



HAL
open science

Design Exploration Framework for 3D-NoC Multicore Systems under Process Variability at RTL level

Charles Emmanuel Effiong, Abdoulaye Gamatié, Gilles Sassatelli

► **To cite this version:**

Charles Emmanuel Effiong, Abdoulaye Gamatié, Gilles Sassatelli. Design Exploration Framework for 3D-NoC Multicore Systems under Process Variability at RTL level. [Research Report] LIRMM (UM, CNRS). 2018. lirmm-01870671

HAL Id: lirmm-01870671

<https://hal-lirmm.ccsd.cnrs.fr/lirmm-01870671>

Submitted on 8 Sep 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Design Exploration Framework for 3D-NoC Multicore Systems under Process Variability at RTL level

Charles Effiong, Abdoulaye Gamatie and Gilles Sassatelli

Abstract—This paper presents an RTL design and evaluation framework allowing the designer to easily build and analyze 3D NoC models with customizable defect threshold on Through-Silicon-Via (TSV) vertical links. The framework provides enough flexibility for addressing 3D NoC design issues, such as process variability, which can introduce open-resistive TSVs in the design, caused by impurities and/or defect during manufacturing process. Such TSVs lead to slower data transfers compared to non defective TSVs. To illustrate the usage of the framework usage, we evaluate typical application mappings in a 3D NoC multicore system to mitigate the performance penalty related to process variability.

Index Terms—Three-Dimensional Network-on-Chip, Multicore Systems, Design Evaluation, Defective TSV, Process Variability

I. INTRODUCTION

Over the last decade, three-dimensional (3D) integration has been promoted as a solution to the memory bottleneck resulting from pin issues in planar circuits. It increases the performance and the density of systems in package, while potentially reducing manufacturing cost and power consumption [1]. The central idea behind 3D integration is to exploit the vertical dimension by stacking planar dies communicating through pillars represented by shorter Through-Silicon-Via (TSV) vertical links (at μm scale). TSVs provide low power, faster and denser data transfer links compared to conventional PCB-level inter-chip communication. They are therefore excellent candidates for accelerating communications in Networks-on-Chip (NoCs). Fig. 1 illustrates a partially connected three-layer 3D NoC.

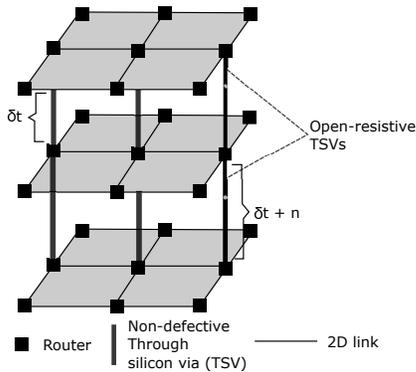


Fig. 1. Partially connected 3D NoC including with open-resistive and non-defective TSVs: δt represents the traversal delay of non-defective TSVs, and $\delta t + n$ denotes n higher delay due to open-resistive defect.

An important challenge about 3D-TSV integration lies in manufacturing process variability due to the rather complex additional process steps required for the physical synthesis of TSVs, the creation of micro-bumps for electrical connection and the final die bonding [2]. This variability reduces the manufacturing yield and often requires using more conservative TSV timing definitions. The delay of a TSV link can indeed vary significantly due to defects and/or impurities that are introduced during the manufacturing process [3]. Such defects

are known as *open-resistive defect*. Open-resistive TSVs maintain electrical connection between dies, albeit inducing higher signal propagation delays [4] due to the higher resistance. Fig. 1 depicts two open-resistive TSVs in the 3D NoC. As inter-die communication often takes place through asynchronous protocols, this results in 3D NoC for which 3D links operate at different speeds, thereby resulting in potentially degraded application performance. Performance guarantee under process variability is thus a major design challenge.

