paths for information flows routing, is relevant.

Work [7] provides an extensive overview of routing algorithms used in NoCs with classical topologies (primarily, mesh and torus) and the problems that arise during routing. The variety of approaches and routing methods suggests that there are still no universal solutions that would satisfy all the requirements of developers. In addition, circulant topologies have their own specificity consisting in their symmetry and regular structure [20], [21], which should be taken into account when developing routing algorithms.

In [25] the authors proved that the problem of finding the shortest paths using a parametric description of circulants of arbitrary dimension, when the value of k is not fixed, is an NP-hard problem. Note that known in the literature pair routing algorithms have been developed for a number of separate families of circulants of dimensions $k = 2, 3$ with an analytical description [8], [10], [26]–[31], for symmetric 3D networks [32]. For circulants of large dimensions $k > 3$, two papers were known earlier. In [10] an algorithm for finding the shortest paths for a family of multiplicative circulants was developed (this algorithm is non-adaptive, since it always uses the fixed shortest path). In [33] a heuristic algorithm of computation of the shortest paths for arbitrary circulant graphs of dimensions $k > 2$ was presented. Since the exact solution for finding the shortest paths is related to the solution of Diophantine equations, in [33] the so-called restricted shortest paths were introduced. After testing the algorithm for circulants of dimensions 3 and 4 and calculating about 70000 restricted shortest paths, the authors of [33] obtained the optimal results (the shortest paths) in 85 percent of cases.

The advantage of the routing algorithm developed in this paper in contrast to algorithms from [5], [8], [10], [26]–[32], [34]–[36] is that it is applicable for circulant networks of a general form and any dimensions $k \geq 2$. This algorithm is also optimal, since it always calculates the shortest paths, and it is also adaptive in case of destructive factors. Due to the using of the reserve shortest paths, the algorithm gives a possibility to maintain the packet transmission through less-congested paths in case of failures, deadlocks, livelocks, and starvation.

The rest of the paper is organized as follows. In Section 2, we present the new algorithm for calculating the shortest path vectors for any graphs from the class of circulant networks and estimate the time complexity of the algorithm. In Section 3, we describe some properties of a set of the shortest path vectors in circulant graphs used further to form mapping tables of the numbers of vertices and shortest path vectors leading to them. An adaptive pair exchange algorithm in circulant networks based on mapping tables is represented in Section 4. In Section 5, a variant of the distribution in circulants of the minimum number of reference vertices containing mapping tables is proposed in order to

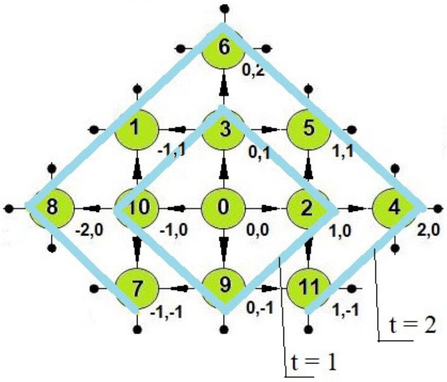

FIGURE 2. An example of the results of execution of Algorithm 1.

reduce the required memory in NoCs, and estimates of their number and an increase in the average distance in graphs when organizing pair exchanges are given. In Section 6, we also give an estimation of the number of reserve paths in circulant networks when a node (or a communication line) is deleted.

II. ALGORITHM FOR CALCULATING THE SHORTEST PATH VECTORS

For a vertex $v, 0 < v < N$, of a circulant graph $C(N; S)$ of dimension k, let us define the shortest path vector from the vertex 0: $P(0, v) = (p_1(v), p_2(v), \ldots, p_k(v))$ as a vector with k coordinates, where $|p_i(v)|, i = \overline{1,k}$, specifies the number of steps (hops) along the edges corresponding to the generator s_i $(-s_i)$ in the shortest path from 0 to v. The + (−) sign specifies the movement direction along s_i $(-s_i)$. We have $D(0, v) = \sum_{i=1}^{k} |p_i(v)|$. There is an obvious property of the shortest paths in the circulant graphs: a shortest path between any two vertices of the graph cannot simultaneously contain s_i and $-s_i$ for any $i = 1, \ldots, k$. Since the circulant is a vertex-transitive graph, $D(0, v) = D(0, N - v)$, $P(0, v) = -P(0, N - v)$, and it is sufficient to consider only the values $0 \le v \le \lfloor N/2 \rfloor$ and 0 as the source vertex in pair exchanges. The generalization of the definition of a shortest path vector $P(i, j)$ from source vertex i to destination vertex j is obvious.

For a circulant network $C(N; S)$ of dimension k, where $1 \le s_1 < \ldots < s_k \le \lfloor N/2 \rfloor$, we present an algorithm for calculating the shortest path vectors from one vertex to all the others. The algorithm is based on the BFS procedure. The input parameters of the algorithm are: the description of the circulant network (the number of vertices N and the list of generators S) and the number of the source vertex. The result of the algorithm is a list of all the shortest path vectors leading from the initial vertex to all the other vertices of the graph. This list we will call a mapping table of the numbers of vertices (absolute) and shortest path vectors leading to them.

Note that the subroutine addSPVneighbor (spv, v, w) allows obtaining all the modified shortest path vectors for the vertex w from the neighboring vertex v with the coordinates changed by one for the corresponding generator.

Algorithm 1 (Algorithm for Calculating the Shortest Path Vectors)

Let s be the source vertex. It receives the shortest path vector with zero coordinates: $P(0, 0)$, where $|p_i(0)| = 0$, $i = \overline{1,k}$. At each step, the algorithm adds all the vertices not belonging to T (T is a shortest path tree rooted at s and partially formed by already checked vertices), adjacent to those vertices from T that were added to T at the previous step, to already checked vertices. When adding a vertex w connected by an edge (w, v) with a mark $s_i = w - v(\text{mod } N)$ with a vertex $v \in T$, the vertex w receives all the shortest path vectors from the vertex v with the coordinates changed by one (increased by one in absolute value) for the generator s_i. The algorithm terminates when all the vertices of the graph are included in the tree T, which connects all the vertices of the graph by the shortest paths to s. This happens after d steps were completed, where d is the diameter of the graph.

If it is required to find the shortest paths only to the vertex i, then the modified version of the algorithm terminates after adding the vertex i to T.

The program for Algorithm 1 in a generalized algorithmic language (whose syntax is similar to that of Python) is given below.

