IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 28, 2019, accepted September 10, 2019, date of publication September 18, 2019,
date of current version September 30, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2942129

NoCs in Heterogeneous 3D SoCs: Co-Design of Routing Strategies and Microarchitectures

JAN MORITZ JOSEPH[1], LENNART BAMBERG[2], DOMINIK ERMEL[1], BEHNAM RAZI PERJIKOLAEI[2], ANNA DREWES[1], ALBERTO GARCÍA-ORTIZ[2], AND THILO PIONTECK[1]

[1]Institut für Informations- und Kommunikationstechnik, Otto-von-Guericke-Universität Magdeburg, 39106 Magdeburg, Germany
[2]Institute of Electrodynamics and Microelectronics, University of Bremen, 28359 Bremen, Germany

Corresponding author: Jan Moritz Joseph (jan.joseph@ovgu.de)

This work was supported by the German Research Foundation (DFG) under Project PI 447/8 and Project GA 763/7.

ABSTRACT Heterogeneous 3D System-on-Chips (3D SoCs) are the most promising design paradigm to combine sensing and computing within a single chip. A special characteristic of communication networks in heterogeneous 3D SoCs is the varying latency and throughput in each layer. As shown in this work, this variance drastically degrades the network performance. We contribute a co-design of routing algorithms and router microarchitecture that allows to overcome these performance limitations. We analyze the challenges of heterogeneity: Technology-aware models are proposed for communication and thereby identify layers in which packets are transmitted slower. The communication models are precise for latency and throughput under zero load. The technology model has an area error and a timing error of less than 7.4% for various commercial technologies from 90 to 28nm. Second, we demonstrate how to overcome limitations of heterogeneity by proposing two novel routing algorithms called $Z^{+}(XY)Z^{-}$ and ZXYZ that enhance latency by up to 6.5× compared to conventional dimension order routing. Furthermore, we propose a high vertical-throughput router microarchitecture that is adjusted to the routing algorithms and that fully overcomes the limitations of slower layers. We achieve an increased throughput of 2 to 4× compared to a conventional router. Thereby, the dynamic power of routers is reduced by up to 41.1% and we achieve improved flit latency of up to 2.26× at small total router area costs between 2.1% and 10.4% for realistic technologies and application scenarios.

INDEX TERMS 3D integrated circuits, network on chip, heterogeneous integration, monolithic stacking.

I. INTRODUCTION

3D integration is one of the most promising paradigms to meet the perpetual demand for chips with higher performance, less power consumption and reduced area [1]. Therefore, many designs and architectures have been proposed: 3D-integrated DRAM subsystems, 3D-FPGAs [2], [3], and even 3D-Vision Systems-on-Chip (3D VSoC) with stacked sensors [4]. Recently, Intel introduced ''Lakefield'', in which Foveros 3D technology is used to stack multicore processors, FPGAs and DRAM [5]. Other manufactures such as Xilinx are also targeting 3D integration [6]. Ultimately, stacking dies even tackles fundamental limits of computation by asymptotically reducing computation time from t to $t^{0.75}$ [7]. All these works impressively demonstrate the advantages of 3D integration that are even exploited in commercial applications.

Despite the aforementioned incremental advancements, 3D integration enables one game-changing key innovation: It allows for heterogeneous integration, in which dies in disparate technologies, i.e. analog, mixed-signal, memory and logic are stacked. As stated in [8], this is ''the ultimate goal of 3D integration'' because it allows to align the requirements of components with the technology characteristics of their die. This is advantageous for applications, in which components with different requirements are integrated to a single SoC: [9] introduces an architecture for Internet of Things (IoT) stacking wireless sensors, RF communication, data processing and energy scavenging. In high-performance processors, interleaving of dedicated dies with either memory or processing increases performance [10], with exemplary designs [11]–[14]. Especially, vision

The associate editor coordinating the review of this manuscript and approving it for publication was Abdallah Kassem.

applications can profit of heterogeneity: 3D VSoCs [4] combining image sensing, mixed-signal conversion and digital image processing. Thus, VSoCs demand heterogeneous integration, intrinsically. In recent research, a SoC is proposed for self-driving cars that realizes up to 10,000 frame per second [15]. Going further, the mixed-signal layers can implement analog accelerators, for instance to calculate a cellular neural network [16]. Such accelerators have been implemented in 180nm [17] and 130nm [18] CMOS technology. To summarize, heterogeneous integration enables to build novel and more efficient systems in previously challenging application areas.

To unleash the full potential of 3D integration, the used interconnection architectures must offer very good PPA (power, performance, area). In general, there are two approaches to distribute interconnects in the third dimension: First, the components of the interconnect architecture can be distributed in 3D. For instance, [19] enables packet transmission in vertically partially connected 3D NoCs using elevator-first routing. [14] presents a NoC that connects cores for a neural network using TSVs. Also, works on inductive coupling have been made [20]. Second, the components of the interconnect itself can be split-up over layers and be distributed. For instance, MIRA [21] is such a 3D stacked router that achieves up to 51% latency improvement for synthetic workloads. While all these works are well-suited for interconnects on homogeneous 3D integration, they do not explicitly account for varying integration properties within the interconnect from heterogeneous 3D integration.

Since the integration properties of any interconnect will differ if it spans multiple heterogeneous layers of a chip, heterogeneous 3D interconnects must account for this property. There are two main integration issues as illustratively shown in Fig. 1: First, *the components of an interconnection architecture are not purely synchronous*, since logic in digital nodes can be clocked faster than in mixed-signal nodes; the clock deviation can be by a magnitude and not only a small deviation. Second, *the feature size of (identical) components differs with technology*. Traditional router architectures cannot be applied, because these cannot cope with different clock speeds or yield unbearable costs in mixed-signal layers. This will be discussed in Sec. II in detail. Because of the aforementioned two arguments, novel models, architectures and concepts are required: For instance, [22] proposes TSV power models that account for heterogeneity and low-power coding with less than 1% error compared to bit-level accurate simulations. In another example, in ref. [23] input buffer distributions among layers are evaluated with area saving between 8.3% and 28% and power savings between 5.4% and 15%. More significant improvements are required, which also improve performance. In particular, latency and throughput in heterogeneous 3D interconnects vary per layer due to different clocks and router count. These severe effects of heterogeneity were, previously to this work, unconsidered for heterogeneous 3D interconnects.

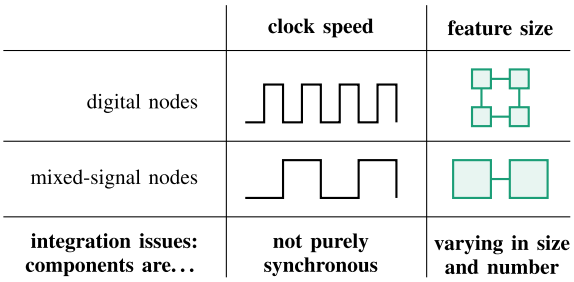

FIGURE 1. Challenges for heterogeneous interconnection architectures.

The aforementioned influence of heterogeneity on interconnects requires a novel approach that *simultaneously* considers routing strategies and architectures; in a separate design, the full potential of the interconnect cannot be unleashed since either throughput or latency are limited. Therefore, this paper provides the following specific contributions:

Contribution 1: We introduce models for network throughput and latency; and we thereby show that heterogeneity drastically degrades network performance.

Contribution 2: We contribute two new principles for routing and two concrete routing algorithms reducing latency. The algorithms exploit the variation in communication speed between layers.

Contribution 3: We propose a novel co-designed router and routing strategies that tackles throughput, which is limited by the slowest router in a packet's path. The router architecture and the used simulation tools to generate results are published open-source.

By these contributions, we tackle throughput and latency limitations in heterogeneous interconnects by an integrated approach. The source code of the simulation tool and the router architecture are available at [24].

The work is structured as follows: We discuss limitations of related approaches for heterogeneous 3D SoCs (Sec. II). To quantify the effects of heterogeneity, we propose a model for technology (Sec. III) and communication (Sec. IV). We thereby show a drastically negative impact on the network performance due to slower packet provision (Sec. V). Based on these findings, we contribute two novel routing algorithms to overcome this issue by improving the latency (Sec. VI). Further, we propose a router architecture adopted to these routing algorithms, which fully nullifies the negative influence of heterogeneity and improves network throughput (Sec. VII). Finally, we present the accuracy of our models and that latency, throughput and dynamic power are improved at minor hardware costs (Sec. VIII). In this section, we also present a comprehensive case study for a realistic system that demonstrates positive effects of our approach under practical conditions including congestion. We thereby discuss in Sec. IX all relevant aspects of routing in

NoCs for heterogeneous 3D SoCs and contribute that, solely, a co-design of algorithms and architectures allows for efficient heterogeneous 3D interconnects.

II. RELATED WORK

As already stated in the introduction, we discuss here why existing 3D interconnects are not considering heterogeneity sufficiently. Therefore, we focus on three individual topics that are also covered by this work: First, we highlight the differentiating aspects of our models for communication in heterogeneous 3D interconnects in comparison to other models. Second, we turn the spotlight to routing and we discuss related approaches in 3D systems. Third, we consider existing architectures for 3D interconnects. Finally, we combine the approaches and argue, why these existing methods and architectures are not well-applicable to build heterogeneous 3D interconnects.

Modeling properties of both technology and communication in interconnects is a well-established research topic with a wide range of works. There are many works on 3D NoCs modeling performance (e.g. [25]) or power and area (e.g. [26]). The majority of the performance models can be applied only under zero load because non-dynamic behavior is easier to model. For instance, in [27], a performance model is proposed with focus on Quality of Service (QoS). It assumes constant service time and purely synchronous routers. This cannot be applied to heterogeneous 3D interconnects, as those are not purely synchronous. Reference [28] models average latency, throughput and network characteristics without QoS guarantees. Again, this model cannot be applied for heterogeneity, because the model assumes one globally synchronized clock. In a similar approach, [29] models performance and power of NoCs with wormhole routers. Again, only homogeneous router architectures and technologies are covered. In a more sophisticated approach, the dynamic properties, namely load and congestion, are covered by some works, as well, e.g. [25], [30]. A common approach is the use of queueing models [25], in which the dynamic behavior is summarized by network statistics. Although there has been considerable effort to analytically model the behavior of interconnection networks, there is an urgent need for models for heterogeneous 3D interconnects as their properties, especially differences in clock speeds within a network, have not been considered sufficiently so far.

Routing for 3D interconnects is, just as models, a very common field, as well. The most traditional approach is the extension of strategies from 2D by one dimension. For instance, dimension-ordered routing can be directly used in 3D, which has already been done over a decade ago [31]. Since then, many improvements have been proposed: DyXYZ [32] is a fully adaptive routing considering congestion in 3D. Elevator-first routing [19] enables packet transmission in vertically partially connected 3D NoCs. LA-XYZ [33] uses look-ahead strategies to improve latency and throughput by approximately 45% and while reducing power by 15.9%. Furthermore, fault-tolerance can be implemented, as well [34]. Despite the large number of papers in this area, none of these works target heterogeneous 3D interconnects, in which the transmission speed is not only impeded by congestion but also and more fundamentally by the used technology nodes.

Architectures for 3D interconnects have also been researched for many years. These solutions manly target performance increases: Ref. [31] was one of the first routers for 3D systems. Extending standard architectures, [35] proposes express virtual channels (EVC) which combine conventional full-swing, short-range wires and low-swing, multi-drop wires, achieving 25% latency reductions. This ultimately lead to the single-cycle NoC router SMART [36] with 60% latency savings. Rather popular is MIRA [21], which was the first 3D-stacked router. All router components, except central arbitration, are sliced in logical-equal parts and are distributed. Thereby, up to 51% latency improvement for synthetic workloads are achieved. For heterogeneous 3D integration, architectures for not-purely synchronous communication are highly relevant. There exist only a limited number of works: Ref. [37] proposes a router architectures which is limited by the slower clock frequency for packets traveling along the asynchronous path. However, enabling asynchronous communication between routers is not a common topic due to its large overhead and limited practical relevance in homogeneous 3D systems. The limitations of non-purely synchronous communication between routers as intrinsically found in heterogeneous 3D interconnects is a key for their integration and, therefore, is one of the key contributions of this publication.

To summarize our discussion of the related work, none of the aforementioned works target the special requirements of heterogeneous 3D integration. In terms of *models*, there exist no well-known works on latency and throughput for heterogeneous 3D interconnects. The majority of *routing algorithms* for 3D interconnects do not consider performance differences between routers due to varying technology nodes; yet, this effect is significant as we will show in this work. The works on *architectures* for 3D interconnects assume synchronous routers; yet, routers in heterogeneous 3D systems are not clocked purely synchronous, as this paper also will show. However, works on asynchronous routers do not target to increase the vertical link bandwidth to bridge the throughput gap posed by heterogeneity. Rather, they decrease the bandwidth to increase yield from TSV manufacturing, e.g. by serialization [38]. This is orthogonal to our targets and therefore, these approach cannot be used. Also, distributed architectures such as MIRA cannot be applied to heterogeneous 3D SoCs: First, processing elements would need to be equally distributed among all layers, but are actually located in that layer best suited for their technological requirements. Second, router delay is limited by the slowest layer and router area is dominated by the most expensive layer. To the best of our knowledge, there are no related works which consider the relationship between routing algorithms and architectures but this topic is very relevant in heterogeneous 3D SoCs due

to latency and throughput limitations. Therefore, we see an urgent need to tackle these issues in one integrated design approach: Efficient heterogeneous 3D interconnects are only possible by means of a simultaneous design of routing algorithms and architectures, as demonstrated by this paper.