Defective TSVs can be identified and replaced by applying post-silicon validation test. This approach requires a prototype of the actual silicon, which is costly and incurs design time overhead. Redundant TSVs are advocated in 3D designs for alleviating defective TSVs [5]. This increases silicon area, cost and design complexity. Another approach promotes the use of asynchronous delay-insensitive logic for TSVs links [6]. This way, the links can be exploited regardless of their variable traversal delays. Nevertheless, this may result in detrimental asymmetric communication performance in NoCs [7].

This paper briefly describes a simulation framework resulting from a number of studies that we previously conducted on NoC design, first in [8] [7] in which the focus was on TSV modeling, then later in [9] [10] where a support for application trace simulation on mesh NoCs has been addressed. In the present paper, the contribution consists of a flexible RTL design and evaluation framework allowing designers to easily build 3D NoC models with customizable spatial defect distributions so as to analyze the impact on performance.

The framework takes 3D defect maps as inputs and generates 3D NoC models with accurate timing-annotated routers and links. As an illustration, we show how application mapping in 3D NoC based multicore systems can be mitigate the performance penalty due to process variability.

II. 3D NOC EVALUATION FRAMEWORK

The proposed design and evaluation framework is summarized in Fig. 2. It includes a library of 3D NoC templates that the user

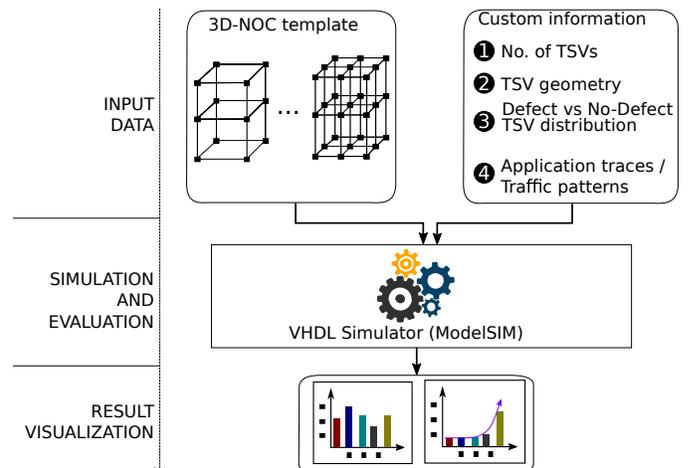


Fig. 2. Implemented evaluation framework

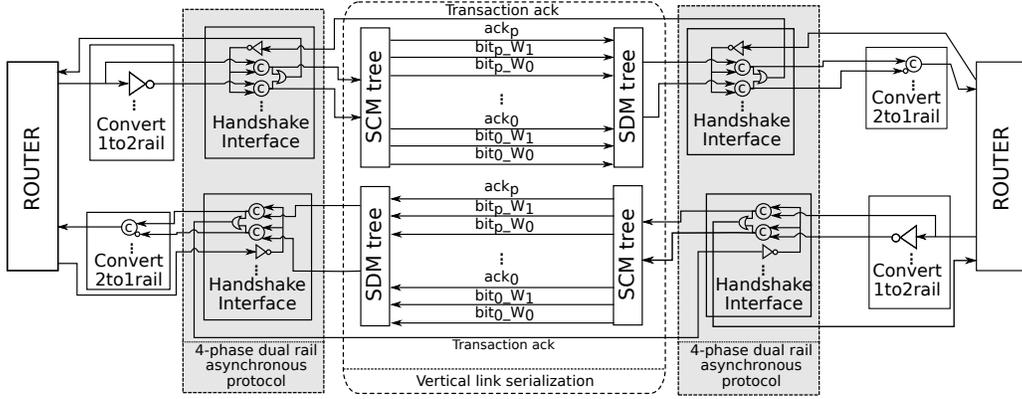


Fig. 3. TSV bundle modeling

can easily instantiate and customize according to desired design parameters. Architecture-specific parameters include TSV geometry / specifications and spatial parametric distribution of defective TSVs. This last feature is particularly relevant because of the availability of defect maps (resulting from process characterization) for 3D TSV processes in which recurring patterns are often observed. Application-specific parameters include NoC architecture parameters (bus width, buffer sizes etc.) alongside traffic patterns or application trace mapping for simulation. A parameterized mixed RTL/gate-level VHDL simulation model is produced and used for performance analysis.

