ENGINEERING Information Technology & Electronic Engineering
https://www.academax.com/EITEE; https://www.engineering.org.cn
https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=11344065
E-mail: eng.itee@zju.edu.cn

Engineering Journals
Available on IEEE Xplore®

Research Article

https://doi.org/https://doi.org/10.1631/ENG.ITEE.2025.0156

A hybrid redundancy and serialization fault-tolerant architecture for through-silicon via interconnects

Chenglong SUN[1,2], Yanqing ZHOU[1,2], Qi WANG[3]✉, Yan ZHANG[1,2]

[1]*School of Computer and Information Engineering, Fuyang Normal University, Fuyang 236037, China*
[2]*Anhui Engineering Research Center for Intelligent Computing and Information Innovation, Fuyang Normal University, Fuyang 236037, China*
[3]*School of Computer Science and Information Engineering, Hefei University of Technology, Hefei 230601, China*

Abstract: Three-dimensional network-on-chips (3D NoCs) are increasingly used to improve scalability in multicore systems. Through-silicon vias (TSVs) are a critical technology for enabling vertical interconnects between NoC layers. However, TSV-based interlayer connections are highly prone to faults resulting from manufacturing defects, aging, or other sources, which compromise system reliability. To address these challenges, particularly in chiplet-based 3D NoCs, robust fault-tolerant mechanisms are crucial for maintaining operational integrity in the presence of TSV faults. We introduce a novel fault-tolerant architecture designed to ensure persistent communication reliability despite permanent vertical link failures, named HyRAS, a hybrid redundancy and serialization method. Our approach is built on two synergistic mechanisms. First, a lightweight spatial redundancy scheme leverages shared TSV resources to mitigate the impact of isolated faults. Second, for more severe fault scenarios, an adaptive serialization strategy is employed to maintain connectivity by efficiently using the remaining functional links. The architecture is rigorously evaluated through functional simulations using both synthetic traffic patterns and realistic application workloads. Compared to contemporary fault-tolerant methods, HyRAS achieves up to 28.2% higher throughput under realistic workloads with significant defect clusters. These gains are achieved with only modest overhead, incurring a 14.52% increase in area and 8.86% in power consumption relative to the standard redundancy-based router.

Key words: Three-dimensional network-on-chip (3D NoC); Through-silicon vias (TSVs); Redundancy; Fault tolerance

1 Introduction

The relentless drive for greater computational power has spurred an architectural shift from computation-centric designs to communication-centric paradigms. In this landscape, network-on-chips (NoCs) have become the de facto interconnect standard for modern multicore systems (Agarwal et al., 2025; Xiong et al., 2025). To push beyond the limits of two-dimensional (2D) circuits, three-dimensional (3D) integration has emerged, stacking multiple device layers and connecting them with high-speed vertical interconnects. The primary advantage of 3D NoCs is the substantial reduction in global interconnect length, which translates to higher performance and lower power dissipation (Bose and Ghosal, 2020; Fu et al., 2021; Taheri et al., 2023).

However, this technological advancement is not without its challenges. The combination of shrinking process nodes, complex 3D packaging, and higher operating frequencies exacerbates error rates, thermal issues, and timing violations. Among the critical components of 3D integrated circuits (3D-ICs), through-silicon vias (TSVs) are favored for their high-speed and low-power vertical communication capabilities. However, as system complexity grows, so does the threat to reliability. The TSV manufacturing process is notoriously prone to defects, leading to permanent (or hard) faults that can arise during fabrication or from operational stresses such as time-dependent dielectric breakdown (TDDB), electromigration, and thermal cycling (Dang et al., 2020a; da Silva et al., 2025). To address these challenges, various studies have proposed dependable NoC architectures to guarantee strong system performance.

Among the vertical interconnection technologies enabling 3D-ICs, TSVs are widely adopted due to their capacity for high-speed and power-efficient communication between layers. As single-chip designs advance toward peta-scale computational performance, with exascale systems anticipated in the coming decade, system reliability has emerged as a critical challenge. The reliability of on-chip architectures is negatively impacted by increasing power consumption, process variability, and component density. For instance, rapid fluctuations in power consumption can induce on-chip voltage variations, leading to data transmission errors (Hou et al., 2024). Permanent faults, also known as hard faults, occur during manufacturing

✉ Qi WANG, wq_hfut@163.com
Chenglong SUN, https://orcid.org/0000-0002-7492-0542
CLC number: TN47

Received: Nov. 26, 2025; Revision accepted: Apr. 15, 2026;
Crosschecked: Apr. 20, 2026

or operation and persistently disrupt circuit functionality from their moment of onset. The primary physical mechanisms contributing to these permanent faults include TDDB, negative bias temperature instability (NBTI), hot carrier injection (HCI), and electromigration, all of which are driven by material degradation and manufacturing challenges.

Faults within NoC systems can precipitate critical failures, such as network deadlocks, packet loss, elevated latency, and data transmission errors, all of which significantly hinder overall chip performance. It is therefore essential to integrate robust reliability mechanisms into the NoC design from the outset to mitigate these risks and ensure continuous system functionality.

To address the limitations of existing designs, we propose a novel fault-tolerant method named HyRAS, a hybrid redundancy and serialization method featuring reconfigurable links for 3D chiplet NoCs. The scope of HyRAS specifically focuses on the issue of fault tolerance in NoC links. The fault tolerance of routers has been extensively addressed by other researchers (Liu et al., 2022; Mercier et al., 2022; Khalil et al., 2024) and is thus outside the scope of this paper. The fault-tolerant methodology of our proposed architecture is founded on the established techniques such as spatial redundancy, the use of idle cycles on existing resources (Ni et al., 2020; Xu et al., 2022), serialization, and time-division multiplexing over fault-free links (Reddy et al., 2017). In this paper, we advance a fault-tolerant method that leverages the combined strengths of redundancy and serialization strategies to adaptively mitigate faulty links according to varying fault granularities, thereby maximizing the preservation of system performance. Specifically, the main contributions of this paper are as follows:

(1) We define the operational conditions under which TSV redundancy and serialization strategies are most effective for preserving link connectivity across different fault granularities.

(2) A hybrid fault-tolerant architecture called HyRAS that integrates these two strategies. For a low number of faults, the system em-ploys spatial redundancy by remapping signals to fault-free bit lines. As the number of faults increases, it transitions to a serialization strategy, or a combination of both, to ensure robust fault tolerance.

(3) We conduct a comprehensive evaluation of the architecture using real-world application benchmarks and provide detailed hardware synthesis results, demonstrating its effectiveness against other recent fault-tolerant solutions.

In summary, HyRAS provides a comprehensive approach to protecting against TSV clustering defects in 3D-ICs by offering both coarse-grained and fine-grained fault-tolerant methods along with recovery strategies.

2 Related works

Reliability in NoC architectures has been a focal point of recent research, targeting both router logic and interconnects. For instance, a self-healing router approach (Khalil et al., 2021) is proposed to effectively mitigate hard faults within the router's logic and buffers, ensuring high intranode reliability and performance. However, ensuring the reliability of vertical interconnects in 3D-ICs remains a distinct and critical challenge that requires dedicated solutions.

In the context of 3D NoC architectures, several TSV fault-tolerant methods have been presented (Akbari et al., 2012; Niazmand et al., 2016; Dang et al., 2020b; Song et al., 2024; Asadboland et al., 2025; Zhang et al., 2025). Fault-tolerant routing algorithms for the 3D mesh architecture are considered lightweight and efficient solutions for addressing coarse-grained TSV failures. However, similar to Niazmand et al. (2016), these algorithms primarily reroute transmission paths based on logical decisions to achieve coarse-grained fault tolerance. The AFRA (Akbari et al., 2012) algorithm enables tolerance for a limited number of faulty vertical links. Additionally, a 3D variant of logic-based distributed routing (LBDR) (Flich and Duato, 2008), termed LBDR3D, was recently presented in Niazmand et al. (2016). Like LBDR, LBDR3D supports various partially adaptive routing algorithms and is fully reconfigurable to handle both faulty horizontal and vertical links. It ensures deadlock- and livelock-free operation, employing the same methodology as elevator-first and requiring the minimal number of virtual channels to separate upward and downward traffic flows. However, LBDR3D stores only a fixed number of bits per router to identify healthy elevators, limiting its adaptability. Notably, all aforementioned methods discard any entire TSV that contains a faulty bit line, reducing resource efficiency.

