Analysis of Approaches for Synthesis of Networks-on-chip by Using Circulant Topologies

IOP Conf. Series: Journal of Physics: Conf. Series **1050** (2018) 012071 doi:10.1088/1742-6596/1050/1/012071

Analysis of Approaches for Synthesis of Networks-on-chip by Using Circulant Topologies

A Yu Romanov[1,2,3], A A Amerikanov[1] and E V Lezhnev[1]

[1]National Research University Higher School of Economics, 20 Myasnitskaya Ulitsa, 101000, Moscow, Russia

[2] e-mail: a.romanov@hse.ru
[3]Author to whom any correspondence should be addressed

Abstract. The article gives a review of existing methods of network-on-chip design based on the approach in which mapping of the characteristic tasks graph is performed on a given regular topology. The networks-on-chip synthesis problem is generally characterized. The analysis and comparison of standard topologies (mesh and torus) with circulant topologies are performed. Advantages and disadvantages of mesh and torus topologies usage, and the effect, achieved by their application to various implementations of networks on chip, are analyzed. Extension of the scope of solutions for standard regular network topologies mesh and torus on the circulant topologies with better characteristics is proposed. This will make it possible to take advantage of the deterministic approach, but with the use of more effective NoC topologies optimized for a particular task.

1. Introduction

The architectural solutions within networks-on-chip (NoCs) become more and more widespread in modern embedded systems. Such systems are generally based on hierarchical principle containing a large number of processors and peripherals. There is a trend of constantly increase of the number of processor cores, as uniprocessor systems cannot cope with applications requiring high computing performance and large data flows (exchange processing of video and still images in IP-telephony, etc.) [1]. In this way, the development of effective communication subsystems is necessary. Under such conditions, implementation of new approaches for NoCs becomes of great importance.

2. NoC synthesis problem

Due to a complex structure of NoCs, their synthesis problem is difficult to formulate and has a lot of ways to be solved. The initial data for NoC synthesis problem is a characteristic task graph. Its vertices correspond to the IPs (intellectual properties – separate modules, usually computational nodes), and are characterized by injection of packages in NoC environment. Mathematically, it can be described as follows: $G=G(C,A)$, a directed graph, where C – set of vertices (IPs), and A – set of communication processes between them. In turn, NoC system architecture is characterized by the followings: $T(R,Ch)$ topology (R and Ch – sets of routers and physical links between them); a routing mechanism (P_R); a mapping function of application characterization graph vertices onto NoC routers ($\Omega(C)$).

The characteristic task graph's directed edges correspond to data transmission process between two IPs. They can be characterized by minimal and maximal communication delay, communication intensity and bandwidth [2].
According to the above definitions, the model can be complemented by communication energy cost dependence formula:

$$E = \sum_{\forall a_{i,j}} \upsilon(a_{i,j}) \times E_{bit}(\Omega(c_i), \Omega(c_j)) \quad (1)$$

where $\upsilon(a_{i,j})$ – capacity of data transmission process between nodes *i, j*; $E_{bit}(\Omega(c_i), \Omega(c_j))$ – energy spent on 1 bit of data transmission between nodes c_i and c_j.

And the communication energy minimization problem can be expressed by a formula (2). It consists in finding such Ω (*C*) which arranges for connections of communication process with high capacity to have a low energy consumption to transfer 1 bit of data [3].
Similarly, the formula of the total volume of data, transmitted between nodes in a NoC, is as following:

$$V = \sum_{\forall a_{i,j}} \upsilon(a_{i,j}) \times L_{i,j}(\mathrm{P}_R(r, i, j)) \quad (2)$$

where $L_{i,j}(\mathrm{P}_R(r, i, j)$ – distance between nodes *i* and *j* according to the routing algorithm.

According to the above formulas, the application problem can be represented as a set of an individual subtasks graphs whose edges describe the data flow between them. In this case, the characteristic task graph is obtained by mapping of one or several subtasks on the IPs and by data flow distribution between subtasks on the edges of the graph.
In this case, the synthesis problem is to choose a NoC topology and projection of a characteristic task graph onto it in accordance with the demands for the cost of the bandwidth, power, chip area, as well as minimal and maximal delay of the packages transmission. Under this approach, the topology is represented as a numbered graph whose vertices correspond to the routers (every router is connected to the corresponding IP) and edges – to the physical data channels (trunks). In addition to the topology, a synchronization and communication subsystem method, a transmission of data streams technology, a quality of service insurance method, a routing strategy, a structure of routers, and so on are chosen.
Thus, according to [4], the synthesis of NoC design requires the solution of four distinct tasks:

- Selection of an appropriate NoC topology.
- Task graph mapping onto the topology.
- Efficient data streams routing insurance.
- Decision on NoC functioning technology.

The second [5, 6], third [7], and fourth [8] tasks are discussed in the scientific literature, but they aren't the object of this research. As for the problem of the efficient NoC topology synthesis, it is chosen either an application-specific topology, or platform dependent topology (in case when the information about the future computing tasks which will be performed by the finite system is known, or topology is defined by hardware), or a regular topology is used. The approaches, based on application-specific topologies, are observed in works [9, 10], and they aren't a subject of this research either. The approach with regular topologies implementation is analyzed below.