The use of asynchronous communication protocols in 3D TSV systems is increasingly preferred for a number of practical reasons, among which facilitating clock distribution, timing closure [3] and better handling of variability [4]. The considered 3D NoC is for this reason Globally Asynchronous Locally Synchronous (GALS). Two asynchronous communication schemes are used: i) bi-synchronous FIFOs for intra-die communication between two routers, and ii) fully asynchronous serialized vertical links for communication between routers belonging to different dies (see Fig. 3). Bi-synchronous FIFOs provide a reliable and area-efficient interface for routers operating at different clock frequencies [11]. Vertical links employ fully asynchronous quasi delay-insensitive asynchronous logic [6]. 32-bit data words are serialized, transmitted and deserialized upon reaching the input port of the destination router. Data on the vertical links is encoded using a four-phase dual-rail asynchronous protocol, which uses two wires to encode one bit of information. An additional wire is used to send back an acknowledgement from the receiver. Therefore, a total of three TSVs are needed to transmit one data bit on the vertical link. Using asynchronous logic makes it possible to exploit each TSV link at its maximum possible throughput, i.e. depending solely on the propagation delays and not constrained by any clock synchronization mechanism.

In Fig. 3, *single-rail* data from the synchronous domain are converted to *dual-rail* data since the four-phase dual-rail protocol is utilized. For this reason a converter i.e. *Convert1to2rail* in Fig. 3 between the router and the handshake interface is required. Conversely, data streams coming from the asynchronous domain are converted to *single-rail data* for the router. Hence, a dual-rail to single-rail converter is inserted between the handshake interface and the receiving router as shown in Fig. 3. The handshake interface is composed of *c-gate* or *muller gates* and an *or-gate*. The *or-gate* serves as a *completion-detector* that sends back an acknowledgment to the sender when valid signals are produced at the outputs of the *c-gates*. The *c-gate* is a state holding an element that produces "1" at the output when both inputs are "1" and a "0" at the output when

both input are "0". Otherwise, the previous output is maintained [12]. In Fig. 3, each arrow represents a TSV. The serial channel consists of trees of autonomous multiplexers used for serializing, i.e., self-controlled multiplexers (SCM) and de-serializing, i.e., self-controlled-demultiplexers (SDM) [6] data traversing the vertical links. For a 3D-NoC with N-bits communication data, 3N TSVs are needed to transmit the data through a vertical link, i.e., no serialization. This is because, 3 TSVs (i.e., $bit_i_{W_0}$, $bit_i_{W_1}$ and ack_i) are needed to transmit 1 parallel bit data through the link as shown in Fig. 3. Serialization is used to reduce the number of TSVs in the network.

The additional TSV link referred to as *Transaction ack* in Figure 3, is used by the handshake interface to connect the asynchronous channel with the router and to inform the router when a new transaction can be initiated. For this signal to be valid, the receiving router must be able to receive a data and data corresponding to the previous transaction must have been sampled. Since the architecture uses purely asynchronous logic design, the serialization subsystem performance solely depends on propagation delays of both gates and TSVs. The unit latency of each TSV is then fed as a parameter to each instance of asynchronous serializer. This approach makes it possible to accurately analyze various metrics such as bandwidth and communication latencies in the NoC.

III. EXAMPLE OF DESIGN ANALYSIS

A. Impact of process variability on 3D-NoC performance

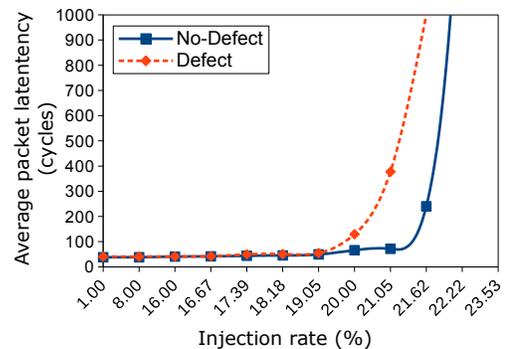


Fig. 4. Impact of process variability on NoC performance (uniform traffic).