III. MODELING TECHNOLOGY HETEROGENEITY IN 3D INTERCONNECTS

Heterogeneity influences every metric of the interconnect and we model the influence on area and timing, which are most relevant. The model accounts for any type of commercial technology and any feature size. We do not model power due to the diverse influence parameters; e.g. data transmitted vastly influence power consumption of links, which is hard to model a priori without simulations [39].

We start this section by our technological assumptions for the models. We model an interconnect with NoC routers vertically connected via vertical interconnects. These interconnects can either monolithic inter-tier vias (MIVs) or trans-silicon vias (TSVs) due to their high interconnection density. We use the following model assumptions:

1) The delay of a vertical interconnect is negligibly small, in comparison to horizontal and logic delay. The reason lies therein that vertical interconnects have a constant length of 50μ m due to substrate thickness [40].
2) We neglect modeling area of vertical interconnects because MIVs nearly have no overhead and TSVs have a constant one.
3) Vertically connected routers must not be located at the same physical 2D position (in their layer). Vertical links and routers can be horizontally connected via redistribution. This variability is limited by the link delay. We model this by conversion of router locations to router addresses.
4) We show advantages of our approach in terms of power using simulations only. We do not model the different power properties of horizontal and vertical interconnects as this is a complex topic on its own. For models we kindly refer to [22], [41].
5) We model synchronous routers within layers and not purely synchronous routers between layers, following a GALS approach (globally asynchronous, locally synchronous). This is reasoned as follows: Heterogeneous 3D interconnects will be in non-purely synchronous settings, since components in disparate technologies are potentially clocked at varying speeds and the slowest, synchronous clock wastes performance. Routers within layers, however, are in the same technology and therefore are clocked synchronous.

To summarize, the chosen assumptions are the most common integration principles for 3D interconnects and therefore are a reasonable choice.

Before introducing our models, we explain our notations and definitions: We consider a chip with ℓ layers and their index set $[\ell] = \{1, \ldots, \ell\}$. We assume n-m-mesh topologies

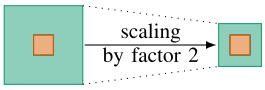

FIGURE 2. Area scaling has reducing parts (green) and constant parts (orange).

of NoCs per layer. The feature size of the technology nodes of layers, measured in [nm], is given by $\tau : [\ell] \rightarrow \mathbb{N}$. We call a chip layer with index ι >>*more advanced node*<< than a layer with index ξ if $\tau(\iota) < \tau(\xi)$ (for easy notation). We define:

Definition 1 (Relative Technology Scaling Factor): Let ξ and ι be the indexes of layers with technologies $\tau(\xi)$ and $\tau(\iota)$ and with $\tau(\xi) > \tau(\iota)$. The relative technology scaling factor Ξ *is:*

$$\Xi(\xi, \iota) := \frac{\tau(\xi)}{\tau(\iota)} \tag{1}$$

A. AREA MODEL

The area, which the communication infrastructure in a layer requires, is influenced by two major factors: The size of an individual router and the number of routers. The effect of both factors is encapsulated into an abstract model. It covers the influence of technology nodes, constraints of synthesis tools and router architectures.

1) AREA OF ROUTERS

Routers in layers in mixed-signal nodes are disproportionately expensive: While routers still will consist of conventional digital circuits, the technology node, e.g. mixed-signal technology, impacts on the size of routing computation, crossbars and buffers, affecting bot combinational and sequential logic. The overall area consists of logic, for which it is commonly known that it reduces its size (ideally) quadratically for more advanced nodes, and the remainder (e.g. power supply) that does not scale approximately and therefore remains constant for different nodes. This is shown illustratively in Fig. 2. These considerations yield an area model of the form $\hat{\alpha}+\alpha\varsigma^2$, in which $\hat{\alpha}$ is the constant part (i.e. the non-scaling part), α is an non-ideality factor (i.e. the deviation of the ideally quadratically scaling parts), and ς is the feature size. By this model we define the area scaling factor as the difference between baseline technology, i.e. the largest node, and any target technology:

Definition 2 (Area Scaling Factor): Let ξ and ι be the indices of two chip layers with technologies $\tau(\xi)$ and $\tau(\iota)$ and with technology difference $\Xi(\xi, \iota)$. The area scaling factor $s_f : (\mathbb{R}) \rightarrow \mathbb{R}$ is given by:

$$s_f(\Xi) := \frac{\alpha + \hat{\alpha}}{\frac{\alpha}{\Xi^2} + \hat{\alpha}} \tag{2}$$

The model assumes that the chip area is normalized to one area unit. The non-ideality factor α denotes, how well the technology scales quadratically. The base technology area offset $\hat{\alpha}$ is dominated by components which do not scale.

Both must be evaluated for the used set of technology nodes. Therefore, a small circuit with typical properties is synthesized, such as a basic router model (see Sec. VIII-A). Then, the parameters can be estimated using function fitting. In an ideal setting, $\alpha = 1$ and $\hat{\alpha} = 0$. As an example, we consider two layers implemented in an ideal theoretical $\tau(1) = 45nm$ and $\tau(2) = 14nm$ technology. The technology scaling factor is $s_f(\Xi(45, 14)) = 10.2$. Between 28nm and 45nm nodes it is $s_f(\Xi(45, 28)) = 2.58$.

2) NUMBER OF ROUTERS

The different technology nodes influence not only the size of routers, but also their number per layer. The scaling factor s_f can also be applied here to approximate a lower bound for the number of additional routers that can be implemented in a more advanced node. In that manner, we model a constant-area NoC per layer, which might not always be the most common integration approach (cf. our case study in Sec. VIII-E). If the area has been non-constant, the router count in faster layers would be reduced. Thus, the model underestimates advantages of our approach and therefore is valid, still.

B. TIMING MODEL

The transmission time of packets is determined by the individual timing of each router and the network topology. We model both characteristics; We consider clock delay of individual routers first and then deduct the propagation speed of packets traversing multiple routers.

1) CLOCK DELAYS

Routers in layers in mixed-signal nodes are potentially slower clocked whilst routers in the more advanced, digital technologies are clocked faster. The ratio, at which the clock delay in different technology nodes scales, is given by the clock scaling factor. There are two effects which influence the clock delay. It is larger than the interconnect delay for large technology nodes; it reduces with node scaling. Interconnect delay does not scale and therefore poses a limit for small nodes. Also, power constrains the maximum achievable clock frequencies. Therefore, the clock scaling factor is modeled by fitting a sigmoid function. Please note that this is an empirical and not a physical model. It has a high accuracy of the fit as shown in in Sec. VIII-A. If another (empirical or physical) model with similar accuracy is used, the results presented in this paper will not change.

Definition 3 (Clock Scaling Factor): Let ξ and ι be the indices of two chip layers with technologies $\tau(\xi)$ and $\tau(\iota)$, with $\tau(\xi) > \tau(\iota)$ and with a relative technology scaling factor $\Xi(\xi, \iota)$. Let c_b be the base clock delay of the layer with index ξ and c_c be the minimum achievable clock delay. Let β be the maximum speedup achievable: $\beta := c_b/c_c$. The clock scaling factor $c_f : (\mathbb{R}) \to \mathbb{R}$ is given by:

$$c_f(\Xi) := \frac{\beta}{1 + \hat{\beta} \exp\left(-\tilde{\beta}\left(\Xi - \bar{\beta}\right)\right)} \tag{3}$$

The function converges to the maximum achievable speedup β. The other parameters must be set by fitting the function to a set of synthesis results (see Sec. VIII-A).

IV. MODELING NoC COMMUNICATION IN HETEROGENEOUS 3D SoCs

We model the horizontal and vertical communication separately, since different factors are relevant: Communication within a layer is synchronous while communication between layers is not always. Our models calculate latency, throughput and transmission speed under zero load.

A. HORIZONTAL COMMUNICATION

The speed at which a packet is transmitted horizontally, at zero load, is called *propagation speed*. The propagation speed differs with technology nodes, since the number of routers and the clock frequency of routers differ between layers. Within a layer, routers are synchronous. The *propagation speed of a packet within a layer is given by the distance traveled divided by the packet latency*. We measure the *distance* that packets travel. All possible positions of routers in a 3D SoC are given by the set $P = \mathbb{R} \times \mathbb{R} \times [\ell]$. The x- and y-coordinates are measured in [m].[1] We use the notation that the symbols p_x, p_y and p_z denote the components of each position $p \in P$. Further, packets have a payload, which is modeled by the number of flits transmitted $l \in L = \mathbb{N}$. Together, the set of packets is given by $D = P \times P \times L$. Packets are transmitted from a current (source) position to a destination position. (Please note, that the current position refers to the location of the packet during transmission. This position changes over time and does not refer to the position the packet was initially injected at into the network.) This yields the definition of the horizontal transmission distance:

Definition 4 (Horizontal Transmission Distance): Let π be a packet with $\pi = (p_1, p_2, l)$, with source node p_1, destination node p_2 and length l. The horizontal transmission distance $s(\pi)$ is defined as the distance between source and destination positions in x- and y- dimension:

$$s(\pi) = \left\| \left(p_{1,x}, p_{1,y}\right) - \left(p_{2,x}, p_{2,y}\right) \right\| \tag{4}$$

For example the distance between source and destination position in x- and y-dimension of a packet $\pi = (p_1, p_2, l)$ is calculated by $s(\pi) := \left\| \left(p_{1,x}, p_{1,y}\right) - \left(p_{2,x}, p_{2,y}\right) \right\|_1$ in a mesh topology. The norm $\|\cdot\|_1$ denotes the Manhattan norm ($\|p\|_1 = \sum_{i=1}^{n} p_n$ for $p \in \mathbb{R}^n$).

The *latency* of a packet is calculated by the cumulative latency each router adds along the path. Each router requires $\delta(\xi)$ clock cycles to process the head flit in the layer $\xi \in [\ell]$. Thereafter, one flit is transmitted each clock cycle until end of packet. A single router finishes the transmission of a single packet with l flits after $\delta(\xi) + l$ cycles. The constant $\rho(\xi)$ is defined as the average distance between routers in the layer ξ. Hence, a packet traverses $s(\pi)/\rho(\xi)+1$

[1] "measured in [...]" refers to SI-units; "$[\ell]$" to the set $\{1, \ldots, \ell\}$.

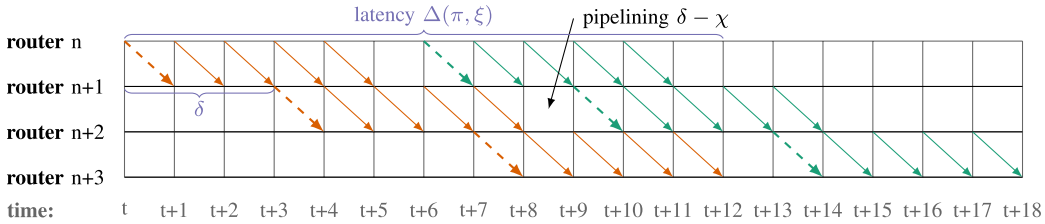

FIGURE 3. Exemplary horizontal communication of two consecutive packets (orange, green).

routers including the destination router. This is illustrated for an example in Fig. 3, in which two consecutive packets are transmitted. In the example, routers have a head delay of $\delta = 3$ and pipelining $\chi = 2$. These considerations yield the following model for horizontal packet head latency that is accurate under assumption of zero load by construction.

Definition 5 (Horizontal Packet Head Latency Under Zero Load): Let π be a packet with $\pi = (p_1, p_2, l)$ and $\xi \in [\ell]$ a layer. The average distance between routers in the layer ξ is $\rho(\xi)$, measured in [nm], and the delay for processing head flits per router is $\delta(\xi)$. The clock delay of routers is $clk(\xi)$, measured in [s]. The horizontal packet head latency under zero load, measured in [s], in layer ξ is

$$\Delta_H(\pi, \xi) = \left(\frac{s(\pi)}{\rho(\xi)} + 1\right) \delta(\xi) clk(\xi). \tag{5}$$

As given in Definition 4, the horizontal transmission distance is measured in [nm], but not in number of hops. Since the horizontal packet head latency is calculated from the number of hops passed by a packet, the horizontal transmission distance is multiplied with the average distance between routers. This yields the number of routers passed. We use average numbers, as routers will not be spaced evenly if the size of processing elements varies. Furthermore, please note, that this model is accurate under zero load by construction. We verified this using simulations, as shown in Sec. VIII-A (Figs. 19 and 20).