Considering the interdependence of functional and spare TSVs, the adaptive fault-tolerant strategy (AFTS) of Chen et al. (2019) initially calculates the maximum number of tolerable faults within each TSV group to optimize fault coverage. Subsequently, the AFTS proposes a fault-tolerant TSV planning methodology to enhance yield awareness during TSV planning. Leveraging the low-overhead characteristics of coding-based approaches, the dynamic modified Fibonacci numeral system-based crosstalk avoidance code (Dy-MFNS-CAC) (Wei et al., 2023) develops the dynamic modified Fibonacci numeral system (Dy-MFNS) and its associated crosstalk avoidance code to address TSV signal integrity issues. Cascaded Dy-MFNS adders and a Dy-MFNS codec are designed to dynamically generate Dy-MFNS-CAC codewords based on the TSV health status, enabling adaptive fault mitigation. These codewords simultaneously suppress crosstalk and repair hard faults guided by the TSV fault flags. However, these methods overlook delay overheads associated with temporal redundancy, compromising performance in latency-sensitive 3D NoC applications. Xu et al. (2022) proposed an effective TSV repair strategy for a router-based TSV redundancy architecture, optimizing fault tolerance while explicitly addressing delay overhead to maintain performance. An integer linear programming (ILP)-based model is developed to repair clustered TSV faults while minimizing both the delay overhead and hardware cost. TSV repair using switch matrix topology (TRUST) is proposed in Lee et al. (2023), using a switch matrix to provide high routing flexibility for TSV connections in 3D-ICs and achieving a 100% repair rate for 3D-ICs with faulty TSVs up to the number of redundant TSVs (RTSVs). Consequently, a 100% repair rate can be achieved for 3D-ICs with faulty TSVs equal to or fewer than RTSVs. Maity et al. (2021) proposed a tree-based TSV repair framework that optimizes hardware resource utilization to achieve a higher repair rate through hierarchical TSV redundancy allocation. This framework partitions spare TSVs (STSVs) into hierarchical groups to enable adaptive redundancy sharing across varying fault patterns. Ni et al. (2021) introduced a honeycomb-time division multiple access (honeycomb-TDMA) TSV design that mitigates multiple clustered faults without repairing

RTSVs, significantly reducing area overhead and improving manufacturing yield. Collectively, these approaches address TSV failures primarily through hardware redundancy, leading to increased area and power consumption.

To the best of our knowledge, the literature lacks a flexible, elastic, and adaptive fault-tolerant method that delivers high performance and a runtime fault-tolerant solution for partially connected 3D NoCs without imposing rigid constraints on balancing time and space redundancy. Motivated by this gap and in contrast to existing approaches, we propose HyRAS, a hybrid redundancy and serialization method designed to tolerate faults in vertical links, prevent packet loss in the presence of TSV failures, and sustain the high performance of 3D NoCs despite faults. We first introduce the partially connected 3D NoC topology used in this study, which features a hybrid configuration of 2D and 3D routers to enhance fault tolerance and connectivity. Next, we describe an adaptive fault-tolerant scheme that dynamically balances temporal and spatial redundancy to mitigate through-silicon via faults while minimizing performance overhead. Finally, the dynamic integration of redundancy and serialization methods ensures robust system reliability.

To better highlight the novelty and structural advantages of the proposed architecture, Table 1 summarizes a qualitative comparison between HyRAS and recent state-of-the-art fault-tolerant works, including recent routing-based methods and TSV modeling approaches.

While existing methods offer significant reliability improvements, they predominantly rely on a single fault-tolerant domain. For instance, spatial redundancy approaches such as CETR provide zero-latency repair for sparse faults but fail completely when local defect clusters exceed the available RTSVs. Conversely, temporal serialization techniques such as SOTR guarantee connectivity under severe fault clustering but impose significant multicycle latency penalties uniformly, even for isolated defects. Furthermore, although several prior adaptive methods attempt to dynamically reallocate spare TSVs across different groups (e.g., AFTS, TRUST), their repair capabilities fundamentally remain bounded by the total physical count of spare resources. The primary novelty of the proposed HyRAS architecture lies in its fine-grained and online integration of both spatial and temporal domains. Unlike prior hybrid schemes, HyRAS implements an autonomous and block-level state-transition mechanism (normal, fine-grained, coarse-grained, and mixed). This unique design resolves sparse faults using latency-free spatial redundancy (akin to CETR) while strictly isolating the performance penalties of temporal serialization (akin to SOTR) only to specific TSV blocks overwhelmed by clustered faults. Consequently, HyRAS pushes the reliability boundary beyond the physical limits of RTSV counts without suffering the pervasive performance degradation characteristic of pure serialization schemes.

3 Baseline architecture

As illustrated in Fig. 1, a 3D NoC architecture comprising multiple 2D mesh layers vertically interconnected through TSVs. Each layer integrates conventional 2D routers, each equipped with five ports (east, west, south, north, and local), alongside 3D routers featuring additional up and/or down ports to facilitate vertical interlayer communication via TSVs. These 3D routers, referred to as "elements" (Dubois et al., 2013), enable connectivity to the upper layer (upward elevator) or the lower layer (downward elevator).

3.1 TSV fault tolerance

In this subsection, we present a concise overview of existing fault-tolerant techniques for TSV in 3D NoC architectures to provide context for our proposed approach. We omit the discussion of TSV testing, as our proposed fault-tolerant method, HyRAS, operates independently of specific testing methodologies, emphasizing runtime fault mitigation. Regarding the fault detection phase, HyRAS assumes the presence of a standard and independent fault detection unit (FDU) capable of both offline and periodic online monitoring (e.g., built-in self-test or concurrent error detection). As HyRAS primarily targets permanent (hard) faults—such as those induced by aging or electromigration—the FDU's detection latency, typically spanning tens of clock cycles, has a negligible impact on the overall system lifespan, and the subsequent HyRAS state reconfiguration completes in only a few cycles. In terms of detection accuracy, our architectural evaluation assumes a high-fidelity FDU. Nevertheless, HyRAS is intrinsically robust to false positives: An erroneously flagged fault will merely trigger an unnecessary transition to the fine-grained or coarse-grained state, slightly penalizing local latency but strictly preserving system functionality and preventing deadlocks. Mitigating false negatives remains the responsibility of the underlying

Table 1 Comparison of fault-tolerant strategies in recent 3D NoC architectures

Method	Fault-tolerant strategy	Addressed fault type	Key advantage	Limitation
(Ouyang et al., 2024)	Fault-aware path detour	Link/Node faults	Low hardware overhead	Has congestion and latency penalties due to detoured traffic
(Kirtonia et al., 2026)	Fault modeling & characterization	High-frequency TSV defects	Accurate physical defect models	Lacks an active, runtime architectural repair mechanism
CETR (Ni et al., 2021)	Shared TSV redundancy	Clustered TSV faults	Low latency at low fault rates	Fails when faults exceed the fixed RTSV capacity
Securet3d (da Silva et al., 2025)	Adaptive bypass routing	Vertical partially connected NoCs	Secure and circumvent faulty nodes	Relies on existing paths and does not physically restore vertical bandwidth
SOTR (Dang et al., 2020a)	Time-division serialization	Clustered TSV defects	High repair rate under severe fault clusters	Incurs significant latency overhead even at very low fault rates
HyRAS (Proposed)	Hybrid: redundancy + serialization	Sparse & clustered TSV faults	Adaptive latency-throughput balance	Has slightly higher control logic complexity for state transitions