The first evaluation that we address concerns the possible impact of open-resistive TSVs in a 3D-NoC. For this purpose, we instantiate two versions of a (3x3x3)-NoC, operating at 1GHz. The first version does not contain any defective TSV, while the second contains 15% of open-resistive TSVs. Here, the traversal latency of non defective

TSV links is considered between 12ps and 18ps, while that of open-resistive TSV links is between 23ps and 48ps. Communication routed through such defective TSVs may therefore suffer from lower bandwidth / higher latency. The network uses ZXY dimension ordered routing and a credit-based flow control without virtual channels.

Fig. 4 shows the average latency of the 3D NoC versus different packet injection rates under uniform traffic. The curve referred to as "No-Defect" (respectively "Defect") denotes the 3D NoC without any (respectively with 15%) defective TSVs. As expected, the saturation threshold enabled by "No-Defect" version is better than that obtained with "Defect" version. Indeed, in the latter case some localized contentions occur due to TSV links with open-resistive defects, resulting in overall increased communication latency. When operating at injection rates near the saturation threshold significant increase in latency is observed as can be seen in Fig. 4 in which for injection rate of 21% a 4-fold increase in latency is observed.

A similar evaluation is conducted by simulating the execution of four application traces on the 3D NoC. For this purpose, we inject traces from real world application benchmarks into the network. Here, we consider applications consisting of sets of tasks which can be executed concurrently on different cores. The chosen applications include: video conference encoder (VCE), Wi-Fi baseband receiver (WIFI), multimedia system (MMS) and E3S consumer benchmarks [13]. The characteristics of these applications are given in Table I.

TABLE I
APPLICATION CHARACTERISTICS

Application	No of tasks	Comm. volume (no. of packets)
VCE	24	52060
MMS	25	644098
E3S	12	131
WIFI	25	2160798

From the different application simulations, we compute the resulting average packet latency in the NoC, as shown in Fig. 5. Here, also we observe that "No-Defect" 3D NoC version provides a better average latency compared to the "Defect" version. It is important to notice that the gap between both latencies, denoted as Δ in Fig. 5, varies according to applications. It is minimal for MMS while it is maximal for VCE. In this experiment, all four applications are mapped in a similar way so as to keep a common basis for comparison. The next section explores the impact of mapping variation.

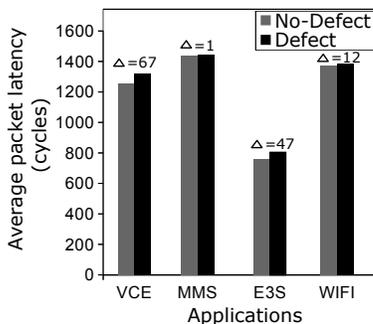


Fig. 5. Impact of process variability on NoC performance (4 applications).

B. Mitigating the impact of process variability via mappings

Application mapping is crucial in multi/many core systems because of multiple application requirements (real-time constraints, performance, energy consumption, etc.) [14]. Mapping heuristics determine

how application tasks are mapped onto cores. In order to carry out performance exploration, we consider a few mapping heuristics and evaluate the performance of the 3D NoC for each mapping. On the other hand, only static application mapping is applied here for the sake of simplicity. Finally, only one task is assumed to be mapped on each core attached to a router in the 3D NoC. The considered mappings are briefly explained below.