The program for Algorithm 1

```
procedure all_shortest_path_vectors (C, N, sk, source):
  # C – circulant graph
  # N – number of nodes
  # sk – list of generators
  # source – number of source node
  # Returns spv – shortest path vectors for
  # the paths from source to all nodes in circulant C
  pv = { }            # spv dictionary
  for s in sk:
    pv[s] = 0
  level = 0           # the current level when seen in BFS
  sp v = [[] for i in range(N)] # list of all spv for all the nodes
  spv[source].append(pv)
  level = 0           # the current level
  nextlevel = [source]      # list of nodes to check at next level
  seen = {source: level}    # level (number of hops) when seen in BFS
  while nextlevel:
    level = level + 1
    thislevel = nextlevel
    nextlevel = []
    for v in thislevel:
      for w in C[v]:
        if w not in seen:
          seen[w] = level
          nextlevel.append(w)
          spv = addSPVneighbor(spv, v, w)
        elif (seen[w] == level):
          spv = addSPVneighbor(spv, v, w)
  return spv
```

Fig. 2 shows an example of how Algorithm 1 works for a circulant network $C(12; 2, 3)$. Inside each vertex, its number

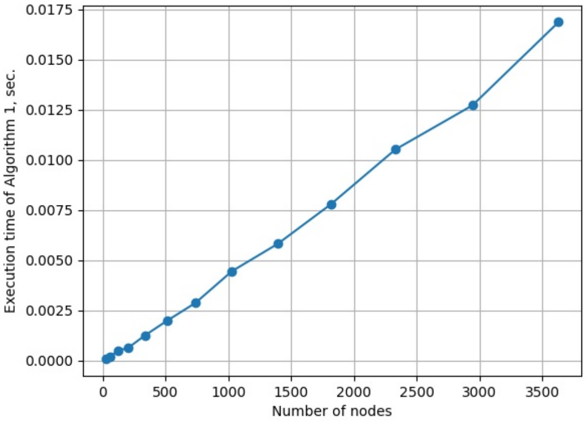

FIGURE 3. **Dependence of the execution time of Algorithm 1 on the number of nodes for 3D-circulants.**

is indicated; on the right and below of each vertex, its shortest path vector is indicated; on the lines t = 1 and t = 2, the vertices traversed in the first two steps of the algorithm are shown.

When estimating the total time complexity of the algorithm, it should be taken into account that each graph vertex is visited once and each its incident edge is viewed from this vertex once. Therefore, the time complexity of the algorithm for a circulant $C(N; S)$ of dimension k can be estimated as $O((2k - 1)N)$, and under the condition $k = const$, the time complexity of the algorithm for calculating the shortest path vectors from one vertex to all the others is of $O(N)$. Fig. 3 demonstrates the experimentally obtained linear dependence of the execution time of Algorithm 1 on the number of vertices for circulant graphs with $k = 3$. The family of 3D-circulants [30] with the maximum number of vertices N for any given diameter (Sect. 6, formula (4)) was used as an experimental model of the communication network. Fig. 3 shows the average values of the execution time of Algorithm 1 (a random selection of the source node for 2000 experiments was used).

III. PROPERTIES OF THE SHORTEST PATH VECTORS IN CIRCULANT NETWORKS

There are some well-known properties of a set of the shortest path vectors in circulant graphs, which will be used further to form mapping tables and develop an adaptive pair routing algorithm. These properties are generalized in Lemmas 1 and 2. Due to the vertex transitivity of circulants [15], the problem of computing the shortest path between vertices i and j is reduced to the problem of finding the shortest path between vertices 0 and $w = (j - i) \bmod N$ (Fig. 4). So, we have

Lemma 1: For any pair source — destination with numbers (i, j) *, a set of the shortest path vectors coincides with a set of the shortest path vectors of the pair source — destination with numbers* $(0, (j - i) \, mod \, N)$.

To prove Lemma 2, we use the following property of circulants: the shortest paths from 0 to a vertex i and from 0 to the vertex $N - i$ differ only in signs in front of the generators, that is, in our case, this property is $P(0, i) = -P(0, N - i)$. We can call this property the mirror symmetry of vertices about the ‘‘axis’’ passing through 0 and $N/2$. Now we simply generalize this property for any vertex i, drawing this axis of symmetry through i and $(i + N/2) \, mod \, N$ and obtain the result of Lemma 2.

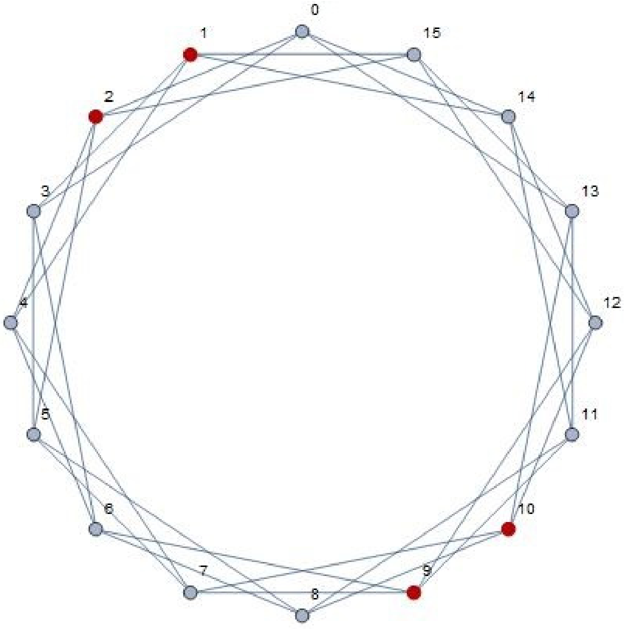

FIGURE 4. **The distribution of reference nodes in the circulant** $\boldsymbol{C(16; 2, 3)}$**.**

Lemma 2: For any pair source — destination with numbers (i, j)*, where* $j \leq (i + \lfloor N/2 \rfloor) \, mod \, N$*, a set of shortest path vectors coincides with a set of shortest path vectors taken with the opposite signs of the pair source — destination with numbers* (i, w)*, where* $w = (2i - j) \;\; mod \, N$.

Thus, the symmetry property of circulant graphs implies the following: (1) having the shortest path vector (vectors) from vertex 0 to any vertex v, $0 < v < N$, by simple cyclic transformation (shift), it is possible to obtain the shortest path vector (vectors) for any pair source – destination without reusing Algorithm 1; (2) having the shortest path vector (vectors) from vertex i to any vertex $j \leq (i + \lfloor N/2 \rfloor) \bmod N$, by changing the signs of the components of the vector (vectors) to the opposite ones, it is possible to obtain the shortest path vector (vectors) from vertex i to any destination vertex $w = (i + i - j) \bmod N$, which allows cutting mapping tables in half, that is for any vertex i, it is possible to store data for vertices with numbers from i to $(i + \lfloor N/2 \rfloor) \bmod N$ only.