Definition 6 (Horizontal Router Throughput): Let π be a packet with $\pi = (p_1, p_2, l)$ and $\xi \in [\ell]$ a layer. The delay for processing head flits per router is $\delta(\xi)$. The router is pipelined with $\chi(\xi) \in [0, \delta(\xi)]$ steps. The clock delay of routers is $clk(\xi)$, measured in [s]. The horizontal router throughput, measured in [flits/s], is given by the number of flits that a router can pass in a period of time:

$$\hat{\Delta}_H(\pi, \xi) = \frac{l}{(l + \delta(\xi) - \chi(\xi))\, clk(\xi)} \tag{6}$$

B. VERTICAL COMMUNICATION

Only vertical communication is effected by varying clock speeds. We model a non-purely synchronous communication, which allows to model different router and link architectures, such as the mesosynchronous proposed in Sec. VII.

Definition 7 (Vertical Packet Head Latency Under Zero Load): Let π be a packet with $\pi = (p_1, p_2, l)$ and ξ and $\lambda \in [\ell]$ layers with $p_{1_z} = \xi$ and $p_{2_z} = \lambda$. Without loss of generality, assume that $\xi \leq \lambda$. The clock delay of routers is $clk(i)$ for all layers $i \in [\ell]$, measured in [s]. The vertical packet head latency under zero load (downwards), measured in [s], is given by the delay each router adds during head flit processing

$$\Delta_V^{\downarrow}(\pi, \xi, \lambda) = \sum_{i=\xi}^{\lambda} \delta(i) clk(i). \tag{7}$$

The vertical packet head latency under zero load (upwards), measured in [s], is given by the delay each router adds during head flit processing plus a clock cycle for synchronization. This occurs only once during the path of the packet, since only two types of technology nodes are combined. The slower clock frequency dominates. This is illustrated in Fig. 4 following the dashed thick arrow for the transmission of the head flit. In the Figure, the example uses routers in two layers, clocked at a frequency of 1 and of 1/2. All routers have $\delta = 0$ and pipelining $\chi = 0$.

$$\Delta_V^{\uparrow}(\pi, \xi, \lambda) = \sum_{i=\lambda}^{\xi} \delta(i) clk(i) + clk(\xi). \tag{8}$$

Please note, that this model, again, is accurate under zero load by construction, (cf. Sec. VIII-A).

Definition 8 (Vertical Router Throughput): Let π be a packet with $\pi = (p_1, p_2, l)$ and ξ and $\lambda \in [\ell]$ layers with $p_{1_z} = \xi$ and $p_{2_z} = \lambda$. Without loss of generality, assume that $\xi \leq \lambda$. Routers are pipelined with $\chi(i) \in [0, \delta(\xi)]$ steps in each layer $i \in [\ell]$. The clock delay of routers is $clk(i)$, measured in [s]. The vertical router throughput, measured in [flits/s], is given by the slowest router:

$$\hat{\Delta}_V(\pi, \xi, \lambda) = \min_{i \in [\xi, \ldots, \lambda]} \left\{ \hat{\Delta}(\pi, i) \right\} \tag{9}$$

Long delays for processing a head flit are not relevant in the case of pipelining. Fig. 4 demonstrates that the slowest clock dominates the throughput of the transmission for asynchronous chips using an exemplary two-layer chip with routers clocked at a frequency of 1 and of 1/2.

V. INTEGRATION ISSUES FOR HETEROGENEOUS 3D INTERCONNECTS

Limitations of heterogeneous 3D interconnects are a result of different transmission speeds in varying technologies as found in Definitions 2 and 3. This can be overcome by routing

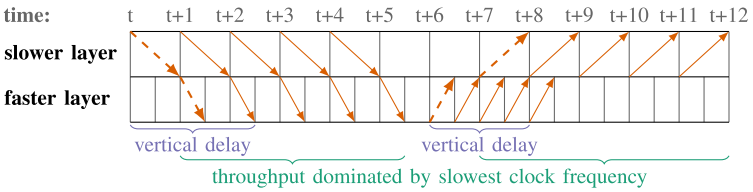

FIGURE 4. Vertical communication is dominated by the slowest clock frequency.

algorithms to achieve latency reductions. Routers are not purely synchronous, which will influence the throughput of routers along the packet's path, if it traverses multiple layers. This can be overcome by router architectures with increased throughput. Only simultaneous consideration of latency and throughput enables development of efficient interconnects, which impressively demonstrates the essential need for a co-design of routing strategies and router architectures in heterogeneous 3D SoCs.

A. TACKLING LATENCY LIMITATIONS VIA NOVEL ROUTING STRATEGIES

This publication answers whether communication via certain layers in heterogeneous 3D SoCs is faster, depending on the technology constraints, which can be exploited by routing algorithms. Intuitively, the first guess is that more advanced technology nodes are faster: routers certainly have a faster clock frequency. But there is a powerful adversary: the size of individual routers shrinks with better technology nodes. Thus more routers are located along the path of packet which add delay. To give a comprehensive answer, the proposed area and the proposed timing model must be considered simultaneously. Using Eq. 4 and Eq. 5, and derivation, yield the propagation speed of a packet under zero load.

Definition 9 (Propagation Speed): Let $\xi \in [\ell]$ be a layer. The propagation speed in layer ξ is

$$\omega(\xi) = \frac{\rho(\xi)}{\delta(\xi) clk(\xi)} \tag{10}$$

measured in [m/s]. It can be obtained by considering any packet π with $\pi = (p_1, p_2, l)$ with distance $s(\pi)$. The speed is distance per time, i.e. $\omega(\xi) = \frac{s(\pi)}{\Delta_H(\pi,\xi)}$.

The propagation speed ω is shown in Fig. 5 for commercial 130nm mixed-signal and 90nm – 28nm digital technology, using the synthesis results for our NoC router with a head flit delay of $\delta = 3$ and a 2×2 NoC in the mixed-signal layer. This yields a horizontal transmission speed improvement of between 2.7× and 4.3×, comparing mixed-signal and digital technologies. We see that the more powerful adversary which dominates is the clock scaling, whose influence is stronger than the effect of area scaling. To show the effect for other technology nodes, we fit the proposed models to the data (see Sec. VIII-A). The results are shown in Fig. 5, as well. The models can be used to predict the propagation speed for technology nodes below 28nm. This demonstrates

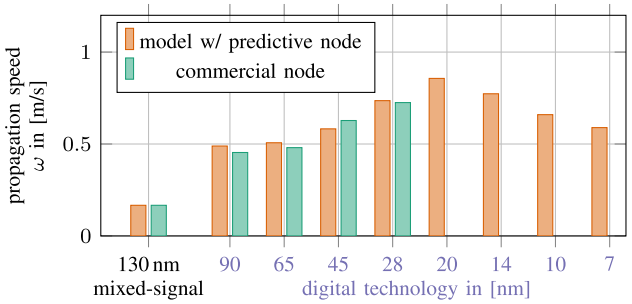

FIGURE 5. Propagation speed ω using a three-cycle router.

potentials of our approach for more modern technologies, but we do not use this for the further evaluation, since it is predictive. The performance speed improvement is between 5.1× and 3.3×. It is lower for more modern technologies due to limits posed by clock frequency scaling. Thus, clock frequency scaling remains dominant over area scaling, yet its advantages decline; Routing algorithms utilizing this are proposed in Sec. VI.

B. TACKLING THROUGHPUT LIMITATIONS VIA NOVEL ROUTER ARCHITECTURES

We consider the influence of heterogeneity on throughput. Let's consider, for sake of simplicity, only packets with length l. Then, according to Eq. 6, the throughput of horizontal communication is $\hat{\Delta}_H = \frac{1}{\text{clk}(\xi)}$: it is determined by the layer's clock frequency. If communication spans layers in another technology (i.e. with another clock frequency), Eq. 9 yields the vertical throughput:

$$\begin{aligned} \hat{\Delta}(\pi, \lambda) &= \min\{\hat{\Delta}_V(\pi, \xi, \lambda), \hat{\Delta}(\pi, \lambda)\} \\ &= \hat{\Delta}_V(\pi, \xi, \lambda) \leq \frac{1}{\text{clk}(\xi)} \end{aligned} \tag{11}$$

We have thereby shown that the throughput of packets which spans heterogeneous layers is determined by the slowest clock frequency: *the chain is only as strong as its weakest link.* This effect poses a universal limitation to routing in heterogeneous 3D SoCs: *communication may not span slower clocked layers if high throughput is required.* This issue cannot be circumvented by routing algorithms, since the only viable option is to avoid slower layers, which is impossible for a packet to and from this layer. This has two consequences: First, horizontal transmission in slower layers must be reduced to a minimum. Second, if a packet originates

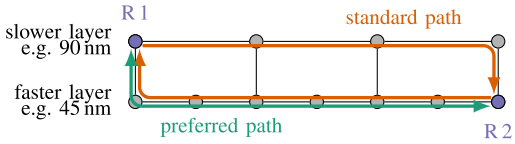

FIGURE 6. *"Stay in faster layers!"*: The green paths are faster than the orange paths.

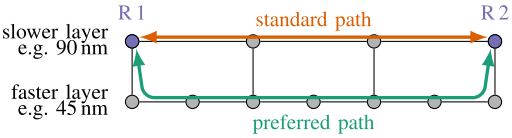

FIGURE 7. *"Go through faster layers!"*: The green path from R 1 to R 2 is longer yet faster than the orange path.

from a slow layer or is designated to a slow layer, the effects of their slow clock frequency must be minimized. This can only be achieved by novel router architectures; We propose an exemplary implementation in Sec. VII.

VI. TACKLING LATENCY VIA ROUTING STRATEGIES

In this section, routing strategies for heterogeneous 3D interconnects are developed. We start by abstracting the findings of our models into principles in Sec. VI-A. Next, we shorty introduce some technical preliminary considerations for our setting in Sec. VI-B. Finally, we can develop our routing strategies based on the principles in Secs. VI-C and VI-D. The validity of the routing strategies is explained in Sec. VI-E by proving deadlock and livelock freedom.

A. PRINCIPLES FOR ROUTING IN HETEROGENEOUS 3D INTERCONNECTS

The potentials as discussed in Sec. V reveal that transmission through different layers can yield a performance advantage which is unique to heterogeneous 3D interconnects. This can be exploited by the following two paradigms for routing strategies:

- *"Stay in faster layers!"*: Packets should stay as long as possible in layers which provide higher propagation speeds. An example is shown in Fig. 6. The sectional drawing of a two-layered chip is depicted. The layers are in MS and digital technology with $s_f = 4$. Usually, the data transmitted from routers R 1 to R 2 stay in the upper layer until reaching the router above R 2 (depicted in orange color). This path is slower than the way back via the lower layer in the more advanced technology node. Thus, it is favorable to route all packets via the preferred path, depicted in green.
- *"Go through faster layers!"* If the performance gain is large, packets can be routed via adjacent, faster layers since the path is faster. An example is shown in Fig. 7. A sectional drawing of a two-layered chip is depicted. The layers are in mixed-signal and digital technology with $s_f = 4$. The routers R 1 and R 2 are communicating. Usually, data is transmitted via the upper layer, which is slower than the lower layer. Therefore, it is favorable to route packets via the orange path.

We apply the two aforementioned paradigms to develop two exemplary routing algorithms. The proposed models provide relevant information on their potentials and to set parameters of the routing algorithms. Our proposed models allow to assess which routings are applicable and under which circumstances, since the models are generally valid, i.e. can be applied to any topology and set of technology parameters (beyond the proposed algorithms and the setting). Thus, we do not lose generality of models, yet demonstrate their expressiveness.

B. PRELIMINARY CONSIDERATIONS

1) SETTING

A heterogeneous 3D SoC with $\ell \in \mathbb{N}$ layers is used. Its layers are ordered by technology node, as in the vast majority of works on 3D SoCs, e.g. [42]. The most coarse-grained technology is at the top whilst the most fine-grained technology is bottom-most. Reordering the layers does not influence the models and principles and only requires minor changes to the proposed routing algorithms; hence, this does not lead to a loss of generality. But the order reduces the complexity of descriptions. Our approach is applicable to scenarios without ordered layers, with minor modifications.

Within the heterogeneous 3D SoC we implement a 3D NoC. Each layer has a grid with m_ξ rows and n_ξ columns, wherein $\xi \in [\ell]$ is the layer index. Routers are disposed in rows and columns. Neighboring routers are connected horizontally forming a m_ξ-n_ξ-mesh topology in layers, which is the most common NoC topology. No router has more than one link in the same direction, e.g. we do not model long range links. All routers, except those on the bottommost layer, have a (bidirectional) vertical link to the adjacent router in the next lower layer. This is possible thanks to the ordering of layers (cf. Fig. 8). The set of routers V is also the vertex set of the network digraph $T = (V, A)$.[2] The set of arcs A contains the directed links between routers.