robust FDU design and higher-level network retransmission protocols. As shown in Fig. 2, current TSV fault-tolerant techniques are broadly classified into two categories: rerouting-based and coding-based methods, commonly termed reroute TSV and coding TSV, respectively (Lung et al., 2011). Rerouting-based methods employ multiplexers (MUXes) to dynamically redirect signals from faulty TSVs to functional TSVs, thereby preserving communication integrity in 3D NoCs. Notable rerouting techniques, such as signal rerouting and signal shifting, use MUX-based switching to reroute signals from defective TSVs to operational TSVs (Lee et al., 2023). Conversely, coding-based methods leverage error-correcting codes, such as Hamming codes, supported by extensive RTSVs to rectify faulty signals, requiring dynamic reconfiguration of encoder and decoder circuits. The inclusion of RTSVs alongside primary links introduces computational delays in point-to-point transmissions, adversely affecting latency in high-performance 3D NoC applications. Within the redundancy strategy framework, as depicted in Fig. 2, signal-switching and signal-shifting methods facilitate signal transmission to a RTSV from the fault location (Kang et al., 2010; Hsieh and Hwang, 2012), with detailed configurations illustrated in Fig. 3. However, in scenarios where TSV defects cluster in dense regions, these methods significantly diminish the yield of 3D integrated circuits, as defective TSVs may remain unrepairable despite the availability of RTSVs, thereby constraining scalability in high-density designs.

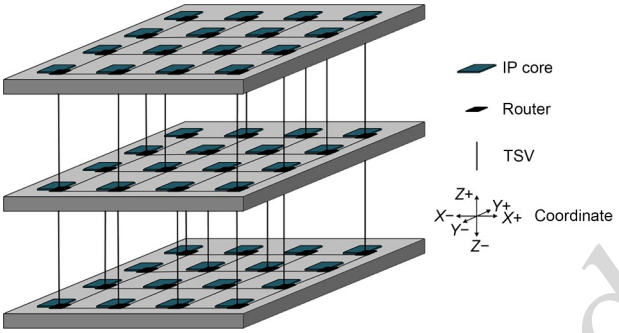

Fig. 1 Structure of a conventional partially connected 3D NoC

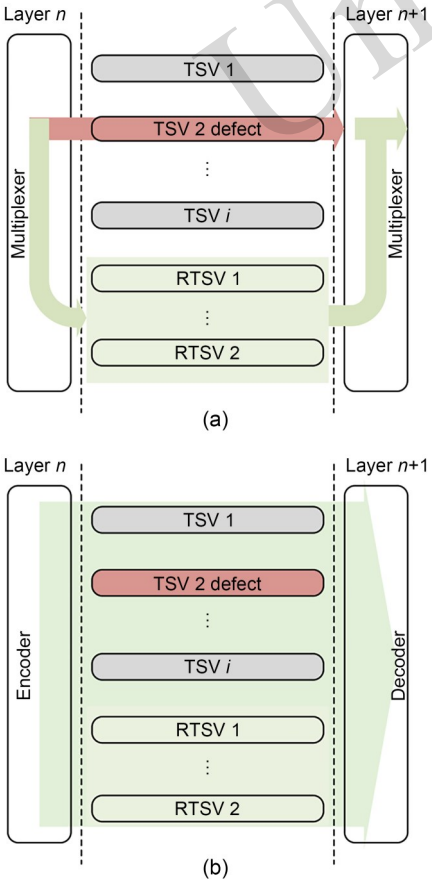

Fig. 2 TSV redundancy methods: (a) rerouting-based; (b) coding-based

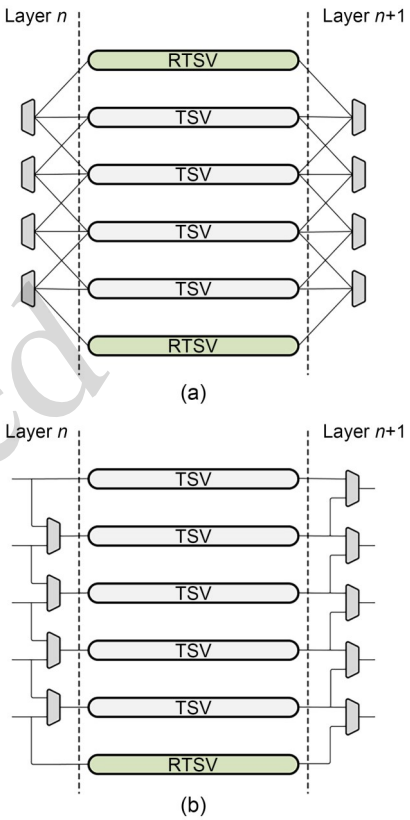

Fig. 3 Variants of Rerouting-based TSV redundancy schemes: (a) signal-switching; (b) signal-shifting methods

3.2 TSV fault tolerance

TSV defect-distribution models are categorized into two main types: uniform defect distribution and clustered defect distribution, each significantly impacting the design of fault-tolerant strategies for 3D NoC architectures. In the uniform defect model, the probability of a defect in each TSV is independent, typically associated with random defects such as void formation or delamination due to thermal stress (Wang et al., 2019). In contrast, many TSV defects result from cumulative failures during imperfect bonding processes, influenced by factors such as wafer warpage, surface roughness, and silicon die cleanliness (Jiang et al., 2013). These conditions lead to a defect clustering effect where a faulty TSV increases the likelihood of defects in adjacent TSVs, a phenomenon observed in both conventional semiconductor manufacturing and 3D-IC fabrication. The clustered defect-distribution model captures this scenario, reflecting heightened defect probabilities in proximity to a faulty TSV. In our proposed HyRAS method, we adopt a redundancy strategy to address isolated or sparse faults within a TSV cluster, complemented by a

serialization strategy to handle multiple clustered faults within TSV groups. By effectively integrating these approaches, HyRAS mitigates diverse TSV fault types in 3D NoCs, ensuring robust fault tolerance.

4 Hybrid redundancy and serialization fault-tolerant architecture

In this section, we present HyRAS, an innovative fault-tolerant approach designed for adaptive and resilient transmission in a 3D NoC architecture. The HyRAS architecture efficiently mitigates both sparse and clustered TSV faults in 3D NoC systems, ensuring robust and reliable vertical communication.

To enhance clarity and ease of reference for the subsequent architectural descriptions and the proposed repair algorithm, Table 2 summarizes the key notations and terminologies used throughout this section.

4.1 TSV serialization

TSVs serve as critical conduits for interlayer communication in 3D-ICs, enabling seamless signal transmission between stacked dies. However, manufacturing defects, such as pinholes and voids, can impair TSV functionality, leading to disruptions in interlayer connectivity. To mitigate these issues, signals intended for faulty TSVs must be redirected to operational TSVs to ensure continuous communication. Thus, a robust fault-tolerant strategy, such as HyRAS, is vital for addressing TSV defects in 3D NoC architectures, maintaining reliable and efficient signal transmission.

To mitigate clustered TSV faults while achieving low overhead and a high repair rate, the HyRAS architecture organizes TSVs into distinct blocks, as illustrated in Fig. 4. Specifically, eight RTSVs are divided into four blocks, each containing two RTSVs, labeled alphabetically for clarity. Each RTSV comprises a single TSV paired with a MUX to facilitate signal rerouting. Upon detecting a TSV defect, the HyRAS system redirects the affected signal from the faulty TSV to an operational RTSV within the same block, ensuring uninterrupted communication.