As already noted, in conditions of the impact of destructive factors on the communication network, it is important to have reserve shortest paths in the network which make it possible to bypass blocked regions without path length increasing. Further, we will distinguish between *alternative* shortest paths from 0 to v given by different not

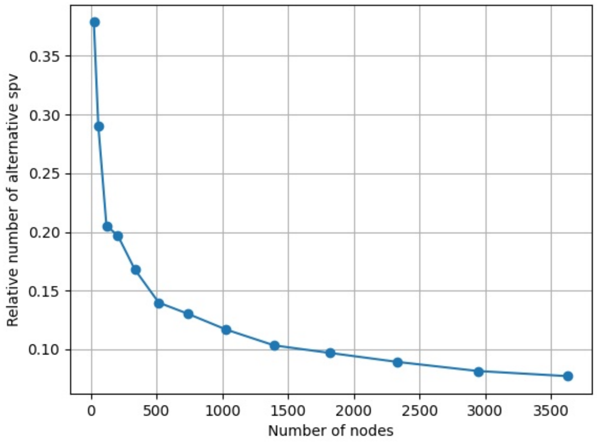

FIGURE 5. Ratio of the number of alternative shortest paths to the number of nodes for 3D-circulants.

coincident vectors

$$\Psi = \left\{P^1(0,v), \ldots, P^q(0,v)\right\}, \quad q \geq 1,$$

and *reserve* (equivalent) shortest paths obtained from one alternative vector $P^i(0,v)$ and differ in the order of passage of the generators included in the shortest path from 0 to v.

For example, in the circulant $C(16; 2, 3)$ (Fig. 4) there are two alternative shortest paths from node 0 to node 7. The first alternative shortest path vector $P^1(0,7)$ contains one path $0 \rightarrow 13 \rightarrow 10 \rightarrow 7$ with generators $-3, -3, -3$. The second vector $P^2(0,7)$ contains three reserve paths: $0 \rightarrow 2 \rightarrow 4 \rightarrow 7$ with generators $+2, +2, +3$, $0 \rightarrow 2 \rightarrow 5 \rightarrow 7$ with generators $+2, +3, +2$, and $0 \rightarrow 3 \rightarrow 5 \rightarrow 7$ with generators $+3, +2, +2$.

The function $q = q(N, S, v)$ depends on the graph description and the destination vertex number and requires the further study. Here, we will give only experimental estimates for $q(N, S, v)$, namely, the ratio of the number of alternative shortest paths to the number of vertices for the graphs of the 3D-circulants family with the order N given by formulas (4). Fig. 5 demonstrates the average values of $q(N, S, v)$ obtained with a random selection of the destination vertex for 2000 experiments.

As studies on the approximation of the obtained dependence have shown, it is inversely proportional to the diameter of the graph ($O(1/d)$). For a given family of circulants, this is explained by the ratio of the number of vertices to the diameter d, which determine the presence of alternative paths in the graph and have the estimate $O(d^2)$ to the total number of vertices in the graph of order $O(d^3)$.

The cardinality of the set of reserve shortest paths from vertex 0 to vertex v for each individual representative of the alternative paths in v can be easily calculated. Let $Q_k(P(0,v))$ denote the total number of reserve shortest paths from 0 to the vertex v to which corresponds the shortest path vector $P(0,v) = (p_1(v), p_2(v), \ldots, p_k(v))$. Since circulant graphs are Cayley graphs of Abelian groups, the value of $Q_k(P(0,v))$ for the vertex v is defined as follows. Consider the case of $k = 2$. In a circulant graph, any permutation of connections along the path from 0 to a vertex v, where $P(0,v) = (p_1(v), p_2(v))$, gives the same path from 0 to v. Thus, a total number of

$$\binom{|p_1(v)| + |p_2(v)|}{|p_1(v)|} = \frac{(|p_1(v)| + |p_2(v)|)!}{|p_1(v)|!|p_2(v)|!}$$

different shortest paths from 0 to v exist, consisting of $p_1(v)$ steps of type s_1 and $p_2(v)$ steps of type s_2. This formula is equal to the binomial coefficients. Lemma 3 gives a generalization of the case of $k = 2$ to any k.

Lemma 3: In a circulant graph $C(N; s_1, \ldots, s_k)$, for any vertex v, $0 < v < N$, let $P(0,v) = (p_1(v), p_2(v), \ldots, p_k(v))$ be the shortest path vector from 0 to v, then $Q_k(P(0,v))$ is equal to:

$$Q_k(P(0,v)) = \left(\sum\nolimits_{i=1}^{k} |p_i(v)|\right)! / \prod\nolimits_{i=1}^{k} |p_i(v)|! \qquad (1)$$

Thus, a compact form of representing the set of shortest paths as a shortest path vector, allows specifying the total set of reserve shortest paths (differing only in the order of passage of the generators) without additional memory costs for storage of reserve paths. Our dynamic approach to the routing algorithm gives a possibility of switching among different alternative shortest paths and makes it more flexible in avoiding network destructive factors.

According to Fig. 5, for some source – destination pairs, there may be several alternative shortest path vectors each of which specifies its own subset of the reserve shortest paths. The choice of the most promising vector (from the point of view of providing the largest number of reserve paths when moving along the shortest path) is determined on the basis of Lemma 4 derived from (1).

Lemma 4: Let $P(0,v) = (p_1(v), p_2(v), \ldots, p_k(v))$ be a shortest path vector from 0 to any vertex v, $0 < v < N$, of a circulant graph of order N and dimension $k \geq 2$, and let $|p_1| > |p_2| \geq \ldots \geq |p_k|$. Then, for any j for which $p_j \neq 0$, we have $Q_k(|p_1|-1, p_2, \ldots, p_k) > Q_k(p_1, \ldots, |p_j|-1, \ldots, p_k)$.

Proof: Since only absolutevalues are used in (1), we can, without loss of generality, consider only non-negative p_i, $i = 1, \ldots, k$. Let $j = 2, p_2 > 0$. Le's compare the value of $Q_k(p_1 - 1, p_2, \ldots, p_k)$ with the value of $Q_k(p_1, p_2 - 1, \ldots, p_k)$. These functions have a common numerator equal to $p_1 + p_2 + \ldots + p_k - 1$, but the first one has a lower denominator provided $p_1 > p_2$. Q.E.D.

This implies the possible strategy of choosing when routing a packet, such a coordinate of the shortest path vector, decreasing which by 1 preserves the greatest number of reserve paths when moving along the shortest path. This, according to Lemma 4, is the coordinate with the maximum absolute value. If the set of vectors of alternative shortest paths $\Psi = \{P^1(0,v), \ldots, P^q(0,v)\}$ calculated at the node of the source of the message contains more than one vector of paths to v, then the choice from Ψ is made according to the criterion of the presence of a potentially larger number of possible reserve paths, that is, a vector

$P(0, v) = (p_1, p_2, \ldots, p_k) \in \Psi$ is chosen having

$$min \sum\nolimits_{i \neq j} \left| |p_i| - |p_j| \right| = \min \{\max_k |p_k| - \min_k |p_k|\}. \quad (2)$$

We will call this strategy the method of flexible descent along the coordinates of the shortest path vector.