2) ADDRESSES IN THE NETWORK

Locations of routers are given by a coordinate system with its origin in the SoC's top left corner, as shown in Fig. 9. Routers have both a physical location and a row and column number. The implementation of routing algorithms must be efficient, i.e. calculating with the physical locations is not realistic; using row, column, and layer numbers is. Rows and columns are based on the network digraph and not the physical locations: For example, pairs of neighbored routers in adjacent layers do not necessarily have the same physical x- and y- coordinate but the same column and row number. This is shown in Fig. 8. We do not depict this in all figures for sake of simplicity. If routers are depicted as stacked (cf. Fig. 6), we will intend a placement comparable to Fig. 8. We use

[2]In *Duato* [43] the network digraph is called *interconnection network*.

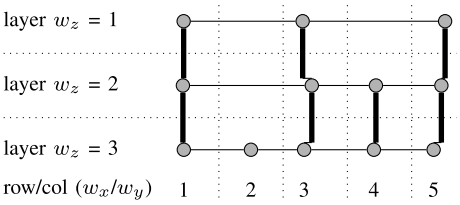

FIGURE 8. Logical order using redistribution.

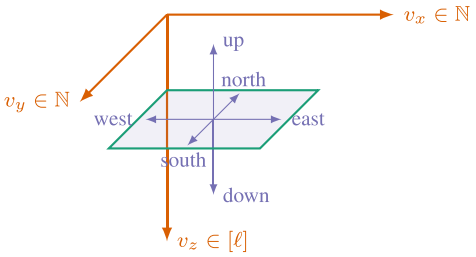

FIGURE 9. Cardinal directions in model coordinates W.

the notation $w = (w_x, w_y, w_z)$ for $w \in W = \mathbb{N}^3$, which determines row, column, and layer of each router, which is equivalent to the address. An injective function $m : W \to P$ converts addresses to locations of routers. Packets with source and destination address are given by $\tilde{D} = W \times W \times L$.

3) CARDINAL DIRECTIONS

We use the six cardinal directions $C := \{\text{north, east, south, west, up, down}\}$ to sort the arcs as shown in Fig. 9. We define functions which return the set of all links in one of these cardinal directions. These are given for all links $(v, w) \in A$:

$$\begin{aligned}
(v,w) \in \text{north}(A) &\Leftrightarrow v_x = w_x, v_y > w_y, v_z = w_z \\
(v,w) \in \text{east}(A) &\Leftrightarrow v_x < w_x, v_y = w_y, v_z = w_z \\
(v,w) \in \text{south}(A) &\Leftrightarrow v_x = w_x, v_y < w_y, v_z = w_z \\
(v,w) \in \text{west}(A) &\Leftrightarrow v_x > w_x, v_y = w_y, v_z = w_z \\
(v,w) \in \text{up}(A) &\Leftrightarrow v_z > w_z \\
(v,w) \in \text{down}(A) &\Leftrightarrow v_z < w_z
\end{aligned}$$

For example, north(A) contains all links pointing north. We further introduce functions that return neighbors of routers in a certain cardinal direction, if a link exists.[3] Routers at the edges of the network do not have links in that direction which is given by the value $\mathbb{0}$. We define for all $f \in C$:

$$\begin{aligned}
f : V &\to V \cup \{\mathbb{0}\} \\
v &\mapsto \begin{cases} w & \text{if } (v,w) \in f(A) \\ \mathbb{0} & \text{otherwise.} \end{cases}
\end{aligned}$$

C. APPLYING PRINCIPLE 1: Z⁺(XY)Z⁻ - ROUTING ALGORITHM

We apply principle 1, *"Stay in faster layers!"* and design a minimal and deterministic routing algorithm. Let $\tilde{\pi} = (v, w, l)$ be a packet. If the packet is not transmitted within a layer, i.e. $v_z \neq w_z$, the faster layer must be identified. Therefore, we apply Eq. 10 to calculate the average propagation speed at design time. This yields the following rules for transmission of packet π (in router with address v):

- If $\omega(v_z) < \omega(w_z)$, XYZ routing will be applied.
- If $\omega(v_z) > \omega(w_z)$, ZXY routing will be applied.
- If $\omega(v_z) = \omega(w_z)$, either will be selected at design time, depending on other network properties such as energy consumption of routers.

We call this routing algorithm $Z^+(XY)Z^-$.[4] Since layers are ordered by technology and hence by transmission speed, the implementation extends deterministic XYZ simply by reordering `if`-statements. Routers will only require additional flag storing information if faster layer is located below, above or is indeed this actual layer. The resulting routing is illustrated in Fig. 10.

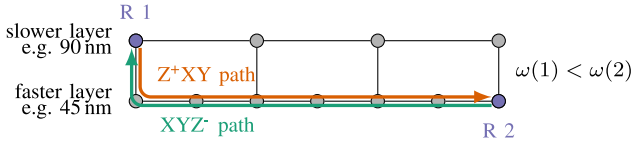

FIGURE 10. Z⁺(XY)Z⁻ routing: transmission through the lower layer.

Definition 1 (Routing Function R_1 for $Z^+(XY)Z^-$ Routing): Let $T = (V, A)$ be the topology digraph with the set of routers V and the set of links A. Further, $\mathcal{P}(A)$ is the power set of A. The routing function $R_1 : V \times V \to \mathcal{P}(A)$ is defined as:[5]

$$(v,d) \mapsto \begin{cases} \emptyset & \text{for } v = d \\ \{\text{north}(v)\} & \text{for } v_x = d_x, v_y > d_y, v_z \geq d_z \\ \{\text{east}(v)\} & \text{for } v_x < d_x, v_z \geq d_z \\ \{\text{south}(v)\} & \text{for } v_x = d_x, v_y < d_y, v_z \geq d_z \\ \{\text{west}(v)\} & \text{for } v_x > d_x, v_z \geq d_z \\ \{\text{up}(v)\} & \text{for } v_x = d_x, v_y = d_y, v_z > d_z \\ \{\text{down}(v)\} & \text{for } v_z < d_z. \end{cases}$$

D. APPLYING PRINCIPLE 2: ZXYZ - ROUTING ALGORITHM

We apply principle 2, *"Go through faster layers!"*. This requires to identify a quicker path for packets using detours. The identification of the best path depends on the position of source and destination, since there is an overhead when routing to the fastest layer for vertical transmission. We assess under which circumstances routing via an adjacent layer is advantageous. Let $\tilde{\pi} = (v, w, l)$ be a packet with source address v and destination address w. Let π be the corresponding packet after applying m to convert addresses to locations. The transmission time under zero load in the layer v_z is $\Delta_H(\pi, v_z)$ (Eq. 5). Let $\lambda \in [\ell]$ be another layer, through which the packet could potentially be transmitted.

[3] Note, that the above functions are only well defined, if no router has more than one link to the same direction.

[4] Minimality refers to the shortest path in the interconnection network. In terms of hop distance the proposed routing algorithm is not minimal. It is, however, if the links in the interconnection graph are weighted with their speed.

[5] Due to the setting all routers have a downwards vertical link (except those in the bottommost layer); thus $\{\mathbb{0}\}$ is impossible by construction (proved in Lemma 3).

The transmission time via layer λ is the transmission time for traversing vertical links, plus time within layer λ. Applying the model yields the condition under which routing via layer λ has a smaller latency:

$$\Delta_H(\pi,\xi) > \Delta_V^{\downarrow}(\pi,\xi,\lambda) + \Delta_H(\pi,\lambda) + \Delta_V^{\uparrow}(\pi,\xi,\lambda) - 2\delta(\lambda)\mathrm{clk}(\lambda) \tag{12}$$

We calculate a threshold distance $\phi(\xi,\lambda)$ that determines the minimum distance in layer ξ for which rerouting via layer λ is faster. Please note that we assume two layers in disparate technologies which are adjacent. It is not useful to use another than the uppermost digital layer to save vertical transmission time. Nonadjacent layers in mixed signal nodes have larger thresholds. Eq. 12, with $\phi := s(\pi)$ yields $\left(\frac{\phi\delta(\xi)}{\rho(\xi)} - \rho(\xi) - 1\right)\mathrm{clk}(\xi) = \left(\frac{\phi}{\rho(\lambda)} + 1\right)\delta(\lambda)\mathrm{clk}(\lambda)$, which is transformed to:

$$\phi(\xi,\lambda) = \begin{cases} \frac{(\delta(\xi)\mathrm{clk}(\xi)+\delta(\lambda)\mathrm{clk}(\lambda)+\mathrm{clk}(\xi))\,\rho(\xi)\rho(\lambda)}{\delta(\xi)\mathrm{clk}(\xi)\rho(\lambda)-\delta(\lambda)\mathrm{clk}(\lambda)\rho(\xi)} & \text{for } \xi < \lambda \\ \infty & \text{else} \end{cases} \tag{13}$$

Note, that ∞ can be replaced by any value larger the size of the chip. The two routing conditions are: (a) If a λ exists with $s(\pi) = s((m(v), m(w), l)) > \phi(v_z,\lambda)$, ZXY routing will be applied in direction of $arg\,min_{\lambda\in[\ell]}\,\phi(v_z,\lambda)$. (b) If $s(\pi) = s((m(v), m(w), l)) \leq \phi(v_z,\lambda)$ for all $\lambda \in [\ell]$, XYZ routing will be applied. There are two bottlenecks for run-time calculation: First, online selection of the best layer by evaluation of *arg min* is too expensive. A layer Λ must be selected at design time. From a practical standpoint, the uppermost digital layer is preferred because it offers high speed and low overhead for vertical transmission.[6] Second, addresses must be converted in locations. Therefore, we convert the location threshold distance ϕ into a hop distance by division through the average router distance in the digital layers:

$$\Phi(\xi,\Lambda) := \lceil \phi(\xi,\Lambda)/\rho(\ell) \rceil \tag{14}$$

It is required that ϕ is smaller than the outside measurements of the chip so that the routing can be applied. For a combination of a commercial 130nm mixed signal node with commercial 90 – 28nm digital nodes and a 4-4 NoC in the layer in mixed signal technology, ϕ is between 0.63 and 0.45 for a chip with edge length normalized to 1. Hence, packets traveling more than 2 or 3 hops in the layer in mixed signal node are routed via the adjacent layer.

To summarize, the routing algorithm has these conditions for a packet $\tilde{\pi} = (v, w, l)$ in router v:

- If $|v_x - w_x| + |v_y - w_y| \leq \Phi(\xi,\Lambda)$, XYZ routing will be applied.
- If $|v_x - w_x| + |v_y - w_y| > \Phi(\xi,\Lambda)$, the packet will be routed down.

[6]Without loss of generality, we set $\Lambda := \ell$ in proofs.

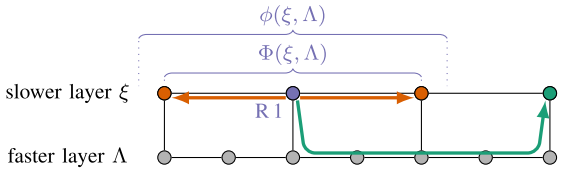

FIGURE 11. ZXYZ routing: A detour is faster for long distances.

We call this routing ZXYZ. It is illustrated in Fig. 11.

Definition 2 (Routing Function R_2 for ZXYZ): Let $T = (V, A)$ be the topology digraph with the set of routers V and the set of links A. Let Λ be a layer which is selected for rerouting at design time. Let $\Phi(\xi,\Lambda)$ be a threshold for rerouting according to Eq. 14. The routing function $R_2 : V \times V \to \mathcal{P}(A)$ is defined as:

$$(v,d) \mapsto \begin{cases} \emptyset & \text{for } v = d \\ \{\mathrm{down}(v)\} & \text{for } |v_x - d_x| + |v_y + d + y| > \Phi(v_z,\Lambda),\ v_z \geq d_z \\ \{\mathrm{down}(v)\} & \text{for } v_z < d_z. \\ \{\mathrm{north}(v)\} & \text{for } v_x = d_x,\ v_y > d_y,\ v_z \geq d_z,\ |v_y - d_y| \leq \Phi(v_z,\Lambda) \\ \{\mathrm{east}(v)\} & \text{for } v_x < d_x,\ v_z \geq d_z,\ |v_x - d_x| \leq \Phi(v_z,\Lambda) \\ \{\mathrm{south}(v)\} & \text{for } v_x = d_x,\ v_y < d_y,\ v_z \geq d_z,\ |v_x - d_x| \leq \Phi(v_z,\Lambda) \\ \{\mathrm{west}(v)\} & \text{for } v_x > d_x,\ v_z \geq d_z,\ |v_x - d_x| \leq \Phi(v_z,\Lambda) \\ \{\mathrm{up}(v)\} & \text{for } v_x = d_x,\ v_y = d_y,\ v_z > d_z \end{cases}$$

E. PROOF OF VALIDITY: DEADLOCK AND LIVELOCK FREEDOM

We prove that the routing algorithms are free of deadlocks and livelocks. We make use of *Duato's theorem* [43], according to which a routing is deadlock-free if the routing function is connected and the channel dependency graph is cycle free. We also use terms and definitions from [43] without further explanation, such as *routing function*, *adaptive*, *connected*, *direct dependency*, and *channel dependency graph*. If there is a direct dependency from a to b, we also say: >>*b is direct dependent on a.*<< Graph related terms like *path*, *closed walk*, or *cycle* are used as defined in [44].