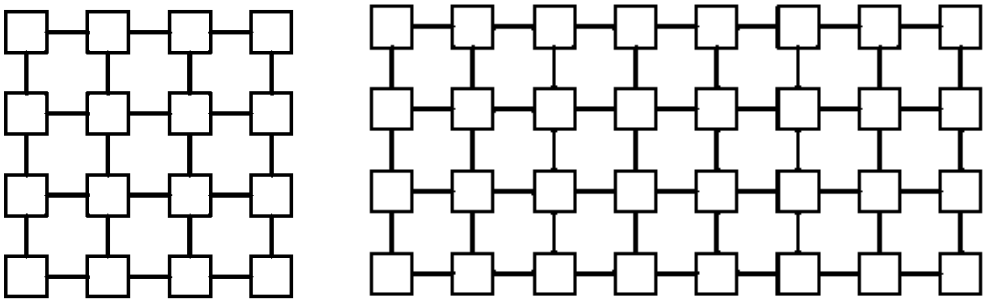
2.1. The common regular topologies: mesh and torus

The definition of a regular topology in various sources is different. Thus, in [11] there is a definition according to which topologies with the same degree of vertices are regular – that is, those having homogeneous routers. According to this definition, torus and hypercube topologies, unlike mesh, are regular. But in practice, all topologies with a scalable homogeneous structure (we will also use this

definition) belong to regular topologies, therefore, mesh topology can also be referred to regular topologies [3, 8, 12, 13]. Most topologies can be multidimensional [14–16], but for their implementation, it is necessary that NoC components can be located on several physical layers of the crystal; therefore, 2D flat topologies are most often used in NoC design.

The main characteristics of the topology are as follows: number of vertices – routers (*N*); number of edges – physical connections between the routers (*Ed*); order of vertex – the number of edges emanating from it (*St*); diameter of the graph – the maximum among minimum distance between any two nodes (*D*); average distance between the shortest paths among all nodes of the graph (L_{av}).

The most common among regular topologies is mesh topology which is a network of $N = m \cdot n$ nodes of rectangular type, or a network of $N = n \cdot n$ nodes of square type. Each node is connected with the four neighboring ones (Figure 1). The peripheral nodes have some unused ports [8, 17].



(a) (b)

Figure 1. Mesh topology for 16 (a), and 32 (b) nodes.

Mesh $n \times n$ (Figure 1 (a)) networking characteristics are [8, 17]:

$$Ed = 2 \cdot (N - \sqrt{N}),\ \ St = 2 \div 4,\ \ D = 2 \cdot (\sqrt{N} - 1),\ \ L_{av} = \frac{2(N-1)}{3\sqrt{N}},\ \ Ext = 2\sqrt{N} + 1. \qquad (3)$$

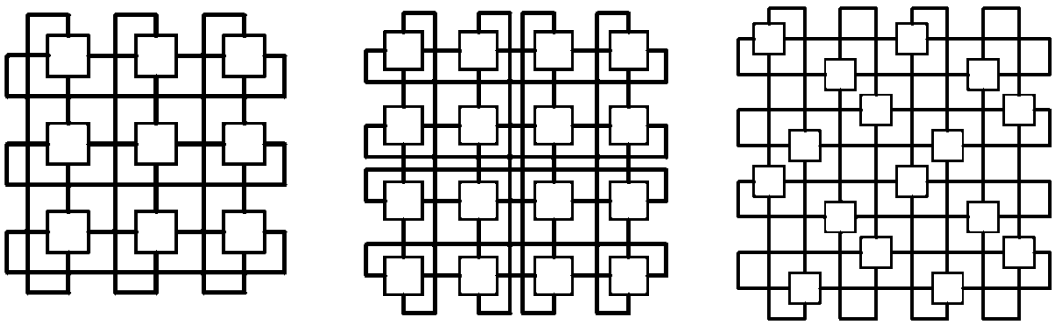
Characteristics of a rectangular mesh $m \times n$ (Figure 1 (b)) are:

$$Ed = 2mn - m - n,\ \ St = 2 \div 4,\ \ D = m + n - 2,\ \ L_{av} = \frac{(m+n)(mn-1)}{3mn},\ \ Ext = \min(m,n), \qquad (4)$$

where $\mathrm{mod}(x, y) = x - y\left\lfloor \frac{x}{y} \right\rfloor$.

The main disadvantage of mesh topology is a too large diameter. So, for a network with 3x3 nodes, the distance between routers in the opposite corners is of 4 hops. Moreover, in such networks, the load on the routers is distributed unevenly.

The attempt to eliminate these disadvantages at the expense of the large resource costs is the topology of torus obtained by the combining of extreme nodes of mesh with its opposite ones (Figure 2 (a, b)). In this case, $D = 2 \cdot \lfloor \sqrt{N}/2 \rfloor$, but $Ed = 2 \cdot N$, and *St*=4. A kind of a modification is the folded torus which is to reduce the length of the connecting lines between the extreme nodes (Figure 1 (c)) [8, 11].



(a) (b) (c)

Figure 2. Torus topology for 9 (a) and 16 (b) nodes and folded torus for 16 (c) nodes.