IV. ADAPTIVE PAIR EXCHANGE ALGORITHM IN CIRCULANT NETWORKS

The process of obtaining the shortest path vectors at all vertices (network nodes) and formation of mapping tables for organizing pair exchanges in the network is as follows.

1) Algorithm 1 is executed in one node with number 0 in which, as a result of the operation of this algorithm, a mapping table between the number of the destination node and the shortest path vectors to it from the source node 0 is formed. In this case (using Lemma 2), it is possible to store only half of the table with the numbers of vertices up to $N/2$.
2) The mapping tables for all the other nodes are obtained from the table built at node 0 based on the cyclic transformations according to Lemma 1.
3) If there are several alternative shortest path vectors for storage in the mapping table, it is advisable to choose a vector that has a potentially larger number of possible reserve paths according to criterion (2).

Note that the mapping table between the number of the destination and the shortest path vectors to it from the source is used only at the stage of generating the message header and is not used during its transmission.

The estimation of the required memory for mapping tables grows linearly with the number of nodes in the system. Moreover, the shortest path vector contains the entire set of reserve shortest paths. In addition, we have proposed a method for reducing the number of mapping tables in the system by using so-called reference nodes. This method is indicated in Section 5.

Let us consider a routing algorithm that implements message pair exchanges between nodes in multiprocessor systems, including NoCs, given by the communication network topology in the form of a k-dimensional circulant graph, $k \geq 2$. Let it be required to transmit a packet from the 0 node of the data source to a destination node v.

In the absence of destructive factors at the nodes and communication lines, in the source node, according to the number of the destination node, the shortest path vector $P(0, v) = (p_1(v), p_2(v), \ldots, p_k(v))$ between them is taken from the mapping table (or is computed) and is written to the message header.

At each transit node (including the source), on the way to the destination, any of the output directions corresponding to a nonzero coordinate $p_i(v)$ of the shortest path vector is selected in accordance with criterion (2) (or randomly), and the shortest path vector is modified by the operation

$$(p_i(v))' = p_i(v) - sgn(p_i(v)) * 1,$$
$$p_i(v) \neq 0, \quad p_i(v) \in \{p_1(v), \ldots, p_k(v)\} \quad (3)$$

in order to reduce in absolute value the nonzero coordinates of the vector. The modified vector P', where only the value of $p_i(v)$ has changed, is written into the packet header which is transmitted further. The end of the routing procedure is the equality to 0 of all coordinates of the vector P'.

The proposed adaptive pair routing algorithm allows using reserve shortest paths and thus increasing the reliability of the structure of the communication network. Below there is a description of a distributed adaptive pair routing algorithm for circulant graphs of dimension $k \geq 2$ obtained using the BFS procedure described in Section 2 in the presence of failures of nodes or communication lines.

Note that if (when choosing the best variant) the coordinates turn out to be equal in absolute value, then the next coordinate is chosen randomly or taking into account the load (deadlocks) of neighboring nodes.

V. V REDUCING THE NUMBER OF NODES CONTAINING MAPPING TABLES IN NETWORKS WITH CIRCULANT TOPOLOGY

Along with reducing the diameter and average distance between the nodes for NoC topologies, an important characteristic is the amount of memory and logical resources of the chip required to implement the routing algorithm. In order to reduce the required memory in NoCs with a circulant topology when the pair routing algorithm is running, we consider the possibility of reducing the number of such nodes in the circulant network to a minimum, which should contain the mapping tables. We will call them reference nodes. In this case, the following condition must be satisfied: every node not in reference set must have at least one neighbor in the set of reference nodes. Fig. 4 shows the distribution of reference nodes in the circulant $C(16; 2, 3)$ highlighted in red.

In order to start the process of sending a message, the source node, if it is not a reference node, must first transmit the packet header to the neighboring reference node. Thus, the diameter of the network during routing increases by one, and the average distance during transmissions also increases.

For two-dimensional optimal circulants [8], we considered a version of the routing algorithm using the reference nodes and experimentally found the minimum number of reference nodes for them (Fig. 4). We also obtained estimates of memory and the average path length for the routing algorithm in these networks.

The search for the minimum number of reference nodes for the class of circulant graphs was carried out by us in Wolfram Mathematica 10. Table 1 contains a fragment of the results for two-dimensional optimal circulants with the number of nodes $N = m$^2, where $3 \leq m \leq 12$. There are two options of routing protocols: 1) when the cardinality of reference nodes set coincides with the number of graph vertices; 2) when the number of reference nodes is less than the number of graph vertices (a possible minimum of number of reference nodes is shown in Table 1).

Algorithm 2 (Adaptive Pair Routing Algorithm)

Step 1. The algorithm selects a vector from the set of alternative shortest path vectors $\Psi = \{P^1(0, v), \ldots, P^q(0, v)\}$ at the message source node using the destination number v. The choice of the shortest path vector from Ψ is made according to the criterion (2).

Step 2. Having the vector $P(0, v) = (p_1(v), p_2(v), \ldots, p_k(v))$, which provides the largest number of possible reserve paths, at each intermediate node, the algorithm chooses the number of such an output direction belonging to the shortest path to the destination which corresponds to the maximum value of the absolute value of the coordinates of the selected vector. Thus, the vector P' modified by operation (3) specifies the remainder of the shortest path to the destination. In this case, the method of flexible descent along the coordinates of the vector $P(0, v)$ is implemented. The choice of one of the many paths defined by the algorithm can be made with adaptation to the situation on the network. The choice of the number of the output direction can take into account not only the nonzero coordinates of P', but also the current states of adjacent nodes and communication lines (failures, deadlocks, livelocks, starvation).

Step 3. The algorithm ends when all the coordinates of P' are equal to 0.

It should be noted, that the estimate of the time complexity of Algorithm 2 is $O(1)$.

In a simplified version of Algorithm 2, step 1 can be changed as follows:

Step 1. The algorithm calculates one vector of the shortest path $P(0, v) = (p_1(v), p_2(v), \ldots, p_k(v))$ at the node of the message source according to the number of the destination node v.

Let us show the implementation of the flexible descent method along the coordinates of the shortest path vector using the example of a 3D-circulant network $C(117; 1, 6, 9)$ of diameter 8. Suppose, it is required to transmit a message from 0 to the destination node $v = 56$. Using the destination number 56, the algorithm calculates the set of alternative shortest path vectors: $\Psi = \{P^1(0, 56) = (2, 0, 6), P^2(0, 56) = (-1, 2, 5), P^3(0, 56) = (-1, -1, -6)$.