We introduce the terms *possible turn* and *impossible turn* according to a routing function R. These terms denote, if the routing functions permits consecutive flow of packets in these directions.

Definition 3: A pair of cardinal directions $(f, g) \in C \times C$ is called a possible turn according to R, if there exist two consecutive arcs, (u, v) and $(v, w) \in A$, with: $(u, v) \in f(A)$, $(v, w) \in g(A)$ and there is a direct dependency from (u, v) to (v, w). A pair of cardinal directions that is not a possible turn is called an impossible turn according to R.

Lemma 1: If there is a cycle in the channel dependency graph (CDG), then we can also find a closed walk $(v_1, a_1, v_2, \ldots, v_k, a_k, v_1)$ (for $k \in \mathbb{N}$) in the topology digraph with

- a_{i+1} is direct dependent on a_i for all $i \in \{1, \ldots, k-1\}$,
- and a_1 is direct dependent on a_k.

Proof: Assume that there is a cycle $(\{a_1, \ldots, a_k\}, \{(a_1, a_2), \ldots, (a_{k-1}, a_k), (a_k, a_1)\})$ in the CDG. According to the definition of direct dependency, the destination node of a_i in the topology digraph is also the source node of a_{i+1} (for all $i \in \{1, \ldots, k\}$, and $a_{k+1} := a_1$). Let us call this node v_{i+1} (for all $i \in \{1, \ldots, k\}$). Then, $(v_{k+1}, a_1, v_2, \ldots, v_k, a_k, v_{k+1})$ is a closed walk in the topology digraph. □

F. $Z^+(XY)Z^-$: R_1 IS DEADLOCK-FREE

By looking at the definition of R_1, we can determine the impossible turns and the possible turns. Here, we assume that the numbers of rows, columns and layers m_ξ, n_ξ and ℓ are not too small. We assume $m_\xi, n_\xi, \ell \geq 2$ for all $\xi \in \{1, \ldots, \ell\}$ as a precaution. Table 1 shows which turns are possible.

Lemma 2: When R_1 gives a direction, then the necessary link always exists.

Proof: Places without links in some directions are: (a) At the outside faces of the 3D NoC cuboid links at edges of layers, upward links from the topmost layers, and downward links from the bottommost layer do not exist. (b) Some upward links do not exist between layers if one layer is in another technology than the other layer. a) By looking at the definition of R_1, one can check that every routing step brings the packet nearer to d. Hence, the nonexistent links on the outer border of the 3D-NoC are never taken by R_1. b) Not every router has an up-link. Every router, except those in the bottommost layer, has a down link by premise. Downward links in a router are upward links in the router below:. When router v has the same x- and y-coordinates as the destination router d and v is below d, v has an up-link. These are also the conditions for traveling up in R_1. □

Lemma 3: R_1 is connected.

Proof: Let s and d be any two vertices in V. R_1 returns a direction for every vertex except d (it returns $\emptyset$). The links in the chosen direction always exist (Lemma 2). If we apply the routing function step by step and proceed through the network in the returned directions, we will find a route. As shown in the proof of livelock-freedom, the route is not infinite (Theorem 3). Hence, it terminates. Termination can only happen at d, by definition. Hence, with the routing function R_1, we always find a path from s to d. □

Theorem 1: R_1 is deadlock-free.

Proof: R_1 is connected, because of Lemma 3. Assume, that the CDG of T and R_1 has a cycle. Lemma 1 proves that T has a cycle where each two consecutive arcs are direct dependent.

TABLE 1. Possible turns (f, g) in R_1 and R_2.

	g:	n.	e.	s.	w.	u.	d.
f:	n.	1	0	0	0	1	0
	e.	1	1	1	0	1	0
	s.	0	0	1	0	1	0
	w.	1	0	1	1	1	0
	u.	0	0	0	0	1	0
	d.	1	1	1	1	0	1

Case 1: All vertices of the cycle are in the same layer. We know by [45] that XY routing has a cycle free CDG due to impossible turns. Thus, Case 1 does not occur.

Case 2: The vertices of the cycle are in at least two different layers. Since the vertices are in different layers, there is at least one arc, which goes up. According to table 1, the only possible direction after >>up<< is >>up<< and the cycle could never be closed. Hence, Case 2 is also impossible.

We have shown by contradiction that the CDG is cycle-free and apply Duato's Theorem on R_1. □

G. ZXYZ: R_2 IS DEADLOCK-FREE

Again, we can determine the set of possible turns. It can be seen in Table 1.

Lemma 4: R_2 is connected.

Proof: Let s and d be any two vertices in V. We construct a path $(s = v_1, \ldots, v_k = d)$ with $v_i \in V$ for all $i \in [k]$, $k \in \mathbb{N}$ from s to d by using links $(c_1 = (v_1, v_2), \ldots, c_{k-1} = (v_{k-1}, v_k))$ with $c_i \in A$ for all $i \in [k-1]$, which are consecutively delivered by the routing function R_2.

Case 1 (The source is above the destination $s_z < d_z$): As in the proof of Lemma 3, the route starts with a sequence of downs until the destination layer is reached. Now the routing goes as explained in Case 2.

Case 2 (The source is below the destination or on the same layer $s_z \geq d_z$): The next links depend on the logical value of $||s - d|| \geq \Phi(s_z)$.

Case 2.1 ($||s-d|| \geq \Phi(s_z)$): If the condition is true, the next link will be down. The value of $||s - d||$ is the same as $||v_2 - d||$. The value of $\Phi(z)$ is the same for all $z < \Lambda$. Hence, layer Λ will be reached via a sequence of downs. The rest of the path is constructed as in Case 2.2.

Case 2.2 ($||s - d|| < \Phi(s_z)$): Here, R_2 is identical to R_1. Connectivity is proven in Lemma 3. □

Theorem 2: R_2 is deadlock-free.

Proof: The proof is analog to the proof of Theorem 1. R_2 is connected because of Lemma 4. We assume that the CDG of T and R_2 has a cycle. Then T has a cycle, in which each two consecutive arcs are direct dependent, according to Lemma 1.

Case 1: All vertices of the cycle are in the same layer. Case does not occur, cp. Theorem 1, *Case 1*.

Case 2: The vertices of the cycle are in at least two different layers. There is at least one arc going up. According to table 1,

the only possible direction after >>up<< is >>up<<. Thus, the cycle can not be closed. Hence, case 2 is impossible.

We have shown by contradiction that the CDG is cycle-free. We apply Duato's Theorem on R_2. □

H. LIVELOCK FREEDOM

Palesi and Daneshtalab [46] define that "livelock is a condition where a packet keeps circulating within the network without ever reaching its destination". Hence the following definition.

Definition 4 (Livelock-Free): A routing algorithm is livelock-free, if every packet has no other choice, but to reach its destination after a finite number of hops.

Remark. A routing algorithm consists of a routing function and a selection. R_1 and R_2 are examples for routing functions. If an adaptive routing function returns more than one link, the selection chooses one. The property *livelock-free* belongs to the routing algorithm. Nevertheless, we call a routing function *livelock-free* if, independent of the selection, every routing algorithm with this routing function is livelock-free.

Theorem 3: R_1 and R_2 are livelock-free.

Proof: Assume there were two vertices s and d with the property that the routing R_1 makes infinite steps and never reaches d starting from s (the same arguments hold for R_2). Under this assumption, at least one cardinal direction must be traveled infinite times. We do a case-by-case analysis in which we assume that this applies to the different cardinal directions. We thereby show that it works for none of them. This contradicts the assumption that there could be a livelock.

Case 1: >>up<< *is traveled infinite times.* By the definition of R_1 (Definition 1), *up* is only used if $v_x = d_x$ and $v_y = d_y$ and $v_z > d_z$, with v being the current vertex. Traveling *up* one layer will remain $v_x = d_x$ and $v_y = d_y$ and results either in $v_z = d_z$ or $v_z > d_z$. The only possible direction after >>up<< is >>up<<. Since there are only $\ell < \infty$ layers, d will be reached after finite steps. Thus, Case 1 can not occur.

Case 2: >>down<< *is traveled infinite times.* Since *up* can not be traveled infinite times (Case 1), *down* can not either. It is limited by the layers count, ℓ, plus the number of times *up* could be traveled.

Case 3: >>east<< *and* >>west<< *are traveled infinite times.* Similar to Case 2, infinite steps to *west* imply infinite steps to *east* and vice versa. From the definition of R_1, we know:

- *east* and *west* are the only directions, which affect the x-value of v.
- A step to *east* is only done if $v_x < d_x$
- A step to *west* is only done if $v_x > d_x$
- A step to *west*õr *east* is only done if $v_z \geq d_z$.

We never step on a router with $v_x = d_x$. If we reached a router with $v_x = d_x$, *up*- or *down*-routing would be done and the destination would be reached. Steps to *east*õr *west* are only done in the destination layer or below. In these layers, each row has a router at position d_x. Routing from west to east and back without using one of these routers is impossible.

Case 4: >>north<< *and* >>south<< *are traveled infinite times.* This case is analog to Case 3.

None of the cases occur. Thus, the assumption is wrong. R_1 is livelock-free.

The same arguments hold for R_2 (defined in Definition 2). R_2 is livelock-free.

Remark: The proof relies on our special setting. It requires that for u and v with down(u) $= v$ it holds: up(v) $= u$, $u_x = v_x$, and $u_y = v_y$. It also requires the mesh topology in layers. □

VII. TACKLING THROUGHPUT VIA ROUTER ARCHITECTURES

We have shown a fundamental limitation in heterogeneous routing paths using standard techniques in Sec. V-B: Throughput is limited by the slowest clock along a packet's path, or in other words, *the chain is as strong as its weakest link*. This is not an issue for 2D or homogeneous 3D systems, since the deviation of clocks is rather small there. In heterogeneous 3D SoCs, in contrast, this poses a severe limitation, since clocks potentially deviate by a large factor. This limitation, previously unexplored, is revealed by this paper. To solve this issue in combination with the proposed routing strategies, we propose to use a novel router mircoarchitecture. Thereby, we assume an integer relation between the clock frequencies c_f with a constant phase shift. Our architecture exploits the observation that optimized routing algorithms must minimize horizontal transmission in slower layers. With our proposed routing strategies, horizontal transmissions are always conducted in the fastest layer along the path. Thus, for heterogeneous packet-paths, packets are directly routed from local ports of a router to the port in direction of the faster layer (down). In the opposite direction, from downwards, packets can only be routed to the upward port or the local port for ejection. The architecture enables a small part of the router in the slower layers, comprising the local and vertical ports, to communicate multiple flits in parallel in order to provide the same throughput between the local and the vertical ports as faster routers from digital layers. Thereby, heterogeneous packet-paths are traversed with the throughput the standard router in the fastest technology provides. We refer to our new architecture as *high vertical-throughput router*.

A. HIGH VERTICAL-THROUGHPUT ROUTER DESIGN

As previously outlined, the router architecture in the slower layers has to be modified, using parallelism, to obtain a higher throughput between the local and the vertical ports. In the fast layers, only the vertical links towards the slower layers need to be modified (see below). Our new router architecture exploits that processing elements, connected to the local ports, are able to provide multiple parallel flits, since packet transmission is initialized for full packets. A conventional input buffered 3D router design, with link width of N, is modified as shown orange, in Fig. 12.

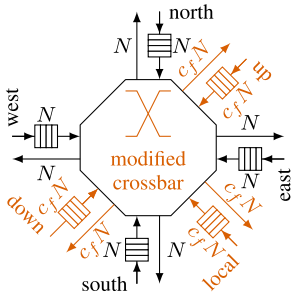

FIGURE 12. Modified router architecture with support for higher vertical throughput. The link width is N, and c_f the clock scaling factor of the current layer compared to the fastest layer.

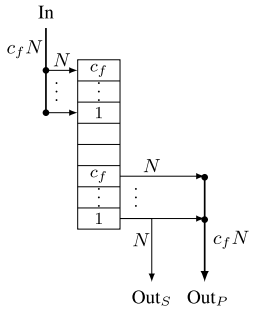

FIGURE 13. Modified input buffer. c_f flits can be read and written at once.

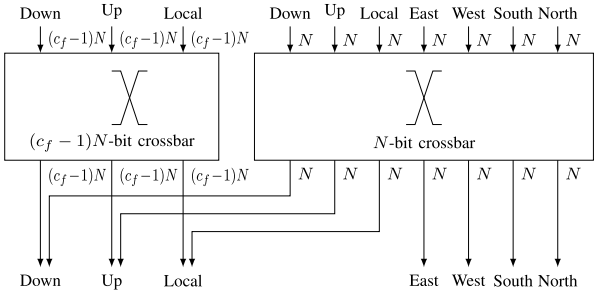

FIGURE 14. Modified crossbar which allows to route c_f flits between the local and the vertical ports.