While RTSVs enable fault-tolerant vertical connectivity in 3D NoC architectures, they become ineffective in two scenarios: (a) when faults in a TSV block exceed the two available RTSVs and (b) when network contention causes persistent occupation of candidate reroute routers by higher-priority routers. To overcome these challenges, the HyRAS framework integrates a serialization mechanism alongside its redundancy strategy to ensure seamless vertical communication under severe TSV fault conditions. When faults in a TSV block surpass the capacity of available RTSVs, HyRAS employs serialization to maintain interlayer connectivity. This serialization approach uses time division multiple access (TDMA) to multiplex signals across a limited set of functional TSVs, ensuring robust connectivity even in high-fault scenarios, as shown in Fig. 5. However, serialization introduces additional buffer requirements and clock cycles for flit processing, resulting in increased area overhead and potential latency impacts on 3D NoC performance. Despite these trade-offs, serialization ensures reliable operation in fault-prone environments with elevated TSV defect rates. HyRAS allows designers to dynamically toggle between serialization and redundancy-based fault-tolerant routing during design or runtime, optimizing both performance and reliability based on system requirements. When the number of faulty TSVs exceeds the RTSV capacity within a block, HyRAS employs TDMA-based serialization to sustain connectivity with minimal additional hardware overhead, as illustrated in Fig. 5.

In the serialization strategy, a 3D router requires at least one operational TSV block to sustain vertical connectivity within a 3D NoC architecture. When only a single functional TSV block is available, HyRAS employs 1:4 serialization to maintain communication; When two functional TSV blocks are available, it adopts 1:2 serialization to enhance bandwidth utilization and fault tolerance. The unified serialization state table is used to record key information during the serialization process to ensure the sequentiality of serialization, mainly including port, virtual channel (VC), subflit, number, etc. In

Table 2 Summary of key notations and terminologies

Term	Definition and description
STSV	Signal TSV: a standard and primary vertical link designated for transmitting regular data flits.
RTSV	Redundant TSV: a spare vertical link allocated exclusively for fault-tolerant purposes to replace a faulty STSV.
TSV block	A localized grouping of STSVs and RTSVs. In HyRAS, each block contains two RTSVs, serving as the fundamental unit for fault mitigation.
$Fault_i$	A boolean flag indicating the presence of one or more faults detected within the i^{th} TSV block (where I = 0, 1, 2, and 3).
$Threshold_i$	A status field indicating whether the number of faulty STSVs in the i^{th} TSV block exceeds the capacity of its available RTSVs (0: within capacity; 1: exceeds capacity).

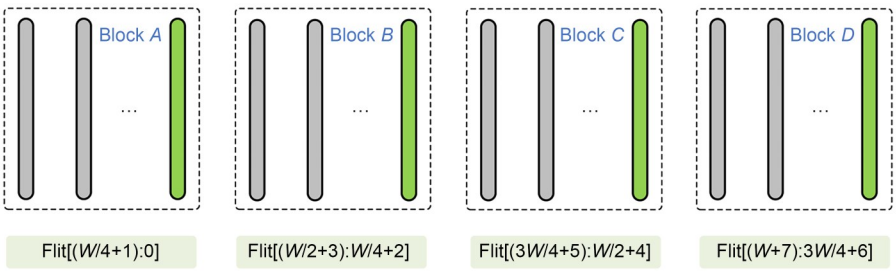

Fig. 4 Situation of TSV grouping

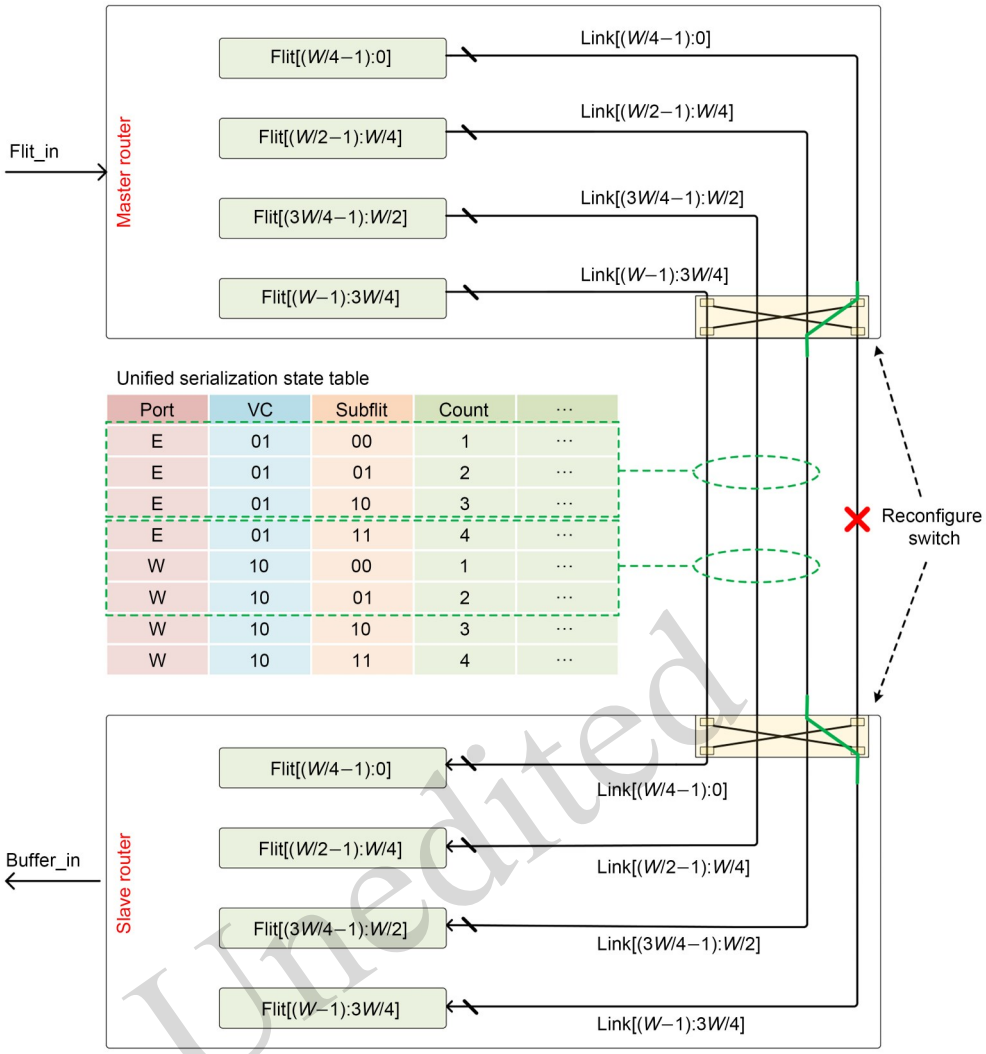

Fig. 5 Schematic representation of the serialization strategy

1:4 serialization, serial counters synchronize the transmission sequence, guaranteeing precise delivery of flit segments across four clock cycles using TDMA. Each flit is segmented into four parts, transmitted sequentially over these cycles. For 1:2 serialization, HyRAS applies a similar TDMA-based approach, dividing each flit into two segments, with each segment further split into quarters and transmitted over two clock cycles using functional TSV blocks, as illustrated in Fig. 5.

4.2 Composition of RTSV

To achieve a high repair rate for defective TSVs, each RTSV in HyRAS is designed to cover a specific subset of TSVs within a TSV block. When an RTSV is activated, it indicates that a fault has occurred within the TSV block, requiring the RTSV to reroute the affected signal to maintain vertical connectivity. Each TSV block includes two RTSVs, enabling fault tolerance for up to two faulty TSVs per block. Consequently, the two RTSVs in each TSV block are shared among faulty TSVs within the block and are dynamically reassigned to cover detected faults. At time period 0, when a single faulty TSV is detected within a TSV block, HyRAS assigns the nearest RTSV based on physical distance for fault tolerance, as illustrated in Fig. 6a. However, handling two faulty TSVs within a TSV block introduces greater complexity due to the limited availability of two RTSVs, necessitating optimized reassignment strategies. When two faulty TSVs occur simultaneously within a TSV block, HyRAS selects the nearest faulty TSV for each RTSV by comparing their serial numbers to the RTSV endpoint positions, ensuring efficient signal rerouting. When a first STSV fault occurs at time period 0 followed by a second fault at time period 1 within the same TSV block, three distinct reassignment situations arise, as depicted in Fig. 6. Faulty TSVs are marked in red, and RTSVs are marked in green, visually distinguishing defective and redundant components. In situations 0 and 1 of Fig. 6, the second faulty STSV does not disrupt the signal rerouting of the first faulty TSV, allowing the second fault to use the remaining RTSV (RTSV 0), with faulty TSV1 assigned to RTSV 0 and faulty TSV 0 assigned to RTSV 1 for effective fault tolerance. In situation 3 of Fig. 6, when the second faulty TSV is closer to the endpoint RTSV 1 than the first faulty TSV, HyRAS reassigns signal paths to optimize proximity, linking faulty TSV 1 to RTSV 1 and faulty TSV 0 to RTSV 0 for efficient fault-tolerant signal transmission.