Choice of a vector from Ψ according to criterion (2) gives the vector $P^2(0, 56) = (-1, 2, 5)$. For it, the number of reserve shortest paths to node 56 is $Q_3\left(P^2(0, 56)\right) = 168$ and is the largest among the vectors of Ψ. At each successive step, the algorithm selects the number of the output direction corresponding to the maximum of the module of the coordinate, modifies the shortest path vector, and puts it in the output queue. As a result of the algorithm operation, we have the following chain of transformations of $P^2(0, 56)$: $(-1, 2, 5), (-1, 2, 4), (-1, 2, 3), (-1, 2, 2), (-1, 2, 1), (-1, 1, 1), (-1, 1, 0), (-1, 0, 0), (0, 0, 0)$. The path obtained is marked in red in Fig. 6.

The values for diameters, average distances and required mapping tables memory for the entire network are shown for both options. We can see, for example, for a network with $N = 144$, that the amount of memory for mapping tables is reduced by 4.5 times, and the average distance during routing increases by 14 percent when using 32 reference nodes compared to the case when all 144 nodes have mapping tables.

In this article, we do not consider the impact of the algorithm on hardware costs (for example, occupied memory resources (REG) and logical blocks (ALM)), since this is a large task that requires implementation of NoC model algorithm at the router level in HDL and multiple RTL synthesis of models for a different number of nodes and topology options. This is the subject of further research, but even now it is possible to give a comparative estimation with the algorithms already implemented in NoCs [8].

According to Table 1, it can be argued that the proposed algorithm (in the basic version) will consume approximately 1.5 times less REG than the Table routing algorithm from [8]; and in its reduced reference node version, it will consume less memory than the following algorithms: Pair exchange algorithm, Clockwise algorithm, and Algorithm for circulants of type $C(N; d, d+1)$ [8, Table 5].

No such conclusions can be drawn based on ALM. The time complexity of the proposed algorithm is $O(N)$ at the stage of forming the shortest paths between nodes and $O(1)$ at the stage of message transmission in the network. Thus, when transmitting messages in the network, it is comparable with Table routing algorithm and Pair exchange algorithm from [8]. According to Table 3 from [8], it is also comparable with the algorithm from [28] and it is better than the algorithms from [26], [33].

VI. VI ESTIMATION OF THE NUMBER OF RESERVE PATHS IN CIRCULANT NETWORKS WITH FAILURES

Let us now consider the operation of the routing algorithm in the presence of a single failure of a node (or a communication line). It is clear that if a node fails, then the communication lines adjacent to it also become non-working. We show that due to the commutativity property in a circulant of any dimension when a node (or a communication line) is deleted, the length of the shortest path between two vertices of the graph can increase by at most 2.

Lemma 5: In a circulant graph $C(N; s_1, \ldots, s_k)$, where $k \geq 2$ and $1 \leq s_1 < s_2 < \ldots < s_k < N$, if a vertex (or an edge) is deleted, then the length of the shortest path between two vertices of the graph can increase by at most 2.

Proof: Consider a circulant graph $C(N; s_1, \ldots, s_k)$ of diameter d. When a vertex or an edge of the graph is deleted, the distance from 0 to vertex v can increase only if between 0 and v there is a unique shortest path of length less or equal than d, and the vertex (or edge) deleted is on this path, that is, if the path contains only generators of type s_i(or$-s_i$) for any $i = 1, \ldots, k$. Let it be necessary to get from 0 to vertex u, and there is no loss of generality if we suppose that the shortest path from 0 to u contains a fragment with two generators of type $s_1 : 0 \to s_1 \to 2s_1, \ldots, \to u$. If there are no failures, the length of this fragment is 2.

1) Let the edge be removed between 0 and $v = s_1$. The commutative property of circulants allows constructing $k - 1$ different paths of length 4: $0 \to s_i \to s_i + s_1 \to s_1 \to 2s_1$.

TABLE 1. **Parameters for two variants of the routing algorithm.**

Circulant	Reference nodes (all)	Diameter	Average distance	Memory for mapping tables	Reference nodes	Diameter	Average distance	Memory for mapping tables
$C(9;\ 2,3)$	9	2	1.50	162	3	3	2.34	54
$C(16;\ 2,3)$	16	3	1.93	512	4	4	2.79	128
$C(25;\ 3,4)$	25	3	2.33	1250	5	4	3.20	250
$C(36;\ 4,5)$	36	4	2.86	2592	9	5	3.65	648
$C(49;\ 4,5)$	49	5	3.33	4802	12	6	4.12	1176
$C(64;\ 5,6)$	64	6	3.78	8192	15	7	4.57	1920
$C(81;\ 6,7)$	81	6	4.25	13122	18	7	5.05	2916
$C(100;\ 7,8)$	100	7	4.74	20000	23	8	5.52	4600
$C(121;\ 7{,}8)$	121	8	5.20	29282	27	9	5.99	6534
$C(144;\ 8{,}9)$	144	8	5.65	41472	32	9	6.44	9216

Here, the generators s_i, s_1, $-s_i$, s_1, where $i \in \{2, \ldots, k\}$, are used sequentially.

2) Let the vertex $v = s_1$ be removed. Similarly, we obtain $k - 1$ different paths of length 4: $0 \rightarrow s_i \rightarrow s_i + s_1 \rightarrow s_i + 2s_1 \rightarrow 2s_1$, where $i \in \{2, \ldots, k\}$. Here, the generators s_i, s_1, s_1, $-s_i$ are used sequentially. In both cases, usage of the generators $-s_i$ gives $k - 1$ more different workarounds.

Thus, if an edge (or a vertex) fails once, the total path length from 0 to u can increase by at most 2. Q.E.D.