The input-buffers (see Fig. 13) of the vertical and local ports can read up to c_f flits of N bits simultaneously. A single or c_f flits are inputed to the crossbar, which increases the bit width of the connection by factor c_f. The crossbar is also modified (see Fig. 14). Firstly, due to the proposed routing strategies, some turns (e.g. down to north, east, west or south) cannot occur. Secondly, the crossbar has to be extended to route c_f flits between local and vertical ports. In paths which do not include the fastest layer, horizontal routes via a slower layer cannot be avoided (still the fastest among all included ones is chosen). In this scenario, routes of single flits from the horizontal ports towards the up or local output ports occur. All remaining $(c_f - 1)N$ lines of the crossbar output are zero and only one flit can be written to the local port, or the input port of the overlying router, per cycle.

However, in the most common 3D NoC scenario with only one slower (mixed-signal) layer located at the top, the complexity of the proposed router architecture is reduced

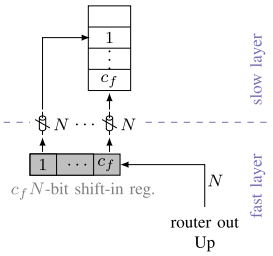

FIGURE 15. High-throughput connection from a faster layer to a slower layer employing a large MIV array and a shift register.

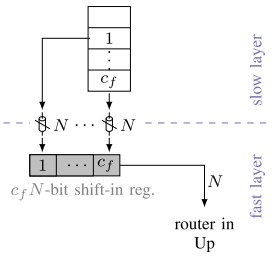

FIGURE 16. High-throughput connection from a slower layer to a faster layer employing a large MIV array and a shift register.

drastically for two reasons. Firstly, the modified routers at the top have no up port. This results in only tree ports, local, up and down, requiring a high-throughput connection. Thereby, the $(c_f - 1)N$-bit crossbar shown in Fig. 14 is added to the design; it has only three input and output ports. Thus, the local input port is directly connected to the downwards output port and vice versa, which does not incur any hardware cost. Furthermore, only two input buffers (local and down) need to be modified, which again reduces the hardware complexity. Secondly, all heterogeneous packets path will include a fast layer, thus routes of single flits from/to the downwards input ports will not occur. This again reduces the complexity of the input buffer as it only needs to send and/or receive c_f parallel flits and never single flits.

B. HIGH VERTICAL-THROUGHPUT LINKS

The vertical links must support the higher throughput of the modified routers. c_f flits are transmitted in parallel employing a large MIV array. (A large TSV array can also be implemented in case of if non-monolithic 3D integration.) On the way from a slower to a faster layer, data is transmitted in parallel with the slower clock frequency via the MIV array. The modified input buffer in the faster technology fetches the c_f flits in parallel with a rate equal to the clock speed of the slower layer. If data are transmitted to a slower layer, the data is first parallelized in the faster layer using a shift register. The full content of the $c_f N$-bit shift register is transmitted via the wide MIV array to the slower layer, where the flits are fetched in parallel by the modified input buffer. This is shown in Fig. 15. The inverse path from the slower layer to the faster layer is shown in Fig. 16. The architecture is analogous; Flits are transmitted in parallel from the slower layer and serialized using a shift register in the faster layer.

VIII. RESULTS

This section consists of four parts: First, we discuss the accuracy of our models for a set of commercial mixed-signal and digital technology nodes in Sec. VIII-A. Second, we show the impact of latency of our routing algorithms for 130nm commercial mixed-signal and 90nm – 28nm commercial digital nodes in Sec. VIII-B. Third, we focus on our router architectures by analyzing throughput improvements in Sec. VIII-C. Forth, we conclude the co-design of routing strategies and algorithms by considering the implementation costs and power improvements in Sec. VIII-D. Finally, we show the practical applicability of our proposed solution for heterogeneous 3D interconnects by means of a 3D VSoC case study in Sec. VIII-E using a heterogeneous combination of 30nm mixed-signal and 15nm digital technology nodes.

A. MODEL ACCURACY

First, we present results on the model accuracy of our area and timing model. Second, we give simulation results that support our claim of accurately modeling communication under zero load.

We fit the area and timing model to the synthesis results of a 3D NoC router with two virtual channels, four flit deep buffers per channel, credit based flow control, wormhole switching, decentralized arbiters and deterministic XYZ-routing using Synopsys design compiler for commercial 130nm mixed-signal technology and commercial 28 – 90nm digital technology. We use both general purpose (GP) and ultra low voltage (ULV) mixed-signal technology to exemplify potential differences. The synthesis results are used to evaluate the accuracy of the model fit.

The synthesis results and the fitted models for the *area scaling factor* are shown in Fig. 17. Curve fitting is conducted with Mathematica 10. The example yields a non-ideality factor $\alpha = 3462.7$ and an offset of $\hat{\alpha} = 29.8$ for 130nm GP technology with a root mean square error (RMSE) of 0.1286. ULV technology yields $\alpha = 13.2$ and an offset of $\hat{\alpha} = 0.124$ with a RMSE of 0.1414.

The synthesis results and the fitted models for the *clock scaling factor* with a predicted maximum achievable clock frequency of 5.0GHz are shown in Fig. 18. (Smaller commercial technology nodes below 28nm are not available, thus we set β instead of fitting it to the model.) The fitting is conducted with Mathematica 10. The results for GP nodes are $\beta = 32.85$, $\hat{\beta} = 7.88$, $\tilde{\beta} = 0.76$, and $\bar{\beta} = 1.26$ with a RMSE of 0.30. For ULV nodes, the model yields the parameters $\beta = 77.45$, $\hat{\beta} = 2.48$, $\tilde{\beta} = 0.76$ and $\bar{\beta} = 2.77$, with a RMSE of 0.71.

We claimed that our models for head flit latency are accurate under zero load by construction. To validate this, we use simulations for the latency enhancement of packets traversing the network for our two proposed routing strategies. These are shown in Figs. 19 and 20. We report the latency enhancement both from model and simulations. One can see that the results are matching and that our models, indeed, are accurate under zero load.

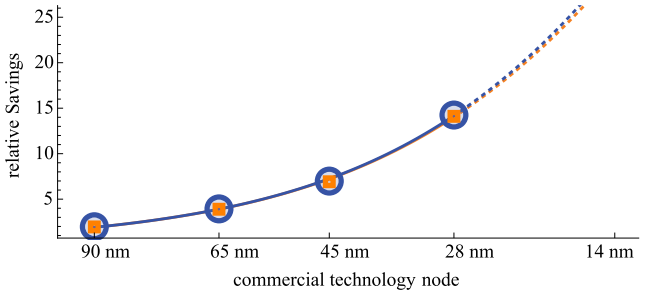

FIGURE 17. Area model accuracy using exemplary fit (orange – ULV, blue – GP).

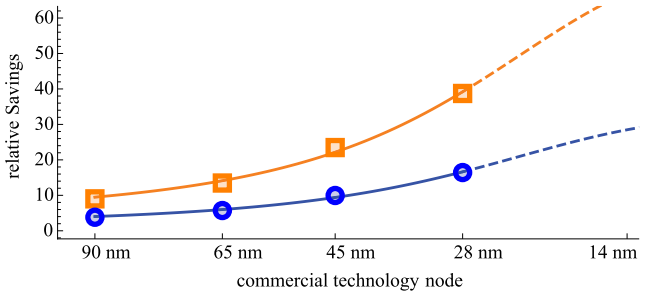

FIGURE 18. Timing model accuracy using exemplary fit (predictive maximum achievable clock frequency of 5GHz; orange – ULV, blue – GP).

B. LATENCY OF ROUTING ALGORITHMS

1) LATENCY OF $Z^+(XY)Z^-$

Packets from any node in the mixed-signal layers to any node in the digital layers profit from $Z^+(XY)Z^-$. We compare their latency under zero load to conventional XYZ. As an exemplary use case, we use a 3D SoC, which consists of two layers: One in a commercial mixed-signal technology implementing a 4×4 NoC and one in 90nm – 28nm commercial digital node implementing a NoC with more nodes according to the area model (Eq. 2) on basis of synthesis results. The achieved speedup is calculated using both a cycle-accurate NoC simulator with 16 flit deep buffer, wormhole routers and four VCs [47] and Δ_H from Eq. 5. The results are shown in Fig. 19 for all available hop distances in the layer in mixed-signal technology. Simulation and model results are identical; the model is accurate under zero load. The latency speedup is between $1.5\times$ and $6.5\times$. It is larger if a more advanced digital node is used, which is consistent with the expectations from Sec. V. Note that this speedup is achieved without any implementation costs.

2) LATENCY OF ZXYZ

Packets from any node in the mixed-signal layers to any node in the mixed-signal layers profit from ZXYZ. Again, we compare their latency under zero load to conventional XYZ. As an exemplary use case, we use a the same 3D SoC as before with two layers. The achieved speedup is calculated using both a cycle-accurate NoC simulator with 16 flit deep buffer, wormhole routers and four VCs and Δ_H and Δ_V from Eqs. 5, 7 and 8. The results are shown in Fig. 20 for all

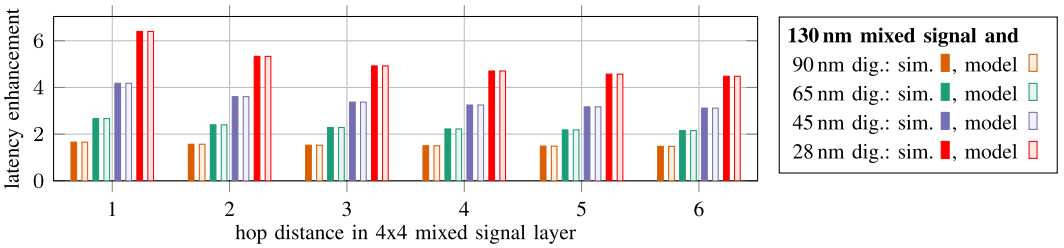

FIGURE 19. Latency enhancement of $Z^{+}(XY)Z^{-}$ to conventional XYZ.

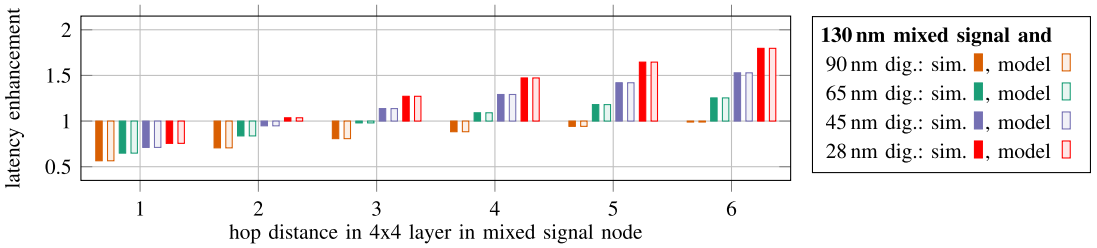

FIGURE 20. Latency enhancement of ZXYZ to conventional XYZ.

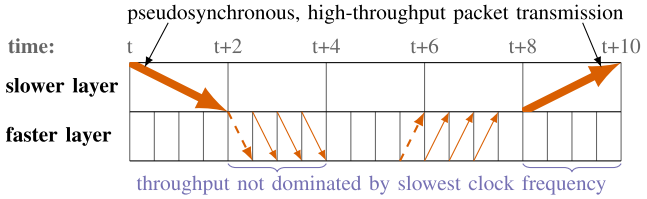

FIGURE 21. Throughput of high-vertical-throughput router architecture.

available hop distances in the layer in mixed-signal technology. The latency speedup is between 0.54× and 1.79×. It is noteworthy that any speedup is achieved with negligible implementation costs.

C. THROUGHPUT OF HIGH VERTICAL-THROUGHPUT ROUTER

Using the novel high vertical-throughput router architecture, the throughput of packets can be increased if the slower layer is contained in the path. In fact, the throughput will be as high as in the faster layer, if area for links and routers is expendable. This is shown in Fig. 21. For a transition from a slower to a faster layer (shown on left-hand side), the packet throughput is not determined by the slower clock frequency because the packet can be completely transmitted once it is available. For the opposite direction (right-hand side), the throughput is also not determined by the slower clock, since the complete packet becomes available at the faster router.

D. AREA AND POWER OF PROPOSED ROUTER ARCHITECTURE AND ROUTING ALGORITHMS

We synthesize the baseline router using conventional XYZ routing and the proposed *high vertical-throughput router* using $Z^{+}(XY)Z^{-}$/ZXYZ routing in a commercial 45nm ULV mixed-signal technology (We only synthesize for mixed-signal since the routers in the digital faster layer do not have a modified crossbar). The same crossbar optimizations are applied for both conventional and vertical-high throughput architectures. We assume a 4×3×3 NoC with one digital layer. The flit width is 32b, the input buffer depth is eight, the flow-control is credit-based and four virtual channels are only used in the digital layer. Both architectures, the proposed high vertical-throughput router as well as the baseline baseline, can run with a maximum frequency of 500MHz. Area and power results are shown in Tab. 2 and elaborate as follows:

TABLE 2. PPA comparison of proposed routing algorithms and high-throughput routers to conventional router.