In the HyRAS fault-tolerant architecture, a streamlined switching connection structure is implemented, using 2:1 MUXes for RTSVs located at the endpoints of each TSV block to reduce design complexity. This contrasts with the use of 3:1 MUXes for RTSVs positioned in the central region of the block.

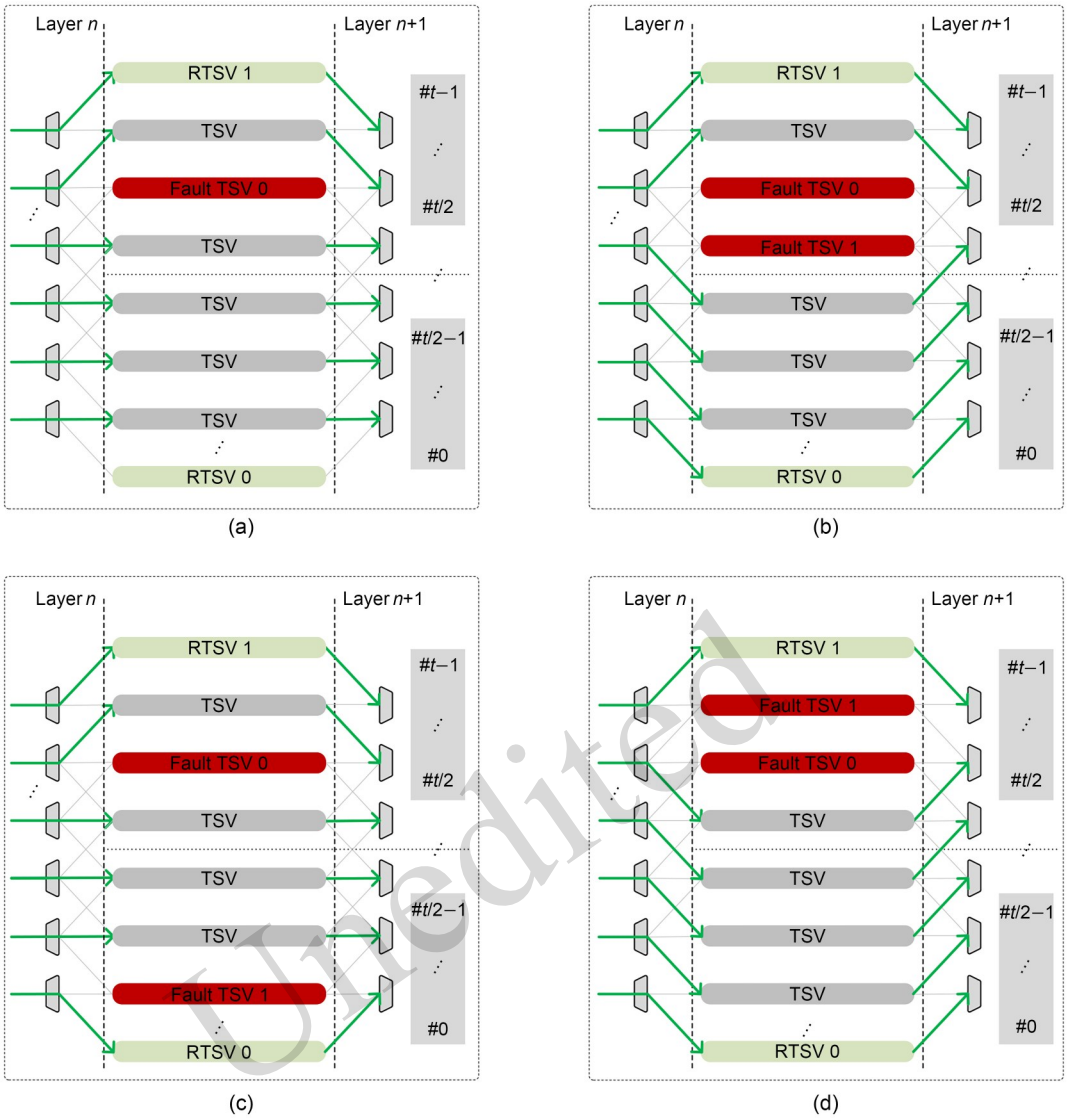

Fig. 6 Four situations of fault tolerance in redundant-based fault-tolerant method: (a) period 0; (b) period 1 & situation 0; (c) period 1 & situation 1; (d) period 1 & situation 2

4.3 Serialization and deserialization

As outlined previously, each TSV group in the HyRAS architecture includes two RTSVs, resulting in a total of eight RTSVs across four blocks for a node-to-node link. To accommodate serialization constraints, which mandate that TSV blocks be configured in multiples of two, HyRAS supports two configurations: two blocks with four RTSVs each or four blocks with two RTSVs each. The four-block configuration, with two RTSVs per block, is preferred for its optimal balance of fault tolerance and efficiency, as described earlier. Adopting a two-block configuration, with four RTSVs per block, enhances fault-tolerant capacity within each block but introduces significant design complexity. This arises from the intricate cross-switching required among RTSVs, which complicates the fault-tolerant logic needed to address third or fourth STSV faults within a block, demanding sophisticated rerouting mechanisms. Consequently, the four-block configuration, with two RTSVs per block, effectively manages most STSV faults while minimizing hardware overhead, making it the preferred choice in HyRAS's architecture for achieving a balance between fault tolerance and resource efficiency.

Upon deactivating faulty TSV bitlines, the operational bitlines maintain connectivity among 3D routers. In cases where RTSVs are insufficient to address defective TSVs, serializer segments flit into smaller units for serialized transmission. This mechanism relies on serializers and deserializers embedded within each router to leverage the remaining functional TSV links, ensuring dependable data transfer.

As illustrated in Fig. 7, the serializer in HyRAS's fault-tolerant architecture partitions each flit into four subflits through a parallel-input serial-output shift register, facilitating serialized transmission over constrained TSV links. When a TSV fault exceeds the capacity of RTSVs, it disrupts one-quarter of the TSV link's transmission capacity, requiring alternative fault-tolerant strategies to sustain connectivity. To address this, the segmented subflits are stored in an extend buffer within the router to prevent data loss during transmission interruptions. These subflits are then transmitted serially over operational TSV links using TDMA, leveraging the remaining fault-free TSV capacity for reliable communication. Crucially, a deserializer at the downstream 3D router reassembles the delayed subflits into a complete flit, ensuring uninterrupted data transmission.