- 3D Minimum Communication mapping.** In 3D minimum communication mapping (also referred to as 3D-MinComm), tasks with highest mutual communication activities are mapped close to each other on 2 neighbouring dies such that they can communicate using TSV links. The goal of this mapping is to exploit the high-bandwidth TSV links in order to optimize performance. Fig. 6a(a) shows a 3D-MinComm mapping of an application with four tasks onto a two-tier 3D NoC. In the application task graph, nodes represent tasks, while the number between two nodes represents the exchanged communication volume. Each node on the communication architecture has a corresponding x,y,z coordinates within the 3D NoC. If two tasks exchange large communication volume, 3D-MinComm attempts to map the first task on a node at address x_n, y_n, z_n and the second task on another node at address x_n, y_n, z_{n+1} or x_n, y_n, z_{n-1} . This is illustrated in Fig. 6a(i) where Task T_0 is mapped on a core with address x_1, y_1, z_0 and Task T_1 is mapped on a core with address x_1, y_1, z_1 .
- Least Communication Middle mapping.** In *Least Communication Middle mapping* (referred to as LCM) mapping, tasks with lowest mutual communication activities are preferably mapped onto the middle tier(s). The motivation behind this choice is to balance the traffic load of the network since a high number of packets tend to traverse the middle tier(s). Fig. 6a(ii) however illustrates a specific case without any middle layer. Here, the least communicating tasks T_2 and T_3 are mapped around the central TSV.
- Critical Path mapping.** In *Critical Path* mapping (referred to as CP), the tasks that occur on the longest communication path of an application task graph are first mapped before the other tasks. The goal is to reduce network contention and packet latency on those "critical" paths, while exploiting TSVs as much as possible. As shown in Fig. 6a(iii) tasks T_0, T_1, T_2 and T_3 , which are all on the critical path are mapped in such a way to exploit TSVs.

Fig. 7 summarizes the average packet latency resulting from the simulation of the previous four applications, according to the above mapping heuristics. Here, all reported scenarios take into account the presence of 15% open-resistive TSVs in the 3D NoC. Note that the results already reported in Fig. 5 rely on the LCM mapping. Now, in Fig. 7 we observe the improvements made possible by alternative mappings, i.e., 3D-MinComm and CP. This suggests that the choice of suitable mapping heuristics helps to mitigate the performance penalty related to process variability in 3D NoC systems.

Fig. 6b focuses on the VCE application and shows the link utilization of the different mappings. NoC links with a traffic load that is greater than a given threshold are considered as *over-loaded*. It can be observed that for 3D-MinComm mapping, most of the TSVs links are over-loaded, while most 2D-links are either idle or under-utilized (see Fig. 6b(i)). The reason being that this mapping attempts to exploit the high bandwidth TSVs. On the other hand, the CP mapping creates hotspots in the first two layers, while the top-most layer is under-utilized as shown in Fig. 6b(iii). Hotspots appear on each layer of the network for LCM mapping as shown in Fig. 6b(ii). Compared to the other mappings, a greater percentage of the network