Characteristics of a square torus $n \times n$: $Ed = 2N$, $St = 4$, $D = 2\left\lceil \sqrt{N}/2 \right\rceil$, $L_{av} = \frac{N\sqrt{N}(\text{mod}((\sqrt{N}+1),2))}{2(N-1)} + \frac{\sqrt{N}(\text{mod}(\sqrt{N},2))}{2}$, $Ext = 2\sqrt{N}+1$. For a torus topology of 3x3 in size, the number of connections is already 18, and the diameter is 2 hops, that is, an increase in the number of connections by 50% provides a reduction in diameter by a factor of 2. In addition, the degree of vertices in this topology is fixed which makes it possible to use homogeneous routers. The disadvantage of this topology is the large length of the end trunks, so, its modification – folded torus (Figure 2 (c)), which allows reducing the length of connecting lines between the end nodes [18], is often used.

All mentioned topologies reside in the restriction in the number of nodes to which they can be applied. For example, mesh and torus topologies are applicable for networks of the square form with the number of the nodes equal to the natural numbers to the power of 2 (4, 9, 16, 25, ...), but to create a network of 11 routers is not possible at all. The use of a rectangular topology (Figure 1 (b)) is possible, but this leads to an increase in the diameter of the graph and a decrease in the width of the division of the graph in half. And for the number of nodes, which is a prime number (5, 7, 11, ...), the task becomes even more complicated, since one must abandon either deterministic routing algorithms (for example, XY [19]), or take a topology for more nodes, leaving one router as an auxiliary and unconnected to the computing node. Thus, there is a need for the new, more flexible approaches able to offer the efficient topologies for the networks with the arbitrary number of nodes and the configured number of connecting lines.

2.2. Circulant networks for NoC development

There are a number of topologies, such as: chord rings, octagon, spidergon, etc. which are variations of circulant networks [20].

Let's define the circulant network. Circular networks (graphs) are Cayley graphs of Abelian groups. Let N – number of vertices of the graph, and $S = \{1 \le s_1 < \ldots < s_k < N\}$ – set of k integers (generatrices). Then, an undirected graph $C(N; s_1, \ldots, s_k)$ with a set of vertices $V = \{0, 1, \ldots, N-1\}$ and a set of edges $E = \{(v, v \pm mod(s_i, N)) \mid v \in V, i = \overline{1,k}\}$ is a circulant of dimension k.

Thus, the circulating network $C(N; s_1, \ldots, s_k)$ can be represented as a ring structure, where each vertex is associated with k successive vertices and k previous vertices in steps of $s_1, \ldots, s_k$ (Figure 3).

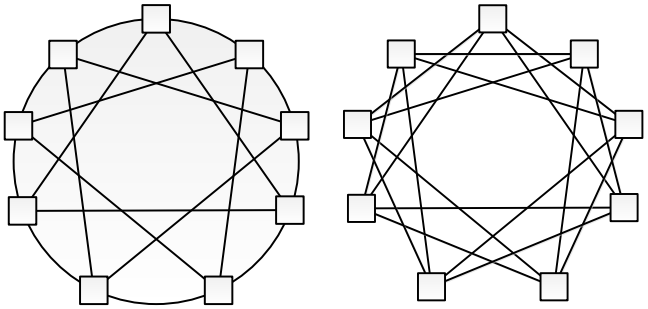


Figure 3. Circulant topology: (a) – $C(9;1,3)$; (b) – $C(9;2,3)$.

For implementation as NoC topologies, circulants of dimension 2 are of primary interest. This class of circulants has a degree of vertices 4 and good topological characteristics which makes it promising for use as a basis for networks-on-chip. Since relatively small routers with 4 inputs / outputs are suitable for their construction, plenty of examples were developed (for example, Netmaker library [4]), and their high efficiency and balance were shown.

The theory of second-order circulants is well developed, and it was shown in [20, 21] that circulants with generatrices, calculated by formula

$$C\left(n;\ D-1,\ D\right),\quad \text{where } D=\sqrt{n/2},\ n>2, \tag{5}$$

are optimal ones.

Thus, by synthesizing them, we were able to compare their characteristics with the characteristics of the mesh and torus graphs. Since the diameter and average distance between the nodes are one of the most important characteristics of NoC, we give the obtained characteristics for topologies with number of vertices from $3^2 = 9$ to $23^2 = 529$ (we take topologies only for the number of vertices which are numbers in the second degree; for mesh and torus topologies, the square form is the most optimal). Dependencies of L_{av} and D on the number of vertices are shown in the figures below.