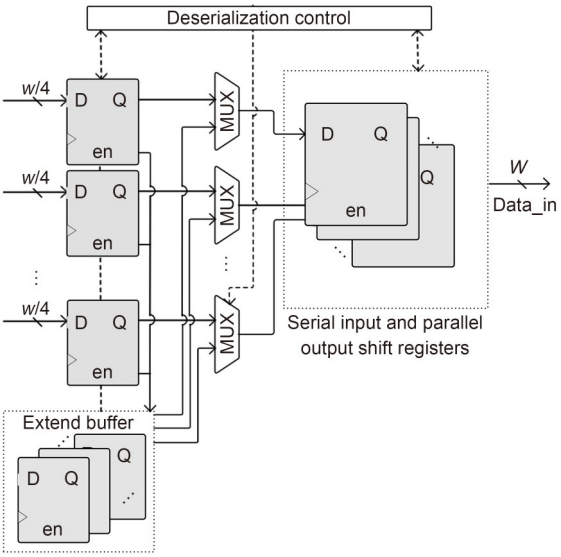

Fig. 7 Schematic representation of the serialization architecture

Following serialized transmission to the downstream 3D router, subflits are reconstructed into a complete flit using a serial-input parallel-output shift register, enabling continued data transmission across the 3D NoC. As depicted in Fig. 8, the deserializer facilitates this reassembly by processing serialized subflits received via functional TSV links. Critically, an extend buffer within the deserializer temporarily stores subflits from the upstream router that await reassembly, safeguarding data integrity amid fault-induced transmission delays. Additionally, the serial-input parallel-output shift register verifies the sequence numbers of subflits to confirm that they belong to the same flit, ensuring precise reconstruction for downstream transmission.

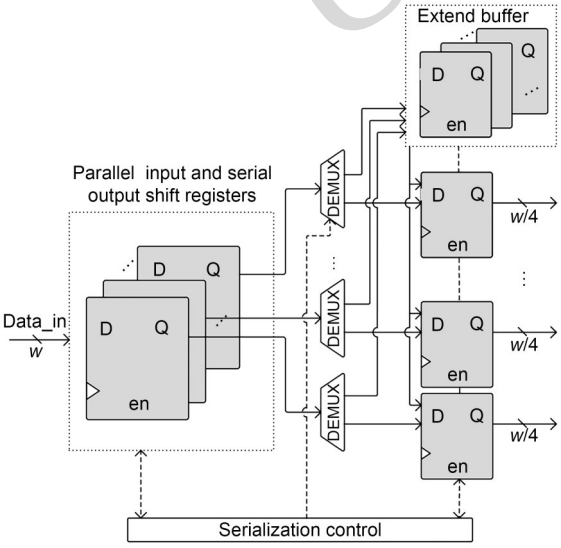

Fig. 8 Schematic representation of the deserialization architecture

Furthermore, to guarantee operational synchronization between the upstream and downstream routers during fault-tolerant transmission, HyRAS employs a robust coordination mechanism. While redundancy-based rerouting operates combinationally within a single cycle, serialization introduces multicycle delays. To manage this, data synchronization is strictly maintained via the unified serialization state table and serial counters at the sender, which dictate the TDMA sequence. Concurrently, the receiver verifies the sequence numbers of incoming subflits before reconstruction in the shift registers. Finally, to handle the bandwidth reduction and prevent data loss during serialized transmission, the routers seamlessly leverage the NoC's native credit-based flow control, applying backpressure to upstream routers whenever the deserializer's extend buffer is processing a delayed flit.

4.4 Fault-tolerant state transition

HyRAS employs two synergistic strategies to mitigate TSV faults across different granularities: RTSV switching for coarse-grained fault management and time-division serialization for fine-grained fault tolerance. The state transition diagram, depicted in Fig. 9, outlines the operational dynamics of HyRAS fault-tolerant mechanisms. This diagram defines four distinct states, described below, which guide the adaptive application of RTSV switching and time-division serialization in response to varying TSV fault conditions.

(1) Normal (N) state: In the N state, the 3D NoC operates without detected TSV faults, enabling standard router functionality without the need for fault-tolerant mechanisms.

(2) Fine-grained (F) state: The F state uses RTSV switching to address localized TSV faults. It reassigns signals from up to two faulty signal TSVs per block through MUX-based rerouting, maintaining connectivity with minimal overhead.

(3) Coarse-grained (C) state: In the C state, HyRAS employs time-division serialization to handle severe TSV faults when RTSVs are insufficient. Flits are segmented into subflits and transmitted over functional TSVs using TDMA, ensuring robust communication.

(4) Mixed (M) state: The M state integrates RTSV switching and time-division serialization to tackle complex TSV fault patterns. This state dynamically applies both fine-grained and coarse-grained fault-tolerant strategies, either within a single block or across multiple blocks, to optimize reliability and performance.

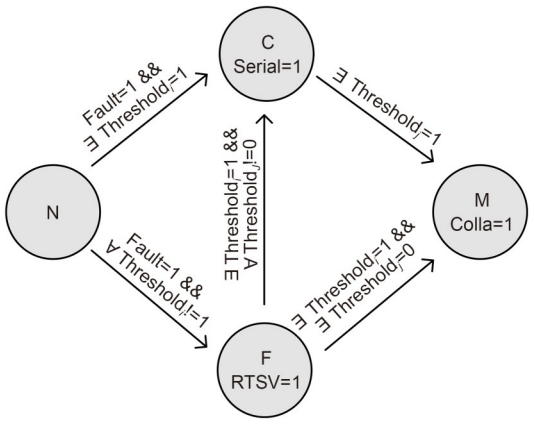

Fig. 9 Schematic representation of fault-tolerant state transition

When a fault is identified in a TSV link, the fault field is set to 1, and the Threshold$_i$ fields (i=0, 1, 2, and 3) indicate the fault-tolerant status of the i^{th} TSV block, enabling seamless state transitions within the HyRAS architecture. A Threshold$_i$ value of 1 denotes that the i^{th} TSV block contains more than two faulty STSVs, necessitating time-division serialization for fault mitigation. Conversely, a Threshold$_i$

value of 0 signifies two or fewer faulty STSVs, triggering RTSV switching to manage the faults. To execute the serialization and RTSV switching strategies outlined previously, TSV link fault handling is divided into three primary scenarios to ensure robust data transmission.

(1) F fault tolerance: When the number of faulty STSVs in a block is within the capacity of available RTSVs, HyRAS employs RTSV switching to mitigate faults without incurring latency overheads. This corresponds to the state transition from N to F in Fig. 9.

(2) C fault tolerance: When the number of faulty STSVs in a block exceeds the available RTSVs, HyRAS activates time-division serialization to manage faults, introducing latency penalties due to flit segmentation and TDMA-based transmission. This aligns with the state transition from N to C in Fig. 9.

(3) Transition from F to C fault tolerance: If the number of faulty TSVs in a block already operating under F fault tolerance surpasses the RTSV capacity, HyRAS transitions to time-division serialization to ensure continued fault coverage. This shift, accompanied by latency penalties, is reflected in the state transition from F to C in Fig. 9.

The transitions from C to M and from F to M states are elaborated below, highlighting the adaptive orchestration of fault-tolerant strategies across multiple TSV blocks in the HyRAS architecture. The M state signifies that at least two TSV blocks in the 3D NoC system are affected by faults, requiring a hybrid application of F and C fault-tolerant mechanisms to ensure reliable data transmission. The transition from F to M occurs when an additional TSV block encounters faults beyond its RTSV capacity, triggering the use of time-division serialization in the affected block, which shares functional TSV links with other blocks, as illustrated in the state transition diagram. Conversely, the transition from C to M takes place when another TSV block experiences faults within its RTSV capacity, allowing RTSV switching to address these faults while the original block persists with time-division serialization, thereby preserving link transmission functionality.

To attain a high repair rate for multiple clustered TSV faults in 3D NoC systems, HyRAS incorporates an advanced fault-tolerant repair algorithm seamlessly integrated with the state transition framework. This algorithm dynamically orchestrates RTSV switching and time-division serialization across fault-impacted blocks, ensuring robust and adaptive fault management.

4.5 TSV repair algorithm

In the preceding section, we described how HyRAS mitigates faulty TSVs by leveraging adjacent fault-free RTSVs and TSV blocks to ensure continuous connectivity and reliable operation. Algorithm 1 elucidates the fault state transitions for TSVs, starting from an initial fault-free N state and coordinating transitions to F, C, or M states based on fault conditions across TSV blocks.

The Fault field indicates whether any faults are detected across the entire TSV link, serving as the initial trigger for state transitions.

The $Fault_i$ field denotes whether faults are detected in the i-th TSV block (i=0, 1, 2, and 3), enabling block-specific fault monitoring within state transition logic.