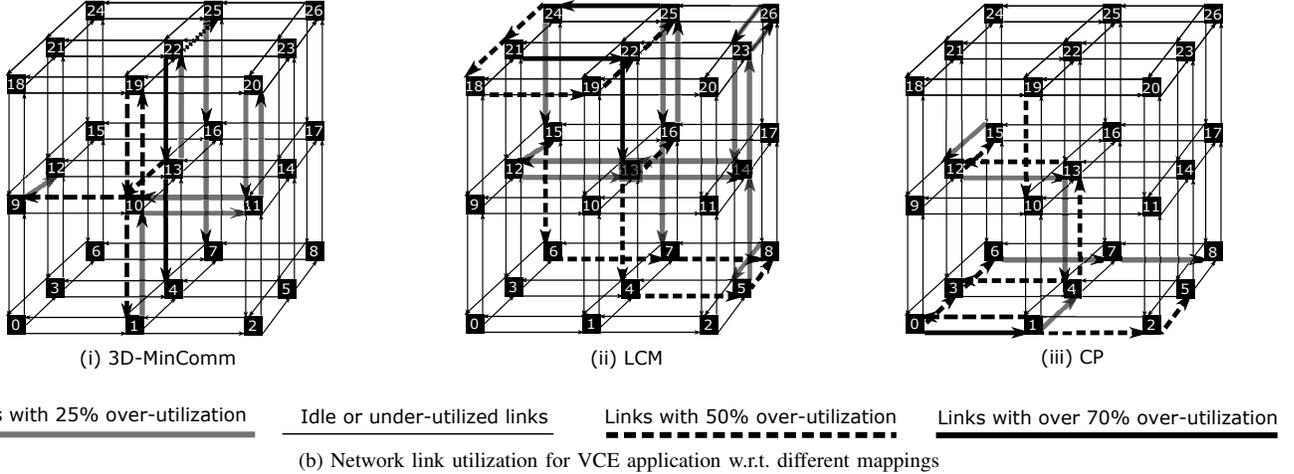
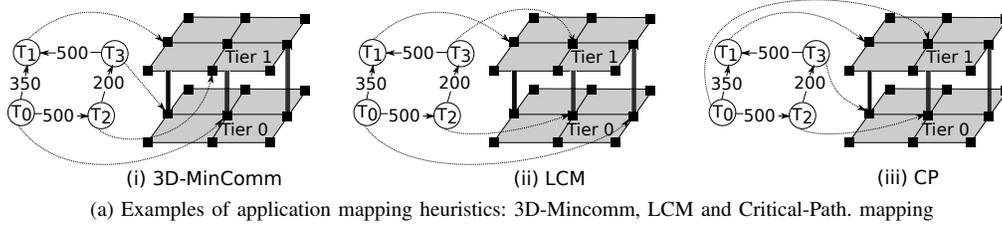


Fig. 6. Simulation results for the network.

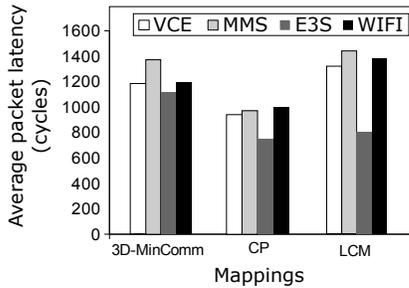


Fig. 7. Evaluation of different mappings in presence of process variability.

links are over-utilized with LCM mapping.

IV. CONCLUDING REMARKS AND PERSPECTIVES

We presented a design and evaluation framework aiming to enable an easy and flexible analysis of 3D NoC models in presence of process variability, which results in lower quality TSVs, caused by impurities and/or defect during manufacturing process. As this leads to performance penalty, having a tool that makes it possible to assess performance degradation and possibly apply mitigation techniques such as variability-aware task mappings is key. We illustrated one example showing how application mappings can be leveraged to mitigate this penalty. This experiment can be further enhanced by means of taking into account realistic defect maps in which certain areas (e.g. chip boundaries) are more affected than others.

Users could benefit from the flexibility of the framework for exploring further design challenges, e.g., new routing paradigms [15] by designing TSV defect-aware mapping strategies (assuming that defects locations have been determined in post fab testing). Further directions could address thermal and power consumption modeling so as to analyze process variability impact w.r.t. those metrics, beyond performance issues.

ACKNOWLEDGMENT

The authors would like to thank their colleagues who contributed to the elaboration of this framework. This work has been partly supported by the French ANR agency under the grant ANR-15-CE25-0007-01, within the CONTINUUM project.