Performance		Area	Power
throughput increase	average *latency* speedup of flits	total *area* increase	dynamic *power* savings
2×	2.26×	2.1 %	41.4 %

The *area overhead* of the proposed routing algorithms is negligible. In fact $Z^{+}(XY)Z^{-}$ routing has -1.32% overhead compared to conventional XYZ routing. For ZXYZ, the area is only increased by three gate equivalents, which affects the whole router area by less than -2.38%. The area of the crossbar and the input buffers depends on the clock frequency of the digital routers. To bridge to a clock frequency of 1GHz in the faster layer (c_f=2), the total area required for the routers is increased by 2.1%. If the routers in the fast layer are clocked at 2GHz (c_f=4), the total area increases by 10.6%.

Dynamic power savings are possible. We simulated the aforementioned NoC with 1M clock cycles, injecting uniform random traffic at 4% injection rate. The digital layers is implemented in 15nm digital technology and the mixed signal layer in 45nm ULV node. For a clock difference

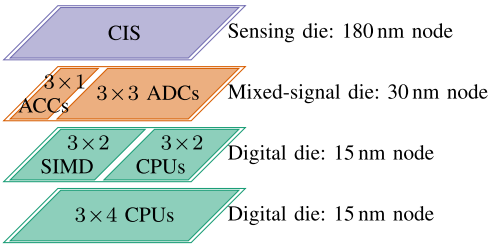

FIGURE 22. 3D VSoC case study based on [4].

of $c_f = 2$, the proposed routing algorithms saved 41.1% dynamic power in comparison to conventional XYZ-routing; For a clock difference of $c_f = 4$, the proposed routing algorithms saved 30.3% dynamic power.

E. CASE STUDY

We analyze our approach for a 3D VSoC based on [4] with four layers as shown in Figure 22: The first layer is a sensing die, implementing a 180nm CIS (CMOS Imaging Sensor). The second layer implements nine analog digital converters (ADCs) and three analog accelerators [17] in 90nm mixed-signal node. The third layer implements 6 processors and 6 SIMD (single instruction multiple data) acceleration units in 15nm digital node. In the fourth layer there are 12 processor cores in 30nm digital node. The first and second layer are connected via point-to-point links. The second, third and fourth layer are connected via a 3D NoC with 32b wide links, 8 flit deep buffers and 4 VCs. Packets are 32 flits long with one flit header. Routers in the digital layer are clocked at 1GHz and in the mixed-signal layer at 0.5GHz.

The 3D VSoC implements an image processing pipeline for face recognition. The image sensor records at 720p. The ADCs send the digital raw image to the processors in the third layer, which apply Bayer filter. Then, the SIMD units reduce the resolution by a factor of 4 to increase feature extraction speed. The result is transmitted to the analog accelerators in the second layer, which extract features using Viola-Jones algorithm [48]. The resulting region of interest is transmitted to the fourth layer, in which the processors execute Shi and Tomasi algorithm [49] to find features to track and Kande-Lucas-Tomasi algorithm [50] tracks them. Work is split up equally among the available resources in each step.

We simulate the VSoC's NoC using the described application traffic. Thereby, we compare $Z^+(XY)Z^-$ and ZXYZ with conventional XYZ routing. We simulate 3M clock cycles in the digital layers and 1.5M in the mixed-signal layer. We measure the average flit latency as 145.91ns for conventional routing and as 64.46ns for the proposed routing. This equates to a speedup of 2.26×. Using the models, we calculate a theoretical speedup of 2.28× under zero load. Average delay for whole packets is reduced from 229.23ns to 123.07ns, which is a speedup of 1.86×.

IX. DISCUSSION

First, we discuss the model accuracy as those are the basis for the subsequent evaluation of the routing algorithms. The aim of the models is to estimate the impact of heterogeneity on NoCs. Figs. 17 and 18 demonstrate a very good fit of the models for available nodes to academia. The area model has small RSMEs, which is a result of the model's physical foundation. It was not beneficial to add a linear term to this model; this increases the RMSEs. The timing model is empirical and thus the fit is overall less accurate than the area model fit, shown by higher RMSEs. The model converges to the target maximum clock frequency, as desired. If more modern technology nodes were available, either a better model with a physical foundation could be found or the fit of our model could be improved. Nonetheless, the model serves its purpose here: both the timing and the area model provide sufficient accuracy to assess the influence of heterogeneous integration on routing, as we further quantify. Therefore, we apply the fitted data to calculate the propagation speed ω for a predictive technology. This is shown in Fig. 5. Comparing predictive technology calculated with the models to the synthesis results for 130nm commercial mixed-signal and 90nm – 28nm commercial digital technologies yields an accuracy of between 1.4% and 7.8%. This supports that the proposed models are valid. We also propose models for latency and throughput. That they are accurate is given by construction and validated using simulations. The results are shown in Figs. 19 and 20. The results for latency will be identical, regardless if obtained from simulations or from the proposed model. Therefore, the communication models are precise under zero load. There is no need to model the behavior under load for the purpose of this paper. Of course, the models will not be valid if further traffic is injected and the assumption of zero load is violated. However, our model can also be extended to cover dynamic effects by applying a queueing model [25]. This is not required here because the unique effects of heterogeneity have already been revealed under zero load. Load is applied in our case study and our routing show a latency enhancement, as well. In fact, we see a speedup of 2.26× in simulations under load, while our models predict a speed-up of 2.28×. This shows that our models are accurate enough to find useful routing algorithms under real conditions, even though they only account for zero load within our case study. Thus, by means of our model, we are able to conduct powerful routing strategies and architectures for heterogeneous 3D interconnect.

Second, the exemplary implementations of routing algorithms and router architectures are evaluated. The aim of the implementations is to mitigate the negative effects of heterogeneity (worse latency and throughput), with as few area costs as possible. The largest limitations of heterogeneity emerge if the difference between mixed-signal and purely digital technology are large; therefore we focus on a chip using 130nm commercial mixed-signal technology and 28nm commercial digital technology. The results can also be

applied to any other combination of technologies with similar relative technology scaling factor Ξ. The proposed routing algorithms $Z^+(XY)Z^-$ and ZXYZ provide up to 6.5× latency reductions for packets from routers in the mixed-signal nodes to routers in the digital layer and up to 1.79× latency reductions for packets within the layer in the mixed-signal node in comparison to dimension order routing. This is shown in Figs. 19 and 20. For ZXYZ, there is a performance penalty for distances below Φ (Eq. 14) of up to 45%, as expected (see Fig. 20, left-hand side). The threshold distance shrinks for more advanced technology nodes, which is also expected. The conventional XYZ outperforms ZXYZ for low technology differences for all distances.

We compare the router for a practical scenario with a clock difference of 2 between layers in different nodes to show advantages of our approach. The results are summarized in Tab. 2. For a real-world based benchmark, we simulate a face recognition image processing pipeline on a 3D VSoC based on [4] with 45nm mixed-signal technology and 15nm digital technology. The proposed vertical high-throughput router offers 2.26× better latency and an increased throughput of up to 2×, in simulations, at 2.1% area increase comparing to a standard router for conventional XYZ routing. If a larger throughput increase is desired, additional area costs must be expended. While the area is increased, dynamic power is saved: We showed 41.4% dynamic power in simulations. The performance speedups and power savings demonstrates the impressive benefit of the proposed approach for typical applications of heterogeneous 3D SoCs.

To summarize, $Z^+(XY)Z^-$ and ZXYZ, in combination with the novel router architectures, have small area overhead and better performance than state-of-the-art both in theoretical and practical evaluations. Therefore, limitations of heterogeneity on routing in 3D NoCs are mitigated. Only by an integrated design of routing strategies and architectures, we are able to design an efficient and powerful heterogeneous 3D interconnect.

X. CONCLUSION

Heterogeneous 3D SoCs need to combine disparate technologies, e.g. mixed-signal and purely digital technologies; However, the impact of heterogeneity on interconnection networks was previously not considered. We show that varying throughput and latency of NoCs in layers in disparate technologies drastically degrades network performance. To prove this, models for area and timing of routers, and for latency and throughput under zero load have been proposed. The models are well-founded and express the relevant effects of heterogeneity on routing; the model accuracy is high and shows an error of 1.4%-7.8% for an exemplary technology scenario. Based on the model's findings, we develop principles for routing in heterogeneous 3D SoCs. We show their practical applicability by proposing two new exemplary routing algorithms. These reduce the network latency for packets between nodes in mixed-signal and purely digital technologies and between nodes in a mixed-signal layer by utilization of faster transmission speeds in digital layers. For an exemplary SoC, with layers in commercial 28nm digital and commercial 130nm mixed-signal technology, we achieve a latency reduction of up to 6.5× at negligible area overhead in comparison to conventional dimension ordered routing. We further propose a novel vertical high-throughput router architecture and a vertical link design to overcome the throughput limitations, which increase throughput by up to 2× at 6% reduced router area costs for the same exemplary set of technologies. Within simulation of a case study for a 3D VSoC using 30nm mixed-signal and 15nm digital technologies implementing a face recognition algorithm, we could validate our theoretical findings with a speedup of 1.86× to 2.26× for average latency and 2× for throughput. We also showed 41.4% reduced dynamic power in simulations using uniform random traffic. Summing up, the proposed co-design of routing algorithms and router architectures mitigate limitations of NoCs in heterogeneous 3D SoC. It allows much better performance and dynamic power consumption at small to negligible area overhead.

ACKNOWLEDGMENT

This work was funded by the German Research Foundation (DFG) Projects PI 447/8 and GA 763/7.

REFERENCES

[1] X. Dong and Y. Xie, "System-level cost analysis and design exploration for three-dimensional integrated circuits (3D ICs)," in *Proc. Asia South Pacific Design Autom. Conf.*, Jan. 2009, pp. 234–241.
[2] V. F. Pavlidis and E. G. Friedman, *Three-Dimensional Integrated Circuit Design*, 2nd ed. Amsterdam, The Netherlands: Elsevier, 2010.
[3] R. Chaware, G. Hariharan, J. Lin, I. Singh, and G. O'Rourke, K. Ng, S. Y. Pai, C.-C. Li, Z. Huang, and S. K. Cheng, "Assembly challenges in developing 3D IC package with ultra high yield and high reliability," in *Proc. Electron. Compon. Technol. Conf.*, May 2015, pp. 1447–1451.
[4] A. Zarándy, *Focal-Plane Sensor-Processor Chips*. New York, NY, USA: Springer, 2011.
[5] *Intel Previews New Hybrid CPU Architecture with Foveros 3D Packaging*. Accessed: May 17, 2019. [Online]. Available: https://newsroom.intel.com/video-archive/video-intel-previews-new-hybrid-cpu-architecture-with-foveros-3d-packaging/
[6] X. Wu, "3D-IC technologies and 3D FPGA," in *Proc. Int. 3D Syst. Integr. Conf. (3DIC)*, Aug./Sep. 2015, pp. KN1.1–KN1.4.
[7] I. L. Markov, "Limits on fundamental limits to computation," *Nature*, vol. 512, pp. 147–154, Aug. 2014.
[8] M. Lee, J. S. Pak, and J. Kim, *Electrical Design of Through Silicon Via*. Amsterdam, The Netherlands: Springer, 2014.
[9] P. E. Garrou, M. Koyanagi, and P. Ramm, *3D process Technology: Robust Circuit and Physical Design for sub-65 nm Technology Nodes*. Hoboken, NJ, USA: Wiley, 2009.
[10] X. Yu, L. Li, Y. Zhang, H. Pan, and S. He, "Performance and power consumption analysis of memory efficient 3D network-on-chip architecture," in *Proc. 10th IEEE Int. Conf. Control Automat. (ICCA)*, Jun. 2013, pp. 340–344.
[11] H. Sun, J. Liu, R. Anigundi, N. Zheng, J. Lu, R. Ken, and T. Zhang, "Design of 3D DRAM and its application in 3D integrated multi-core computing systems," in *Proc. IEEE Design Test Comput.*, Sep. 2013, pp. 1–8.
[12] Y. Kikuchi, M. Takahashi, T. Maeda, M. Fukuda, Y. Koshio, H. Hara, H. Arakida, H. Yamamoto, Y. Hagiwara, T. Fujita, M. Watanabe, H. Ezawa, T. Shimazawa, Y. Ohara, T. Miyamori, M. Hamada, M. Takahashi, and Y. Oowaki, "A 40 nm 222 mW H.264 full-HD decoding, 25 power domains, 14-Core application processor with ×512b stacked DRAM," *IEEE J. Solid-State Circuits*, vol. 46, no. 1, pp. 32–41, Jan. 2011.
[13] K. Abe, M. P. Tendulkar, J. R. Jameson, P. B. Griffin, K. Nomura, S. Fujita, and Y. Nishi, "Ultra-high bandwidth memory with 3D-stacked emerging memory cells," in *Proc. IEEE Int. Conf. Integr. Circuit Design Technol. Tutorial*, Jun. 2008, pp. 203–206.