Let us estimate the increase in the reliability of the network when using the developed routing algorithm that employs reserve paths of circulant topologies. In case of a single failure of a node or communication line, as it was shown, the distance from 0 to node v can increase (by at most 2) only if the shortest path contains generators of the same type s_i (or $-s_i$) for any of the possible $i = 1, \ldots, k$. In a k-dimensional circulant graph of order N and diameter d, the number of such vertices (call them unreserved) is less than or equal to $2kd$. These are the vertices with a single shortest path from 0. The number of vertices that have two or more reserve shortest paths from 0, is $N - 2kd$, respectively. The total number of vertices in a circulant graph of diameter d and dimension k [15], [20], [21] is the following

$$N \leq M\left(d, k\right) = 1 + \sum\nolimits_{i=0}^{k-1} C_k^i C_d^{k-i} 2^{k-i}.$$

Thus, the theoretical estimate of a part of the unreserved vertices of the total number of graph vertices is

$$\delta = 2kd/M(d, k).$$

For $k = 2$ this estimate is attainable [13] and equal to

$$\delta = 4d/(2d^2 + 2d + 1) \sim 2/d.$$

For circulant graphs of dimension $k = 3$ and diameter d, the maximum attainable value of N was given in [30] and N is equal to

$$\begin{cases} 32\mathrm{d}^3/27 + 16\mathrm{d}^2/9 + 2\mathrm{d} + 1, & d \equiv 0(\mathrm{mod}\ 3), \\ 32\left\lfloor \mathrm{d}/3 \right\rfloor^3 + 48\left\lfloor \mathrm{d}/3 \right\rfloor^2 + 30\left\lfloor d/3 \right\rfloor + 7, & d \equiv 1(\mathrm{mod}\ 3), \\ 32\left\lfloor \mathrm{d}/3 \right\rfloor^3 + 80\left\lfloor \mathrm{d}/3 \right\rfloor^2 + 70\left\lfloor d/3 \right\rfloor + 21, & d \equiv 2\ (\mathrm{mod}\ 3)\,. \end{cases} \tag{4}$$

For a family of 3D-circulant graphs of diameter d with the maximum number of vertices N given by (4), the number of vertices with one shortest path from 0 can be estimated as $4d + 2$ or $4d$ ([37, theorem 12], vertices of type T_1 – unreserved vertices). Thus, for 3D-circulants,

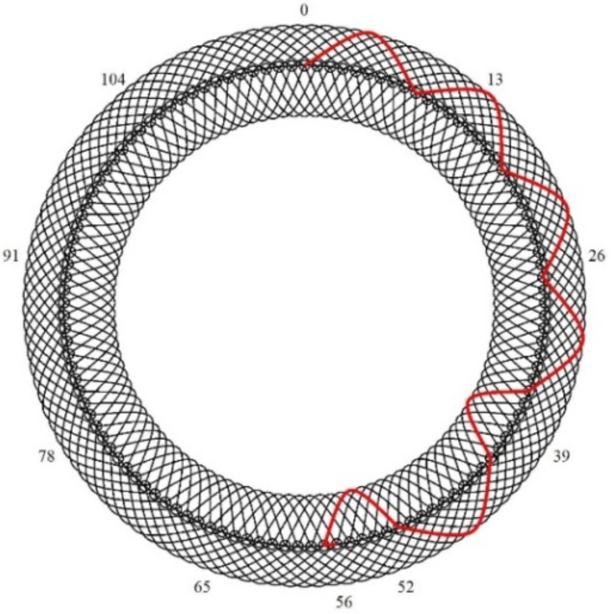

FIGURE 6. **The 3D-circulant $C(117;\ 1, 6, 9)$.**

$$\delta = (4d + 2)/N \sim 3/d^2. \tag{5}$$

Similarly, for a family of circulants of dimension 4 and diameter d with the maximum attainable number of vertices, the estimate for the number of vertices of type T_1 is $4d + 4$ [37, theorem 13], and, accordingly,

$$\delta \sim 8/d^3.$$

In the general case, for arbitrary k, this trend will continue, and $\delta = O(1/d^{k-1})$ tends to 0 fast enough when the diameter of the graph increases.

Fig. 7 shows a plot of variations in a part of unreserved vertices with an increase in the order of graphs for 3D-circulants. The theoretical estimate (5) (shown by line) and the estimate for the family of 3D-circulants with the maximum number of vertices obtained during the testing of the proposed routing algorithm (shown by points) coincide.

Thus, the proposed algorithm provides necessary freedom to use alternative paths in the event of node failures. Here,

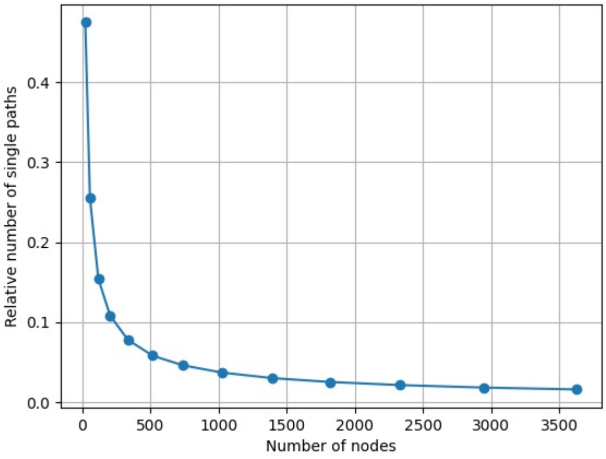

FIGURE 7. Ratio of the number of nodes with one shortest path from 0 to the total number of nodes for 3D-circulants.

we do not consider how the algorithm behaves in deadlock and livelock situations (which are quite likely due to the features of circulants), since this requires modeling the functioning of NoCs using a high-level or HDL model, and this is a separate volumetric task that is not included in the purpose of this work. Nevertheless, it can be argued that the proposed algorithm will be no worse than the classical deterministic algorithms, as it guarantees the shortest paths for the packets to pass. In addition, when a deadlock situation is detected, a group of blocked nodes can be temporarily marked as failed, and this will allow choosing by-pass routes around the blocked section of the network.

VII. CONCLUSION

In this paper, an adaptive dynamical pair exchange algorithm for the class of circulant networks of any dimensions and descriptions is developed for using in multiprocessor systems and NoCs. This algorithm is based on the use of a new algorithm for calculating the shortest path vectors with an estimate of the time complexity of $O(N)$ (in contrast to $O(N^2)$ for the well-known Dijkstra's algorithm). For the algorithm proposed, estimates for the number of reserve shortest paths and a method for their optimal use in case of dynamic changes in topology and overloads in the network for circulants of any dimension are obtained. In order to reduce the required memory in NoCs with a circulant topology when the pair routing algorithm is running, variants of the distribution in two-dimensional circulants of the minimum number of reference nodes containing mapping tables are proposed, and estimates of their number, the memory for mapping tables, and an increase in the average path length in graphs for the routing algorithm experimentally found.

The proposed algorithm can be used in the future as a basis for the development of routing algorithms in any subclasses of parametrically described regular graphs employed as models of communication networks, for example, for the graphs introduced in [14]–[19], which differ from each other in the degree of symmetry of the graph and may be of interest as possible NoC topologies.

REFERENCES

[1] K. Rocki, D. V. Essendelft, I. Sharapov, R. Schreiber, M. Morrison, V. Kibardin, A. Portnoy, J. F. Dietiker, M. Syamlal, and M. James, "Fast stencil-code computation on a wafer-scale processor," in *Proc. Int. Conf. High Perform. Comput., Netw., Storage Anal.*, Nov. 2020, pp. 1–14, doi: 10.1109/SC41405.2020.00062.