REFERENCES

- [1] V. F. Pavlidis, I. Savidis, and E. G. Friedman, *Three-dimensional integrated circuit design*. Morgan Kaufmann, 2017.
- [2] C. Richardson, M. Tsuriya, and H. Fu, "Technology roadmap overviews and future direction through technology gaps," in *IEEE International Conference on Electronics Packaging*, 2017, pp. 35–40.
- [3] A. W. Topol, D. C. L. Tulipe, L. Shi, D. J. Frank, K. Bernstein, S. E. Steen, A. Kumar, G. U. Singco, A. M. Young, K. W. Guarini, and M. Jeong, "Three-dimensional integrated circuits," *IBM Journal of Research and Development*, vol. 50, no. 4.5, pp. 491–506, July 2006.
- [4] C. Metzler, A. Todri, A. Bosio, L. Dilillo, P. Girard, and A. Virazel, "Through-silicon-via resistive-open defect analysis," in *2012 17th IEEE European Test Symposium (ETS)*, May 2012, pp. 1–1.
- [5] A. C. Hsieh and T. Hwang, "Tsv redundancy: Architecture and design issues in 3-d ic," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 20, no. 4, pp. 711–722, April 2012.
- [6] F. Darve, A. Sheibanyrad, P. Vivet, and F. Petrot, "Physical implementation of an asynchronous 3d-noc router using serial vertical links," in *IEEE Annual Symposium on VLSI*, July 2011, pp. 25–30.
- [7] C. Effiong, V. Lapotre, A. Gamatie, G. Sassatelli, A. Todri-Sanial, and K. Latif, "On the performance exploration of 3d noCs with resistive-open tsvs," in *IEEE Annual Symposium on VLSI*, July 2015, pp. 579–584.
- [8] A. Kologeski, F. L. Kastensmidt, V. Lapotre, A. Gamatié, G. Sassatelli, and A. Todri-Sanial, "Performance exploration of partially connected 3d noCs under manufacturing variability," in *IEEE 12th International New Circuits and Systems Conference, NEWCAS 2014, Trois-Rivieres, QC, Canada, June 22-25, 2014*. IEEE, 2014, pp. 61–64. [Online]. Available: <https://doi.org/10.1109/NEWCAS.2014.6933985>
- [9] C. Effiong, G. Sassatelli, and A. Gamatié, "Distributed and dynamic shared-buffer router for high-performance interconnect," in *Proceedings of the Eleventh IEEE/ACM International Symposium on Networks-on-Chip, NOCS 2017, Seoul, Republic of Korea, October 19 - 20, 2017*, A. Jantsch, H. Matsutani, Z. Lu, and Ü. Y. Ogras, Eds. ACM, 2017, pp. 2:1–2:8. [Online]. Available: <http://doi.acm.org/10.1145/3130218.3130223>

- [10] —, “Scalable and power-efficient implementation of an asynchronous router with buffer sharing,” in *Euromicro Conference on Digital System Design, DSD 2017, Vienna, Austria, August 30 - Sept. 1, 2017*, H. Kubátová, M. Novotný, and A. Skavhaug, Eds. IEEE Computer Society, 2017, pp. 171–178. [Online]. Available: <https://doi.org/10.1109/DSD.2017.55>
- [11] I. M. Panades and A. Greiner, “Bi-synchronous fifo for synchronous circuit communication well suited for network-on-chip in gals architectures,” in *Symp. on Networks-on-Chip (NOCS'07)*, 2007, pp. 83–94.
- [12] J. Spars and S. Furber, *Principles of Asynchronous Circuit Design: A Systems Perspective*, 1st ed. Springer Pub. Company, Inc., 2010.
- [13] A. T. Tran and B. Baas, “Noctweak: a highly parameterizable simulator for early exploration of performance and energy of networks on-chip,” ECE Dep., Univ. of Cal., Davis, Tech. Rep. ECE-VCL-2012-2, 2012.
- [14] V. Kiani and M. Reshadi, “Mapping multiple applications onto 3d noc-based mpsoCs supporting wireless links,” *The Journal of Supercomputing*, vol. 73, no. 5, pp. 2187–2213, May 2017.
- [15] S. H. S. Rezaei, A. Mazloumi, M. Modarressi, and P. Lotfi-Kamran, “Dynamic resource sharing for high-performance 3-d networks-on-chip,” *IEEE Computer Architecture Letters*, vol. 15, no. 1, pp. 5–8, Jan 2016.