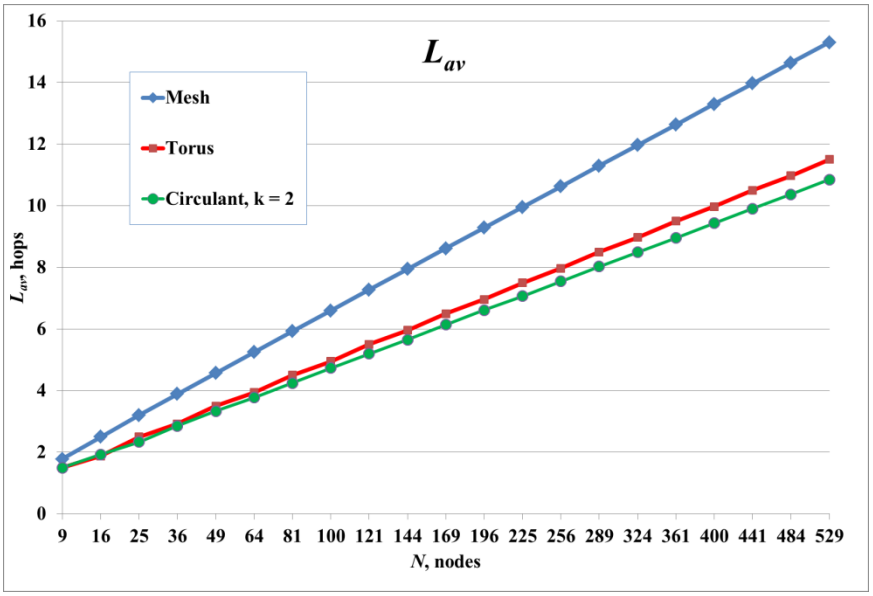


Figure 4. Dependence of average distance between the shortest paths among all the nodes on the number of nodes.

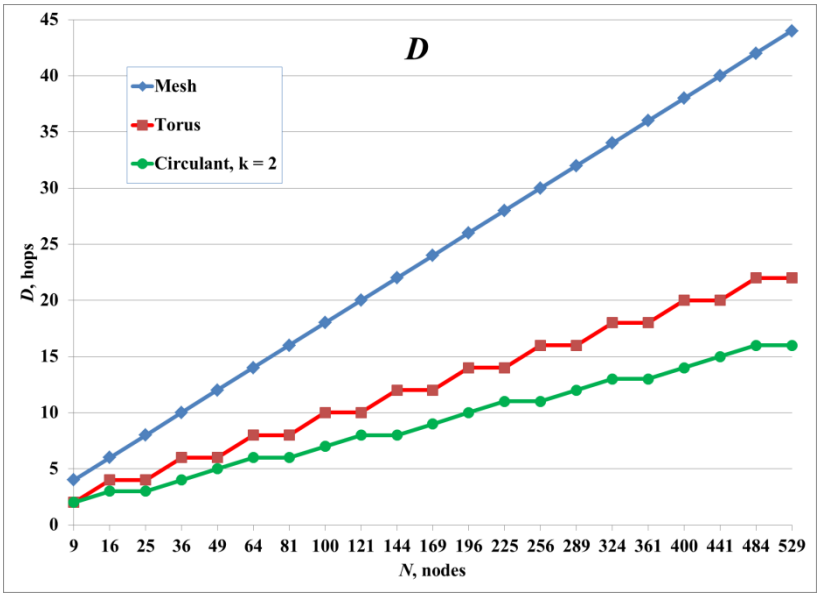


Figure 5. Dependence of the diameter on the number of nodes.

Based on the obtained graphs, it can be concluded that circulants of form $C(n; D-1, D)$, at equal costs of connective resources as torus topology, make it possible to reduce: the diameter to 20.0–59.4% in comparison with torus, and to 50.0–63.9% in comparison with mesh; the average distance – to 2.1–6.7% in comparison with torus, and to 15.6–29.2% – in comparison with mesh. This is generally obvious, since torus can be represented as a circulant-like graph, but not the optimal one. In addition, in comparison with mesh and torus topologies, it is possible to construct circulant graphs for any number of vertices without reducing their effectiveness. Thus, the use of circulant graphs as a topological basis in NoC development makes it possible to improve NoC characteristics in comparison with mesh and torus topologies without losing the regularity of networks and increasing the dimension of routers. Moreover, like in other regular topologies, circulants interconnects can be reconfigured according to the application task graph by adding and removing some links, as proposed in work [22].

3. The review of methods of NoC synthesis by projecting the characteristic task graph onto a regular topology

A widely used approach is when one of the regular topologies is selected at the beginning of the synthesis and remains unchanged in the course of the further mapping of the characteristic tasks graph onto it. The advantage of this approach is that such well-known regular topologies as torus, mesh, hypercube, star, etc. are sufficiently investigated. They have predictable characteristics and deterministic routing algorithms [12]. In addition, there is a wide experience of their use, as well as circulants [23], in global networks which can be used in NoC design, taking into account the specificity of ASIC and FPGA hardware platforms, of course.

An example of such an approach is the algorithm for the energy-efficient construction of a NoC based on mesh topology, proposed in [3]. The characteristic tasks graph is mapped onto a mesh-like topology consisting of homogeneous cells (a computational core and a router) and using the tree search algorithm for node mapping alternatives: at each step of descending through branches of the alternative tree, the consumed power of the mapped nodes is calculated, and less optimal alternatives are discarded. According to the authors, this makes it possible to reduce the power consumption of the NoC by approximately 60%, in comparison with the arbitrary mapping of the problem, and to reduce the search time by 27 times, in comparison with the full-search algorithm. In [24] the same authors use linear programming in optimizing delays in packet transmission by eliminating data collisions in the transmission path and reducing the weighted communication distance when mapping the characteristic

tasks graph onto a mesh topology. This approach gives an increase in capacity by 17%, compared to the previous solution, but leads to an increase in energy costs by 9%. In the next paper [13], the authors attempted to compensate for the shortcomings of mesh topology by realizing the possibility of adding regions that occupy the dimensions of several cells, as well as pre-fixing the placement of certain nodes (for example, those related to the inputs / outputs of the chip). The represented algorithms are of a rather universal nature and can be adapted relatively easily for use with circulant topologies; moreover, a number of problems, associated with uneven load of nodes located on the periphery of the mesh graph, are eliminated.