The $Threshold_i$ field (i=0, 1, 2, and 3) indicates whether the number of faulty TSVs in the i^{th} TSV block exceeds the RTSV capacity, guiding the selection between RTSV switching and time-division serialization for fault tolerance.

Algorithm 1 begins by receiving fault status updates from the detection unit and updating the Fault and $Fault_i$ fields (i=0, 1, 2, and 3) to reflect the presence of faults across the TSV link and its four constituent blocks. Since each TSV link is divided into four blocks, when a fault is detected in the i^{th} block ($Fault_i$=1, i=0, 1, 2, and 3), Algorithm 1 evaluates the $Threshold_i$ field to decide whether to use RTSV switching or time-division serialization for fault tolerance.

Algorithm 1 TSV Repair Algorithm

Input: RTSVnum←0, Serialnum←0, $RTSVstate_i$←0, $Serialstate_i$←0 for all i=0,1,2, and 3, Fault state←N

Output: Fault state(N,C,F,M)

```
if Fault==0 then
  Fault state ←N;
end
else
  for i←0 to 3 do
    If Fault_i==1 and Threshold_i==0 then
      RTSVstate_i←1;
      RTSVnum←RTSVnum+1;
    end
    If Fault_i==1 and Threshold_i==1 then
      Serialstate_i←1;
      Serialnum←Serialnum+1;
      RTSVnum←RTSVnum−1;
    end
  end
  If Serialnum==4 then
    Error: abandon TSV link
  end
  else if 0<Serialnum<4 then
    for i←0 to 3 do
      if Serialstate_i==1 and RTSVnum== 0 then
        Fault state←C;
        Serial←1;
      end
      if Serialstate_i==1 and RTSVnum!= 0 then
        Fault state←M;
        Colla←1;
      end
    end
  else if Serialnum==0 then
    if RTSVnum!=0 then
      Fault state←F;
      RTSV←1;
    end
  end
end
```

The algorithm keeps track of fault counters for each vertical link, monitoring the number of defective TSVs in every block. Upon detecting a fault, it decides the next steps based on the fault severity and the availability of RTSVs, leading to a state transition. In the N

state, all TSVs are operational, allowing the link to function at full bandwidth. If a block experiences faulty TSVs, yet the number of faults does not surpass the available RTSVs (up to 2), it shifts into the F state, substituting faulty TSVs with RTSVs within the same block to preserve bandwidth (Algorithm 1, lines 30–35). When a block runs out of RTSVs and encounters new faults, it transitions to the C state, employing 1∶2 or 1∶4 TDMA serialization to limit bandwidth and maintain communication (Algorithm 1, lines 21–24). If some blocks in a link are in the F state while others are in the C state, the link adopts the M state, harmonizing various block configurations to enhance bandwidth distribution (Algorithm 1, lines 25–28). If all four blocks are beyond recovery (when more than three blocks require serialization), the link falls into the failure state, resulting in link abandonment. Finally, during the link management phase, the network configuration is updated according to the state transitions.

5 Experimental results

This section outlines the experimental setup designed to assess the fault-tolerant 3D NoC architecture of HyRAS. Next, we discuss the evaluation results, examining the performance metrics and overheads of the proposed method, substantiated through comparisons with established benchmarks.

5.1 Experimental setup

The HyRAS system was developed using Verilog-HDL and subjected to synthesis and prototyping with industry-standard computer-aided design (CAD) tools. To simulate realistic application scenarios, traffic was generated using the GEM5 simulator (Binkert et al., 2011) with PARSEC benchmarks (Bienia, 2011; Papaphilippou and Van Chu, 2024) and evaluated through a customized Access Noxim simulator (Catania et al., 2017), which models fault tolerance under horizontal link failure rates of up to 20%. Faults were introduced in both random and clustered patterns to test resilience. The simulations were performed in GEM5's full-system mode, emulating a 64-core ×86 architecture featuring four coherence directories, four 1-MB shared L2 caches, and 32-KB private L1 caches per core. This subsection compares the HyRAS architecture against the baseline, CETR, Securet3d, and SOTR architectures, all of which also address TSV faults. The experimental configuration parameters are detailed in Table 3.

Table 3 Experimental parameters

Parameter	Value
Voltage and frequency	1.2 V and 2.0 GHz
Channel width	64
Network size	5×5×5
Synthetic workloads	Uniform, transpose, bit-reversal, and shuffle
Number of VCs	2
Input buffer	8 flits depth
DSENT Technology model	Bulk45LVT process model
Horizontal link failures	5% and 15%

We compare the performance of HyRAS with the following four methods.

(1) Baseline: a standard 3D NoC architecture lacking fault-tolerant mechanisms, used as a reference for performance benchmarking.

(2) CETR (Ni et al., 2021): a cost-efficient architecture that uses shared RTSVs within TSV clusters to address clustered faults.

(3) Securet3d (da Silva et al., 2025): an adaptive routing algorithm designed for vertically partially connected 3D NoCs, which circumvents TSV faults by leveraging fault-free data paths to ensure secure and dependable communication.

(4) SOTR (Dang et al., 2020a): a scalable online algorithm that employs adaptive serialization techniques to recover from TSV-cluster defects, enhancing reliability.

5.2 Routing performance

The effectiveness of the fault-tolerant approach is evaluated using various synthetic traffic patterns in a simulated environment. The presented results are for a 5×5×5 mesh NoC, with routers that use TSV to maintain vertical communication. The traffic models used in this evaluation are uniform, bit-complement, and shuffle, which produce destination permutations based on corresponding bit manipulation operations on the destination address.

For verification of the proposed mechanisms, two important network performance metrics, average packet latency and throughput, are evaluated under synthetic traffic patterns. Fig. 10 presents the results of the network average latency with a 5% TSV defect rate under different traffic workloads. Baseline, CETR, Securet3d, and SOTR serve as comparative methods. As shown in Fig. 10, CETR, Securet3d,

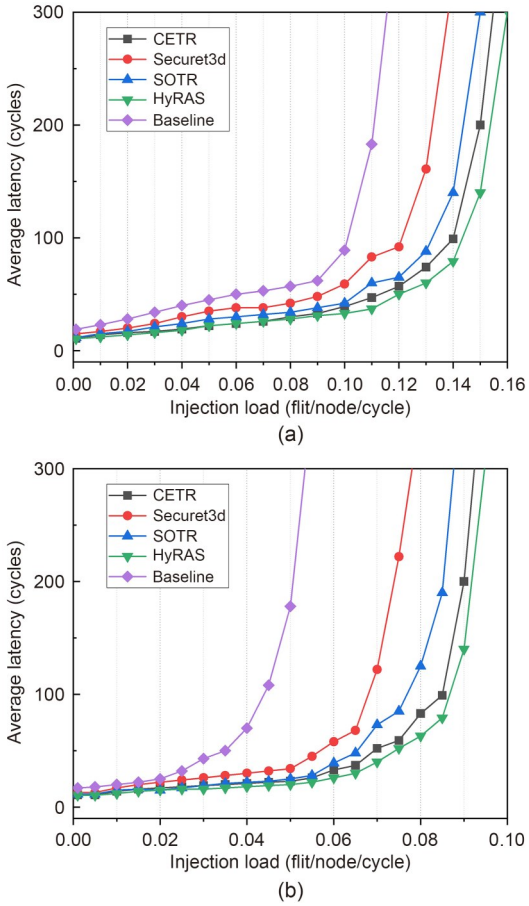

Fig. 10 Network average latency with a 5% TSV defect rate under different traffic workloads: (a) uniform; (b) bit-complement

SOTR, and HyRAS clearly outperform baseline. At low fault rates, RTSVs mitigate faults, enabling CETR to outperform SOTR. Furthermore, the strategies of CETR and SOTR surpass Securet3d, as Securet3d's algorithm bypasses faulty nodes without addressing the underlying faults. HyRAS demonstrates significantly better performance in all simulation conditions. The effectiveness of HyRAS can be attributed to its dynamic assignment of RTSVs and serialization. In the event of faults, HyRAS tolerates up to two TSV-bit line faults per group, using time-division multiplexing with other groups for handling more than two faults. CETR relies solely on shared RTSVs, becoming ineffective when faults exceed the available RTSVs. SOTR implements serialization through fault-free TSV blocks, effectively mitigating high fault counts. However, at low fault rates, serialization results in higher latency and decreased performance compared to redundancy-based fault tolerance.