[2] S. K. Moore, "Cerebras' new monster AI chip adds 1.4 trillion transistors," *IEEE Spectr.*, Apr. 2021. Accessed: Nov. 1, 2021. [Online]. Available: https://spectrum.ieee.org/cerebras-giant-ai-chip-now-has-a-trillions-more-transistors

[3] D. Ditzel, R. Espasa, N. Aymerich, and A. Baum, "Accelerating ML recommendation with over a thousand RISC-V/tensor processors on esperanto's ET-SoC-1 chip," in *Proc. IEEE Hot Chips 33 Symp. (HCS)*, Aug. 2021, pp. 1–23, doi: 10.1109/HCS52781.2021.9566904.

[4] W. Dally and B. Towles, *Principles and Practices of Interconnection Networks*. Amsterdam, The Netherlands: Elsevier, 2003, p. 581.

[5] A. Yu. Romanov, "Development of routing algorithms in networks-on-chip based on ring circulant topologies," *Heliyon*, vol. 5, no. 4, Apr. 2019, Art. no. e01516, doi: 10.1016/j.heliyon.2019.e01516.

[6] M.-J. Tsai, H.-Y. Yang, and W.-Q. Huang, "Axis-based virtual coordinate assignment protocol and delivery-guaranteed routing protocol in wireless sensor networks," in *Proc. 26th IEEE Int. Conf. Comput. Commun.*, 2007, pp. 2234–2242, doi: 10.1109/INFCOM.2007.258.

[7] A. Benmessaoud Gabis and M. Koudil, "NoC routing protocols–objective-based classification," *J. Syst. Archit.*, vols. 66–67, pp. 14–32, May 2016, doi: 10.1016/j.sysarc.2016.04.011.

[8] E. A. Monakhova, A. Yu. Romanov, and E. V. Lezhnev, "Shortest path search algorithm in optimal two-dimensional circulant networks: Implementation for networks-on-chip," *IEEE Access*, vol. 8, pp. 215010–215019, 2020, doi: 10.1109/ACCESS.2020.3040323.

[9] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numer. Math.*, vol. 1, no. 1, pp. 269–271, Dec. 1959.

[10] I. Stojmenovic, "Multiplicative circulant networks topological properties and communication algorithms," *Discr. Appl. Math.*, vol. 77, pp. 281–305, Aug. 1997, doi: 10.1016/S0166-218X(96)00138-2.

[11] M. Karol, S. J. Golestani, and D. Lee, "Prevention of deadlocks and livelocks in lossless backpressured packet networks," *IEEE/ACM Trans. Netw.*, vol. 11, no. 6, pp. 923–934, Dec. 2003, doi: 10.1109/TNET.2003.820434.

[12] S. Das, C. Karfa, and S. Biswas, "Formal modeling of network-on-chip using CFSM and its application in detecting deadlock," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 28, no. 4, pp. 1016–1029, Apr. 2020, doi: 10.1109/TVLSI.2019.2959618.

[13] G. Tel, *Introduction to Distributed Algorithms*, 2nd ed. London, U.K.: Oxford Univ. Press, 2000, p. 596.

[14] O. G. Monakhov and E. A. Monakhova, "A class of parametric regular networks for multicomputer architectures," *Int. Sci. J. Comput. Syst.*, vol. 4, no. 2, pp. 85–93, 2000.

[15] J. C. Bermond, F. Comellas, and D. F. Hsu, "Distributed loop computer-networks: A survey," *J. Parallel Distrib. Comput.*, vol. 24, no. 1, pp. 2–10, Jan. 1995, doi: 10.1006/jpdc.1995.1002.

[16] S. L. Een-Chien, R. N. Farah, and M. Othman, "Some graph properties of the optimised degree six 3-modified chordal ring network," *Malaysian J. Fundam. Appl. Sci.*, vol. 12, no. 4, pp. 143–146, Feb. 2017, doi: 10.11113/mjfas.v12n4.518.

[17] D. Ledzinski, S. Bujnowski, T. Marciniak, J. M. Pedersen, and J. G. Lopez, "Network structures constructed on basis of chordal rings 4^{th} degree," in *Image Processing and Communications challenges* (Advances in Intellgent Systems and Computing), vol. 5, R. S. Choras, Ed. Heidelberg, Germany: Springer, pp. 281–299, 2014, doi: 10.1007/978-3-319-01622-1_33.

[18] J. M. Pedersen, T. M. Riaz, and O. B. Madsen, "Distances in generalized double rings and degree three chord rings," in *Proc. Int. Conf. Parallel Distrib. Comput. Netw.*, Vienna, Austria, 2005, pp. 153–158.

[19] J. M. Pedersen, T. M. Riaz, and O. B. Madsen, "Five novel selection policies for N2R network structures," in *Proc. 8th Int. Conf. Adv. Commun. Technol.*, Seoul, South Korea, vol. 3, Feb. 2006, pp. 2174–2179, doi: 10.1109/ICACT.2006.206430.

[20] F. K. Hwang, "A survey on multi-loop networks," *Theor. Comput. Sci.*, vol. 299, nos. 1–3, pp. 107–121, 2003, doi: 10.1016/S0304-3975(01)00341-3.

[21] E. A. Monakhova, "A survey on undirected circulant graphs," *Discrete Math., Algorithms Appl.*, vol. 4, no. 1, Mar. 2012, Art. no. 1250002, doi: 10.1142/S1793830912500024.

[22] E. A. Monakhova, "Algorithms and lower bounds for P-gossiping in circulant networks," in *Proc. Int. Symp. Parallel Archit., Algorithms Netw.*, Dec. 1997, pp. 132–137, doi: 10.1109/ISPAN.1997.645083.

[23] Y. Chen and C. Han, "Channel modeling and analysis for wireless networks-on-chip communications in the millimeter wave and terahertz bands," in *Proc. IEEE Conf. Comput. Commun. Workshops*, Apr. 2018, pp. 651–656, doi: 10.1109/INFCOMW.2018.8406954.

[24] D. DiTomaso, A. Kodi, D. Matolak, S. Kaya, S. Laha, and W. Rayess, "A-WiNoC: Adaptive wireless network-on-chip architecture for chip multiprocessors," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 12, pp. 3289–3302, Dec. 2015, doi: 10.1109/TPDS.2014.2383384.

[25] J.-Y. Cai, G. Havas, B. Mans, A. Nerurkar, J.-P. Seifert, and I. Shparlinski, "On routing in circulant graphs," in *Proc. Int. Conf.*, vol. 1627, 1999, pp. 360–369, doi: 10.1007/3-540-48686-0_36.