The method of linear programming, used in [25], is to search for the optimal mapping of the characteristic tasks graph onto a mesh topology with a reduction in the average distance between nodes and a decrease in the maximum load of communication lines. Mapping is performed in two stages: nodes are mapped at the first stage, and connections between them – at the second stage. This allowed accelerating the solution of the problem of linear programming $10 \sim 10^3$ times. The result of simulation of various video applications was a decrease in power consumption and bandwidth requirements of NoCs by 55.5% and 49.2%, respectively, compared to heuristic algorithms. Despite the fact that the results of the proposed method are demonstrated using the mesh topology example, we did not find any obstacles to its extension to circulant topologies.

In [26] NMAP algorithm is proposed for optimal mapping of the characteristic tasks graph onto the mesh topology, taking into account the bandwidth requirements. According to the algorithm, at first, the most connected node is mapped to the center of the network on the router having the largest number of neighbors, and after that, depending on the degree of connectivity with nodes already located on the free routers, other nodes are sequentially mapped in order to minimize the cost of communication between them. The obtained result is improved by pairwise permutations of the nodes in order to achieve a minimum average distance between the nodes. Data flows, to which capacity of the communication lines is not sufficient, are split into several threads, and the routing table is given various alternative shortest paths for them. This algorithm was used in synthesis of NoCs for video applications using XPipes library [27] and, according to the authors, this allowed reducing communication costs by 32% and increasing the throughput by 53%. This approach was developed in [28] by planning the optimal placement of components on a chip by linear programming and introducing the possibility of selecting the characteristic tasks graph for various regular topologies from XPipes library. And in [29], specialized software SUNMAP was proposed to map the characteristic tasks graph on a given topology included in NoC NetChip generation environment [30]. The proposed algorithm can be applied to NoC on the basis of circulant topologies, taking into account the fact that the process of mapping the most loaded node of the tasks characteristic graph can be started from any node of the circulant topology, in view of its peculiarity that it is completely symmetric. This will simplify and speed up the algorithm under consideration.

The MOCA algorithm [31], according to the authors, ensures close results, in comparison with the previous approach, but in less time. This algorithm is used to divide the task graph into two equal parts in the form of an artificial tree which are then easily mapped to a mesh topology. In addition, optimization of the capacity of trunks and data transmission delays is performed. According to the authors of the algorithm, this led to a decrease in energy consumption by 14%, compared to the mapping, obtained by linear programming for processing tasks of video and audio data. This algorithm can also be easily applied to circulant topologies.

In work [32], based on Aethereal NoC, it is proposed a solution that can be applied to arbitrary NoC topology, and distribution of data flows with different quality of service is considered. In this case, the UMARS+ algorithm is used, according to which, when choosing the SCC topology, data streams with a guaranteed minimum delay service are mapped on it. Incorrect turns [33] are detected to avoid deadlocks and, in view of this, flows with guaranteed bandwidth service are mapped when routing tables are created, as well as assigning time slots to the ports of the routers. Using the MPEG decoder as an example, it is shown that the use of information about data flows with various services and their spatial time distribution, when mapping the task graph to the NoC topology, gives a 33% gain in

resources and 35% – in energy consumption, and also reduces by four times the maximum delay of passing packets, in comparison with the approach without taking into account information about the quality of service. This approach does not impose any restrictions on the topology.

In [34] it was suggested to generate the variants of mapping the characteristic tasks graph onto a mesh-shaped topology with the help of genetic algorithms. The behavioral model of the obtained mappings is then subjected to discrete-event modeling to estimate the gain in the performance and power consumption of the NoCs and to compare them with the results of other mappings to select the best ones. A two-stage genetic algorithm for mapping the graph of problems onto NoC mesh topology is used in [35]. The algorithm takes into account that the compute nodes are inhomogeneous and are divided into groups with different performance. Each possible mapping of the tasks graph is expressed in the form of a chromosome, where the genes correspond to the vertices of the graph, and their values – to NoC nodes. At the first step of the algorithm, a simplified formula is used to calculate the fitness function of chromosomes, according to which the transmission delay between the vertices of the tasks graph depends directly on the average minimum distance between nodes for mesh topology. The best individuals of the population undergo mutations and rearrangements. The most optimal instances become the initial data for the second step of the genetic algorithm which differs from the first one by a more precise formula for calculating the data transmission delay and depends directly on the minimum distance between nodes and specifies the placement of the vertices of the task graph on the nodes of the corresponding type. Taking into account the fact that, unlike mesh, vertices in the circulants are completely homogeneous, this approach with some improvements can be extended to circulant topologies.