[14] D. H. Kim *et al.*, "Design and analysis of 3D-MAPS (3D massively parallel processor with stacked memory)," *IEEE Trans. Comput.*, vol. 64, no. 1, pp. 112–125, Jan. 2015.
[15] M. Koyanagi, H. Kobayashi, T. Aoki, T. Sueyoshi, and T. Kamada, "A 3D-VLSI Architecture for Future Automotive Visual Recognition," in *VLSI Design and Test for Systems Dependability*. Japan, Tokyon: Springer, 2019, pp. 719–733.
[16] K. Kim, S. Lee, J.-Y. Kim, M. Kim, and H.-J. Yoo, "A 125 GOPS 583 mW network-on-chip based parallel processor with bio-inspired visual attention engine," *IEEE J. Solid-State Circuits*, vol. 44, no. 1, pp. 136–147, Jan. 2009.
[17] K. Jia, Z. Liu, F. Qiao, X. Liu, Q. Wei, and H. Yang, "AICNN: Implementing typical CNN algorithms with analog-to-information conversion architecture," in *Proc. IEEE Comput. Soc. Annu. Symp. VLSI*, Jul. 2017, pp. 80–85.
[18] V. S. Ghaderi, D. Song, J. Choma, and T. W. Berger, "Nonlinear cognitive signal processing in ultralow-power programmable analog hardware," *IEEE Trans. Circuits Syst., II, Exp. Briefs*, vol. 65, no. 2, pp. 124–128, Feb. 2015.
[19] F. Dubois, A. Sheibanyrad, F. Petrot, and M. Bahmani, "Elevator-first: A deadlock-free distributed routing algorithm for vertically partially connected 3D-NoCs," *IEEE Trans. Comput.*, vol. 62, no. 3, pp. 609–615, Mar. 2013.
[20] N. Miura, Y. Koizumi, Y. Take, H. Matsutani, T. Kuroda, H. Amano, R. Sakamoto, M. Namiki, K. Usami, M. Kondo, and H. Nakamura, "A scalable 3D heterogeneous multicore with an inductive ThruChip interface," *IEEE Micro*, vol. 33, no. 6, pp. 6–15, Nov./Dec. 2013.
[21] D. Park, S. Eachempati, R. Das, A. K. Mishra, Y. Xie, N. Vijaykrishnan, and C. R. Das, "MIRA: A Multi-Layered on-chip interconnect router architecture," in *Proc. 35th Int. Symp. Comput. Archit.*, Jun. 2008, pp. 251–261.
[22] L. Bamberg, J. M. Joseph, R. Schmidt, T. Pionteck, and A. Garcia-Ortiz, "Coding-aware link energy estimation for 2D and 3D networks-on-chip with virtual channels," in *Proc. 28th Int. Symp. Power Timing Modeling, Optim. Simulations*, Jul. 2018, pp. 222–228.
[23] J. M. Joseph, C. Blochwitz, and A. García-Ortiz, and T. Pionteck, "Area and power savings via asymmetric organization of buffers in 3D-NoCs for heterogeneous 3D-SoCs," *Microprocessors Microsyst.*, vol. 48, pp. 36–47, Feb. 2017.
[24] *Ratatoskr Framework*. Accessed: Sep. 19, 2019. [Online]. Available: https://github.com/jmjos/ratatoskr
[25] A. E. Kiasari, A. Jantsch, and Z. Lu, "Mathematical formalisms for performance evaluation of networks-on-chip," *ACM Comput. Surv.*, vol. 45, no. 3, Jan. 2013, Art. no. 38.
[26] A. B. Kahng, B. Lin, and S. Nath, "Explicit modeling of control and data for improved NoC router estimation," in *Proc. Design Automat. Conf. (DAC)*, Jun. 2012, pp. 392–397.
[27] N. Nikitin and J. Cortadella, "A performance analytical model for Network-on-Chip with constant service time routers," in *Proc. Int. Conf. Comput.-Aided Design*, Nov. 2009, pp. 571–578.
[28] U. Y. Ogras, P. Bogdan, and R. Marculescu, "An analytical approach for network-on-chip performance analysis," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 29, no. 12, pp. 2001–2013, Dec. 2010.
[29] M. Arjomand and H. Sarbazi-Azad, "A comprehensive power-performance model for NoCs with multi-flit channel buffers," in *Proc. 23rd Int. Conf. Supercomput.*, Jun. 2009, pp. 470–478.
[30] S. Foroutan, Y. Thonnart, R. Hersemeule, and A. Jerraya, "An analytical method for evaluating network-on-chip performance," in *Proc. Design, Automat. Test Eur. Conf. Exhib. (DATE)*, Mar. 2010, pp. 1629–1632.
[31] J. Kim, C. Nicopoulos, D. Park, R. Das, Y. Xie, V. Narayanan, M. S. Yousif, and C. R. Das, "A novel dimensionally-decomposed router for on-chip communication in 3D architectures," *ACM SIGARCH Comput. Archit. News*, vol. 35, no. 2, pp. 138–149, May 2007.
[32] M. Ebrahimi, X. Chang, M. Daneshtalab, J. Plosila, P. Liljeberg, and H. Tenhunen, "DyXYZ: Fully adaptive routing algorithm for 3D NoCs," in *Proc. 21st Euromicro Int. Conf. Parallel, Distrib., Network-Based Process.*, Feb./Mar. 2013, pp. 499–503.
[33] A. B. Ahmed and A. B. Abdallah, "LA-XYZ: Low latency, high throughput look-ahead routing algorithm for 3D network-on-chip (3D-NoC) architecture," in *Proc. IEEE 6th Int. Symp. Embedded Multicore SoCs*, Sep. 2012, pp. 167–174.
[34] A. B. Ahmed and B. A. Abderazek, "Adaptive fault-tolerant architecture and routing algorithm for reliable many-core 3D-NoC systems," *J. Parallel Distrib. Comput.*, vols. 93–94, pp. 30–43 Jul. 2016.
[35] T. Krishna, A. Kumar, P. Chiang, M. Erez, and L.-S. Peh, "NoC with near-ideal express virtual channels using global-line communication," in *Proc. 16th IEEE Symp. High Perform. Interconnects*, Aug. 2008, pp. 11–20.
[36] C. O. Chen, S. Park, T. Krishna, S. Subramanian, A. P. Chandrakasan, and L. Peh, "SMART: A single-cycle reconfigurable NoC for SoC applications," in *Proc. Design, Autom. Test Eur. Conf. Exhibit. (DATE)*, Mar. 2013, pp. 338–343.
[37] P. Vivet, Y. Thonnart, R. Lemaire, C. Santos, E. Beigné, C. Bernard, F. Darve, D. Lattard, I. Miro-Panadès, D. Dutoit, F. Clermidy, S. Cheramy, A. Sheibanyrad, F. Pétrot, E. Flamand, J. Michailos, A. Arriordaz, L. Wang, and J. Schloeffel, "A 4×4×2 homogeneous scalable 3D network-on-chip circuit with 326 MFlit/s 0.66 pJ/b robust and fault tolerant asynchronous 3D links," *IEEE J. Solid-State Circuits*, vol. 52, no. 1, pp. 33–49, Jan. 2017.
[38] F. Darve, A. Sheibanyrad, P. Vivet, and F. Petrot, "Physical implementation of an asynchronous 3D-NoC router using serial vertical links," in *Proc. IEEE Comput. Soc. Annu. Symp. VLSI*, Jul. 2011, pp. 25–30.
[39] A. Garcia-Ortiz, L. Bamberg, and A. Najafi, "Low-power coding: Trends and new challenges," *J. Low Power Electron.*, vol. 13, no. 3, pp. 356–370, Sep. 2017.
[40] *International Technology Roadmap for Semiconductors (ITRS)*, ITRS, London, U.K., 2013.
[41] J. M. Joseph, L. Bamberg, I. Hajjar, R. Schmidt, T. Pionteck, and A. Garcí-Ortiz, "Simulation environment for link energy estimation in networks-on-chip with virtual channels," *Integration*, vol. 68, pp. 147–156, Sep. 2019.
[42] X. Chen and N. K. Jha, "A 3-D CPU-FPGA-DRAM hybrid architecture for low-power computation," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 24, no. 5, pp. 1649–1662, May 2016.
[43] J. Duato, "A new theory of deadlock-free adaptive routing in wormhole networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 4, no. 12, pp. 1320–1331, Dec. 1993.
[44] B. Korte and J. Vygen, *Combinatorial Optimization: Theory Algorithms*, 2nd ed. Berlin, Germany: Springer, 2002.
[45] W. Dally and C. Seitz, "Deadlock-free message routing in multiprocessor interconnection networks," *IEEE Trans. Comput.*, vol. 36, no. 5, pp. 547–553, May 1987.
[46] M. Palesi and M. Daneshtalab, *Routing Algorithms Networks-On-Chip*. Springer, 2014.
[47] J. M. Joseph, S. Wrieden, C. Blochwitz, A. García-Oritz, and T. Pionteck, "A simulation environment for design space exploration for asymmetric 3D-network-on-chip," in *Proc. 11th Int. Symp. Reconfigurable Commun.-Centric Syst.-on-Chip*, Jun. 2016, pp. 1–8.
[48] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Dec. 2001, pp. 1.
[49] J. Shi and C. Tomasi, "Good features to track," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 1994, pp. 593–600.
[50] C. Tomasi and T. Kanade, "Detection and tracking of point features," Carnegie Mellon Univ., Pittsburgh, PA, USA, Tech. Rep. CMU-CS-91-132, 1991.

JAN MORITZ JOSEPH received the B.Sc. degree in medical engineering and the M.Sc. degree in informatics from the Universität zu Lübeck, Germany. From 2008 to 2014, he was a Scholarship Holder of the German National Merit Foundation. He is currently a Research Assistant with Otto-von-Guericke-Universität, Magdeburg, Germany. His main focus includes 3D integration. He also researches about heterogeneous integration, interconnects, and NoCs.

LENNART BAMBERG received the B.Sc. and M.Sc. degrees in electrical and information engineering from the University of Bremen, Germany, in 2014 and 2016, respectively, where he is currently pursuing the Ph.D. degree. Since 2016, he has been a Teaching and Research Associate with the University of Bremen. In 2019, he joined the Georgia Institute of Technology, Atlanta, GA, USA, as a Visiting Scholar for four months. His research interests include low-power design, communication-centric design, and heterogeneous 3D SoCs. He received the Best Paper Award at the PATMOS 2017 and the PATMOS 2018.

DOMINIK ERMEL received the B.Sc. degree in mathematics from Otto-von-Guericke-University, Magdeburg, in 2016, where he is currently pursuing the M.Sc. degree. He is interested in mathematical optimization and has been applying combinatorial optimization on the topics of heterogeneous 3D integration and networks-on-chip.

BEHNAM RAZI PERJIKOLAEI received the B.Sc. degree in computer engineering from the Shahid Bahonar University of Kerman, Iran, in 2007, and the M.Sc. degree in computer systems architecture from the IAU Science and Research Branch, Tehran, Iran, in 2012. He is currently pursuing the M.Sc. degree in control, microelectronics, and microsystems with the University of Bremen, Germany. From 2012 to 2016, he was with the Industrial Automation Department, ACECR Sharif University Branch, Iran. His current research interests include network-on-chip communication architectures, especially for FPGA and heterogeneous 3D architecture.

ANNA DREWES received the B.Sc. and M.Sc. degrees in computer science from the University of Lübeck, Germany, in 2015 and 2017, respectively. She is currently pursuing the Ph.D. degree with the Institute for Information Technology and Communications, Otto-von-Guericke-University, Magdeburg, Germany, where she is also a Research Assistant. Her research interests include communications infrastructure and interconnects, especially for FPGAs, as well as the use of heterogeneous systems for database query processing.

ALBERTO GARCÍA-ORTIZ received the Diploma degree in telecommunication systems from the Universitat Politècnica de València, in 1998, and the Ph.D. degree *(summa cum laude)* from the Institute of Microelectronic Systems, Technische Universität Darmstadt, Germany, in 2003. He was with Newlogic, Austria, for two years. From 2003 to 2005, he was a Senior Hardware Design Engineer with IBM Deutschland Development and Research, Böblingen. He then joined AnaFocus, Seville, Spain. Since 2011, he has been a Full Professor with the Chair of Integrated Digital Systems, University of Bremen. His research interests include low-power design, communication-centric design, SoC integration, and variation-aware design. He received the Outstanding Dissertation Award from the European Design and Automation Association, in 2004, and the IBM Innovation Award for contributions to leakage estimation, in 2005. He serves as an Editor and a Reviewer of several conferences, journals, and projects.

THILO PIONTECK received the Diploma degree and the Ph.D. (Dr.Ing.) degree in electrical engineering from Technische Universität Darmstadt, Germany, in 1999. In 2008, he was appointed as an Assistant Professor of integrated circuits and systems at the Universität zu Lübeck, Germany. From 2012 to 2014, he was the substitute of the Chair of Embedded Systems, Technische Universität Dresden, and the Chair of Computer Engineering, Technische Universität at Hamburg, Harburg, Germany. In 2015, he was appointed as a Professor with the Chair of Organic Computing, Universität zu Lübeck, with research focus on adaptive digital systems. He was appointed to the Otto-von-Guericke Universität, Magdeburg, Germany, in 2016. He is currently the Chair for hardware-oriented computer science at the Otto-von-Guericke-Universität. His research interests include network-on-chips, adaptive system design, runtime reconfiguration, and hardware/software co-design.