[26] D. Gómez, J. Gutierrez, and Á. Ibeas, "Optimal routing in double loop networks," *Theor. Comput. Sci.*, vol. 381, nos. 1–3, pp. 68–85, Aug. 2007, doi: 10.1016/j.tcs.2007.04.002.

[27] P. K. Jha, "A family of efficient six-regular circulants representable as a Kronecker product," *Discrete Appl. Math.*, vol. 203, pp. 72–84, Apr. 2016, doi: 10.1016/j.dam.2015.09.007.

[28] C. Martinez, R. Beivide, E. Stafford, M. Moreto, and E. M. Gabidulin, "Modeling toroidal networks with the Gaussian integers," *IEEE Trans. Comput.*, vol. 57, no. 8, pp. 1046–1056, Dec. 2008, doi: 10.1109/TC.2008.57.

[29] C. Martinez, E. Stafford, R. Beivide, and E. M. Gabidulin, "Modeling hexagonal constellations with Eisenstein-Jacobi graphs," *Problems Inf. Transmiss.*, vol. 44, no. 1, pp. 1–11, Mar. 2008, doi: 10.1134/S0032946008010018.

[30] E. A. Monakhova, "Optimal triple loop networks with given transmission delay: Topological design and routing," in *Proc. Int. Netw. Optim. Conf.*, Paris, France, 2003, pp. 410–415.

[31] A. Thomson and S. Zhou, "Gossiping and routing in undirected triple-loop networks," *Networks*, vol. 55, no. 4, pp. 341–349, Jul. 2010, doi: 10.1002/net.20327.

[32] C. Camarero, C. Martínez, and R. Beivide, "Symmetric interconnection networks from cubic crystal lattices," 2013, *arXiv:1311.2019*.

[33] T. Dobravec, J. Zerovnik, and B. Robic, "An optimal message routing algorithm for circulant networks," *J. Syst. Archit.*, vol. 52, no. 5, pp. 298–306, 2006, doi: 10.1016/j.sysarc.2005.12.003.

[34] B. Robic, "Optimal routing in 2-jump circulant networks," Univ. Cambridge, Cambridge, U.K., Tech. Rep. N397, 1996, p. 7.

[35] J. Zerovnik, B. Robic, and T. Dobravec, "Optimal permutation routing in 2-jump circulant networks," in *Proc. 1st Int. Conf. Softw. Eng. Appl. Netw. Parallel/Distrib. Comput. (SNPD)*, 2000, pp. 175–180.

[36] B.-X. Chen, J.-X. Meng, and W.-J. Xiao, "A constant time optimal routing algorithm for undirected double-loop networks," in *Proc. 1st Int. Conf. Mobile Ad-Hoc Sensor Netw.*, vol. 16, 2005, p. 308316, doi: 10.1007/11599463_31.

[37] R. Lewis, "Distance partitions of extremal and largest known circulant graphs of degree 2 to 9," 2014, *arXiv:1408.0988*.

OLEG G. MONAKHOV received the M.S. degree in system analysis from Novosibirsk Electrotechnical Institute, Novosibirsk, Russia, in 1974, and the Ph.D. degree in computer science from Taganrog State Radio Technical University, Taganrog, Russia, in 1987. From 2000 to 2002, he was a Visiting Professor with the University of Aizu, Aizu-Wakamatsu, Japan. He is currently a Leading Scientific Researcher with the System Modeling and Optimization Laboratory, Institute of Computational Mathematics and Mathematical Geophysics of SB RAS, Novosibirsk. He is the author of two books and 120 articles. His research interests include parallel computer architecture, network topologies, routing algorithms, evolutionary computation, genetic programming, and metaheuristic algorithms.

EMILIA A. MONAKHOVA received the M.S. degree in mathematics and applied mathematics from Novosibirsk State University, Novosibirsk, Russia, and the Ph.D. degree in computer science from Leningrad Electrotechnical Institute, Leningrad, Russia, in 1988. She is currently a Senior Scientific Researcher with the System Modeling and Optimization Laboratory, Institute of Computational Mathematics and Mathematical Geophysics of SB RAS, Novosibirsk. She is the author of two books and more than 100 articles. Her research interests include parallel computer architecture, network topologies, routing algorithms, and evolutionary computation.

ALEKSANDR YU. ROMANOV (Member, IEEE) received the B.S. and M.S. degrees in computer systems and networks from the National Technical University Kharkov Polytechnic Institute (NTU KhPI), Kharkov, Ukraine, in 2007 and 2009, respectively, and the Ph.D. degree in technical sciences from the Federal State-Funded Institution of Science Institute for Design Problems in Microelectronics of Russian Academy of Sciences (IPPM RAS), Zelenograd, Russia, in 2015. He is currently an Associate Professor with the National Research University Higher School of Economics (HSE University), Moscow; the HSE Tikhonov Moscow Institute of Electronics and Mathematics. He is also a University Teacher of the following subjects: SoC development and system design of electronic devices. He is the author of the book *Digital Design: Practical Course* and more than 100 scientific articles. His research interests include SoC, NoC, embedded systems, FPGA, and neural networks. He is an IEEE Industrial Electronics Society Member. He has received numerous awards as the supervisor of students' teams taking part in international competitions in programming, electronics and robotics, as well as the HSE University Best Lecturer Award, in 2018, 2019, and 2021.

ANDREJ M. SUKHOV (Member, IEEE) received the Ph.D. degree in physics and mathematics at Moscow, Russia, in 1993, and the D.Sc. degree in computer networking from the Moscow State University of Electronics and Mathematics (MIEM, HSE University, Moscow), in 2007. He is currently a Professor at HSE University. Over the past 25 years, he has been involved in acting as an investigator for more than 15 telecommunication projects supported by the Russian Government, RFBR, INTAS, NATO, ESA, etc. Five of his postgraduate students received the Ph.D. degree. He is the author of two monographs and over 100 research papers. His research interests include computer networks and network security. In 2020, he was elected as a Senior Member of ACM. He has been a member of IEEE Communications Society for over ten years.

EVGENII V. LEZHNEV (Member, IEEE) received the B.S. and M.S. degrees in computer networks and systems from the National Research University Higher School of Economics (HSE University), Moscow, Russia, in 2015 and 2017, respectively. Since 2018, he has been working as a Research Trainee with the Laboratory of Computer-Aided Design Systems. Since 2020, he has been an Assistant Teacher at HSE Tikhonov Moscow Institute of Electronics and Mathematics. Currently, he is a Graduate Student majoring in design automation systems at HSE University. He is the coauthor of 12 articles. His research interests include low-level modeling and development of NoC topologies. His awards and honors include the Golden HSE Award, in 2019, and the Silver Nestling Nomination and Innovate FPGA (Intel) Award.