A similar representation of chromosomes is used in [36], where the optimal parameter of the specimen is the operating temperature of the crystal which is evaluated using the HotSpot utility [37]. The energy consumption values of each compute node of the two-dimensional mesh network, required for the HotSpot utility, are estimated using Synopsys Power Compiler synthesis. To generate test sequences and to simulate this approach, exemplified by development of a code decoder with a low density of parity checks, NoCSim simulator is used [38]. The development of this approach is in [16], where the projection is performed on 3D mesh, and also in [39], where, in addition to the optimality parameter for energy consumption, the parameter of minimization of the occupied area of the crystal was added. Attention is also worthy of the work [40], where the questions of energy-efficient mapping of tasks on the already established architecture of the location of NoC nodes are considered. This approach is based on the analysis of temperature distribution on the crystal at the stage after synthesis, therefore, the logical topology, which is the basis of NoC communication system, does not play any role here.

In work [41], it is proposed a congestion-aware applications mapping heuristic algorithm, based on betweenness centrality metric of links, which makes it possible to alleviate congestion from highly loaded NoC links. The proposed algorithm utilizes the volume of communication traversing through NoC links to obtain the operational and dynamic characteristics of the system. Compared with other baseline application mapping algorithms, the proposed approach achieved up to 46% and 12% lower channel load and end-to-end latency. Despite the fact that the work of the algorithm is demonstrated on the example of mesh topology, this approach can be extended to any regular topology including circulant graphs.

In [42] the process of synthesis of an energy-efficient NoC consists of such stages as: mapping the graph of subtasks to the characteristic tasks graph; mapping the processor elements to cells in a 2D mesh-like topology; mapping routing information to connectivity resources; assigning priorities between tasks and setting the operating frequency lines of communication depending on their loading according to the DVS algorithm [43]. The first three steps are performed using genetic algorithms. This approach makes it possible to reduce energy consumption by 39%, in comparison with the algorithm which is based on the use of a random mapping of the task on NoC and XY routing. This approach fully adapts to our task by performing the processor elements mapping on the circulant

topology on the second stage instead of mesh and selecting the routing algorithm suitable for circulant topology.

In the latest researches, we observed a two-step approach, when several methods of the optimal mapping of the application task graph onto a regular network topology of the NoC are combined together, and it mutually reinforces the resulting performance gain.

For example, in work [44], it is proposed a dynamic resource balance algorithm to achieve a higher system performance by balancing the utilization of on-chip computing resources and communication resources: at first, a dynamic communication optimization algorithm is applied, and then a multi-rectangle selection algorithm, to choose a corresponding number of resource regions for the constructed mapping scheme to allocate the application, is used. The authors state that their approach improves the system throughput by at most up to 31.6% compared with First Free algorithm [45]; up to 25.2% – compared with Nearest Neighbor algorithm [46]; and up to 9.4% – compared with CoNA-SHiC algorithm [47]. The work contains a rather simplified model of the location of compute nodes and their corresponding routers on the chip area; in reality, they can intersect and overlap, as well as do the communication wires between them. The usage of circulant graphs as topologies for NoCs will require a more accurate calculation of the Manhattan distances between routers which can be obtained after the NoC synthesis stage and its placement on the crystal resources, especially, when it is an FPGA. Nevertheless, this task does not seem impossible.

In other work [48], the problem of energy-efficient contention-aware application mapping and scheduling on NoC is considered. It is presented a model, where voltage scaling techniques for processors are combined with frequency tuning techniques for NoC links to save overall system energy consumption. The two-step approach consists of: 1) application mapping problem formulated as a quadratic binary programming problem to minimize the communication energy and solved by using a relaxation-based iterative rounding algorithm; 2) application scheduling problem which aims to find the optimal voltage level for each application task and optimal frequency level for each communication link to minimize the overall system energy consumption and solved by genetic algorithm to search the solution space for the voltage and frequency assignment that minimizes the overall system energy consumption and meets the application's deadline. As a result, the authors argue that energy saving achieves up to 26.5% in simulations with real application graphs. In this study, the NoC structure and topology are presented at a high level of abstraction with a logical spatial connection of heterogeneous computing nodes to mesh topology, but there are no obstructions to prevent it from being modified to a more optimal circulant topology.

4. Conclusion

Thus, the analysis of various ways of NoC synthesis by projecting the characteristic tasks graph onto a predefined regular topology has shown that all optimization methods consist in finding the arrangement of computational tasks on topology nodes capable to minimize data flows and distances between nodes with the most intensive data exchange. This is achieved through the application of various approaches based on the solution of the problem of linear programming, the use of genetic algorithms and various tools for estimating the distribution of energy consumption, and other performance NoC characteristics.

In most cases, mesh topology is used as the predefined NoC topology, characterized by the simplicity of the organization and use of deterministic routing algorithms, which is its advantage, but, at the same time, determines the drawbacks – the limitations associated with the non-optimal form of topology, especially, when mesh topology of rectangular shape is used. For this reason, the approach to NoC synthesis by projecting the characteristic tasks graph onto a predefined regular topology is mainly used when NoC has a homogeneous structure with homogeneous nodes and a uniform distribution of data flows. Although, even in this case, it is actual to use circulants as NoC topological solutions with better characteristics. In addition, most of the approaches to NoC synthesis are considered to be quite universal and can be applied for NoC synthesis on the basis of circulant topologies which will make it

possible to take advantage of the deterministic approach, but with the use of more effective NoC topologies optimized for a particular task.

Acknowledgments
This work is an output of a research project implemented as part of the Basic Research Program at the National Research University Higher School of Economics (HSE).

References

[1] Meenderinck C, Azevedo A, Juurlink B, Messa M A and Ramires A 2009 Parallel Scalability of Video Decoders *Journal of Signal Processing Systems* **57** pp 173–194
[2] Marculescu R and Ogras U 2009 Outstanding Research Problems in NoC Design: System, Microarchitecture, and Circuit Perspectives *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems* **28** pp 3–21
[3] Hu J and Marculescu R 2003 Energy-aware mapping for tile-based NOC architectures under performance constraints *Proceedings of Asia South Pacific Design Automatic Conference*. pp 233–239
[4] Romanov O and Lysenko O 2012 The Comparative Analysis of the Efficiency of Regular and Pseudo-optimal Topologies of Networks-on-Chip Based on Netmaker *Proceedings of Mediterranean Conference on Embedded Computing* pp 13–16
[5] Naresh K, Vasantha M and Nithin K 2017 System level fault-tolerance core mapping and FPGA-based verification of NoC *Microelectronics Journal* **70** pp 16–26
[6] Marcon C, Webber T and Susin A 2017 Models of computation for NoC mapping: Timing and energy saving awareness *Microelectronics Journal* **60** pp 129–143
[7] Gabis A and Koudil M 2016 NoC routing protocols – objective-based classification *Journal of Systems Architecture* **66–67** pp 14–32
[8] Dally W J and Towles B 2004 *Principles and practices of interconnection networks* (San Francisco, CA: Elseiver) p 550
[9] Huang J, Zhong W, Li Z and Chen S 2017 Lagrangian relaxation-based routing path allocation for application-specific network-on-chips Integration, *the VLSI Journal*. In press.
[10] Cilardo A and Fusella E 2016 Design automation for application-specific on-chip interconnects: A survey *Integration, the VLSI Journal* **52** pp 102–121
[11] Saldana, M, Shannon L and Chow P 2006 The Routability of Multiprocessor Network Topologies in FPGAs *Proceedings of the 2006 international workshop on System-level interconnect prediction (SLIP'06)* (NY: ACM) pp 49–56
[12] Benini L and Micheli G 2006 *Networks on Chips: Technology and Tools* (San Francisco, CA: Morgan Kaufmann) p 408
[13] Hu J and Marculescu R 2005 Energy- and Performance-Aware Mapping for Regular NoC Architectures *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*. **24** 4 pp 551–562
[14] Joseph J M, Blochwitz C, García-Ortiz A and Pionteck T 2017 Area and power savings via asymmetric organization of buffers in 3D-NoCs for heterogeneous 3D-SoCs *Microprocessors and Microsystems* **48** pp 36–47
[15] Pavlidis V F and Friedman E G 2007 3-D Topologies for Networks-on-Chip *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* **15** 10 pp 1081–1090
[16] Addo-Quaye C 2005 Thermal-aware Mapping and Placement for 3-D NoC Designs *IEEE International SOC Conference* pp 25–28
[17] Suboh S, Bachouya M, Gabe J and El-Ghazawi T 2008 An interconnection architecture for network-on-chip systems *Telecommunication Systems* **37** 1–3 pp 137–144
[18] Asaduzzaman A, Chaturvedula S R and Pendse R 2013 A novel folded-torus based network architecture for power-aware multicore systems *Computers & Electrical Engineering* **39** 8 pp 2494–2506

[19] Chawade S D, Gaikwad M A and Patrikar R M 2012 Review of XY Routing Algorithm for Network-on-Chip Architecture *International Journal of Computer Applications* **43** 21 pp 0975–8887
[20] Monahova E A 2011 Strukturnye i kommunikativnye svojstva cirkuljantnyh setej *Prikladnaja diskretnaja matematika* **3**(13) pp 92–115
[21] Beivide R, Herrada E and Balcazar J L 1991 Optimal distance networks of low degree for parallel computers *IEEE Trans. Computers*. **40** 10 pp 1109–1124
[22] Ajwani D, Hackett A, Ali S, Morrison J P and Kirkland S 2016 Co-optimizing application partitioning and network topology for a reconfigurable interconnect *Journal of Parallel and Distributed Computing* **96** pp 12–26
[23] Wang D 2015 Tradeoff study among traffic egression schemes and member allocation optimization for link aggregation groups in integrated switching systems *Computer Networks.* **77** pp 56–72
[24] Chou C and Marculescu R 2008 Contention-aware Application Mapping for Network-on-Chip Communication Architectures *IEEE International Conference on Computer Design (ICCD 2008)* pp 164–169
[25] Rhee C, Jeong H Y and Ha S 2004 Many-to-many Core-switch Mapping in 2-D Mesh NoC Architectures *IEEE International Conference on Computer Design: VLSI in Computers and Processors (ICCD 2004)* pp 438–443
[26] Murali S and De Micheli G 2004 Bandwidth-constrained Mapping of Cores onto NoC Architectures // *Proceedings of the Conference on Design, Automation and Test in Europe (DATE'04)* **2** pp 16–20
[27] Bertozzi D and Benini L 2004 Xpipes: A Network-on-chip Architecture for Gigascale Systems-on-chip *IEEE Circuits and Systems Magazine* **4** 2 pp 18–31
[28] Murali S, Benini L and De Micheli G 2005 Mapping and Physical Planning of Networks-on-chip Architectures with Quality-of-service Guarantees *Proceedings of the Asia and South Pacific Design Automation Conference (ASP-DAC 2005)* **1** pp 27–32
[29] Murali S and De Micheli G 2004 SUNMAP: A Tool for Automatic Topology Selection and Generation for NoCs // *41st Conference on Design Automation (DAC'04)* pp 914–919
[30] Murali S 2007 *Methodologies for Reliable and Efficient Design of Networks on chips* (Stanford University) p 272
[31] Srinivasan K and Chatha K S 2005 A technique for Low Energy Mapping and Routing in Network-on-chip Architectures *Proceedings of the 2005 International Symposium on Low Power Electronics and Design (ISLPED'05)* pp 387–392
[32] Hansson A, Goossens K and Radulescu A 2007 A Unified Approach to Mapping and Routing on a Network-on-chip for Both Best-effort and Guaranteed Service Traffic *VLSI Design* pp 1–16
[33] Starobinski D, Karpovsky M and Zakrevski L A 2003 Application of Network Calculus to General Topologies Using Turn-prohibition *IEEE/ACM Transactions on Networking (TON)* **11** 3 pp 411–421
[34] Ascia G, Catania V and Palesi M 2004 Multi-objective Mapping for Mesh-based NoC Architectures *Proceedings of the 2nd IEEE/ACM/IFIP International Conference on Hardware/software Codesign and System Synthesis (CODES+ISSS'04)* pp 182–187
[35] Lei T and Kumar S 2003 A Two-step Genetic Algorithm for Mapping Task Graphs to a Network on Chip Architecture *Euromicro Symposium on Digital System Design* pp180–187
[36] Hung W, Addo-Quaye C, Theocharides T, Xie Y, Vijakrishan N and Irwin M J 2004Thermal-aware IP Virtualization and Placement for Networks-on-chip Architecture *IEEE International Conference on Computer Design: VLSI in Computers and Processors (ICCD 2004)* pp 430–437
[37] Skadron K, Stan M R and Huang W 2003 Temperature-aware Microarchitecture *Proceedings of the 30th Annual International Symposium on Computer Architecture (ISCA'03)* **31** 2 pp 2–13
[38] Whelihan D 2003 *The NOCsim Simulator Users Guide: Version 2.0* (Pittsburgh: CMU) p 51

[39] Hung W, Xie Y and Vijaykrishnan N 2005 Thermal-aware Floorplanning Using Genetic Algorithms *Sixth International Symposium on Quality of Electronic Design (ISQED 2005)* pp 634–639
[40] Xie Y and Hung W 2006 Temperature-Aware Task Allocation and Scheduling for Embedded Multiprocessor Systems-on-Chip (MPSoC) Design *Journal of VLSI Signal Processing Systems for Signal, Image and Video Technology* **45** 3 pp 177–189
[41] Maqsood T, Bilal K and Madani S A 2016 Congestion-aware core mapping for Network-on-Chip based systems using betweenness centrality *Future Generation Computer Systems* In press
[42] Shin D and Kim J 2004 Power-aware Communication Optimization for Networks-on-chips with Voltage Scalable Links *International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS 2004)* pp 170–175
[43] Schmitz M T and Al-Hashimi B M 2001 Considering Power Variations of DVS Processing Elements for Energy Minimization in Distributed Systems *Proceedings of the 14th International Symposium on Systems Synthesis (ISSS'01)* pp 250–255
[44] Wang C, Zhu Y, Jiang J, Qiu M and Wang X 2017 Dynamic application allocation with resource balancing on NoC based many-core embedded systems *Journal of Systems Architecture* **79** pp 59–72
[45] Carvalho E, Calazans N and Moraes F 2007 Congestion-aware task mapping in NoC-based MPSoCs with dynamic workload *VLSI, 2007. ISVLSI '07. IEEE Computer Society Annual Symposium* pp 459–460
[46] Carvalho E, Ipiranga A and Alegre P 2007 Heuristics for dynamic task mapping in NoC-based heterogeneous MPSoCs *Rapid System Prototyping (RSP 2007) 18th IEEE/IFIP International Workshop* pp 34–40
[47] Fattah M, Daneshtalab M, Liljeberg P and Plosila J 2013 Smart hill climbing for agile dynamic mapping in many-core systems *Proceedings of the 50th Annual Design Automation Conference on - DAC '13* p 201
[48] Li D and Wu J 2016 Energy-efficient contention-aware application mapping and scheduling on NoC-based MPSoCs *Journal of Parallel and Distributed Computing* **96** pp 1–11