Fig. 11 displays the latency results under a 15% TSV fault rate, where the elevated fault density reduces the saturation injection rate of the network. In contrast to low fault scenarios, SOTR surpasses CETR in Fig. 11, as higher fault rates undermine the efficacy of RTSV strategies, highlighting the strengths of serialization-based fault tolerance. HyRAS, the proposed approach, seamlessly combines RTSV allocation with serialization techniques, dynamically adjusting to diverse fault conditions to robustly address TSV faults.

Regarding throughput, Fig. 12 presents the results under a 5% TSV defect rate across various traffic workloads. HyRAS, along with CETR, Securet3d, and SOTR, demonstrates significant throughput

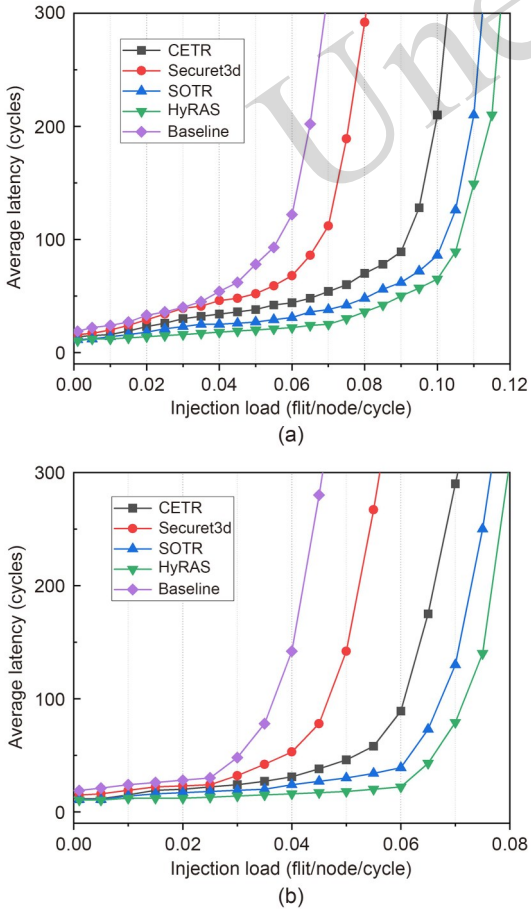

Fig. 11 Network average latency with a 15% TSV defect rate under different traffic workloads: (a) uniform; (b) bit-complement

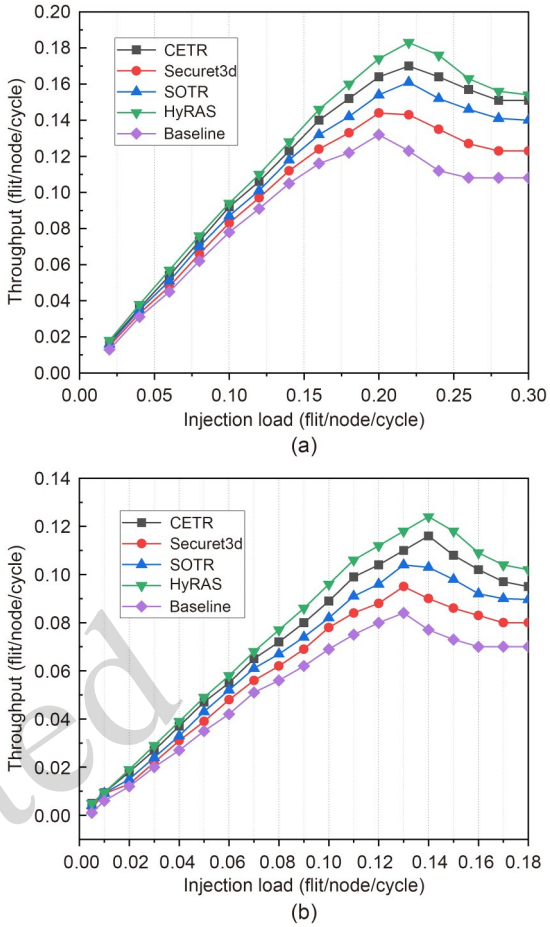

Fig. 12 Throughput with a 5% TSV defect rate under different traffic workloads: (a) uniform; (b) bit-complement

advantages over the baseline. At low fault rates, RTSVs effectively mitigate faults, enabling CETR to outperform SOTR. Moreover, the fault-tolerant strategies of CETR and SOTR surpass Securet3d, which bypasses faulty nodes without resolving the underlying faults. In bit-complement traffic, HyRAS achieves a peak throughput improvement of 42.5% over Securet3d, 6.9% over CETR, and 27.8% over SOTR, as shown in Fig. 12. HyRAS consistently delivers superior performance across all evaluated scenarios, driven by its dynamic combination of RTSV allocation and serialization-based fault tolerance. This contrasts with CETR's sole reliance on shared RTSVs and SOTR's serialization approach, which introduces higher latency at low fault rates.

Fig. 13 presents the throughput results under a 15% TSV fault rate, revealing a reduction in network saturation throughput due to increased fault density. In contrast to low fault scenarios, the SOTR outperforms the CETR at higher fault rates, as the efficacy of RTSV strategies wanes, underscoring the strengths of serialization-based fault mitigation. In bit-complement traffic mode, HyRAS's saturated throughput achieves a 47.7% improvement over Securet3d, a 32.7% enhancement compared to CETR, and a 12.1% gain relative to SOTR. Additionally, SOTR surpasses CETR under elevated fault conditions, leveraging its serialization approach to more effectively maintain link connectivity in the presence of faulty TSVs. HyRAS delivers the best performance by employing a hybrid strategy that dynamically combines RTSV allocation with serialization techniques,

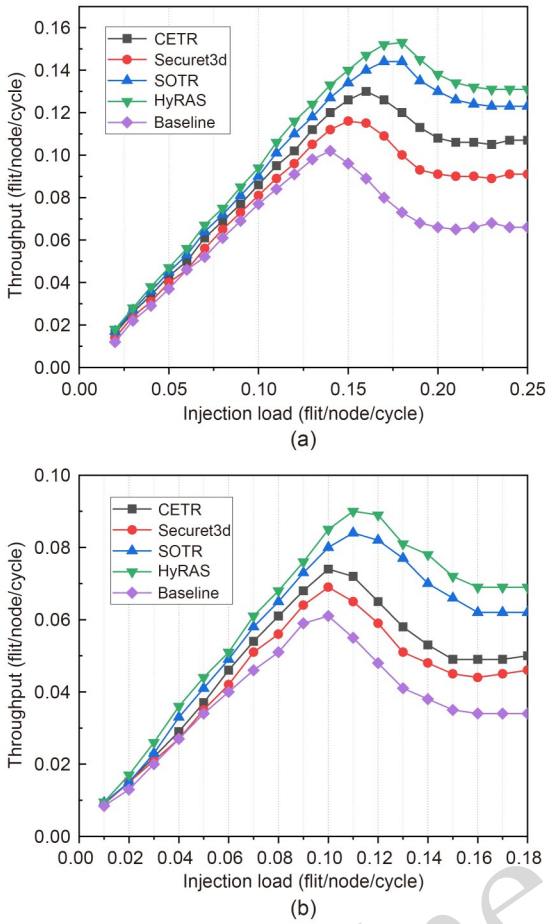

Fig. 13 Throughput with a 15% TSV defect rate under different traffic workloads: (a) uniform; (b) bit-complement

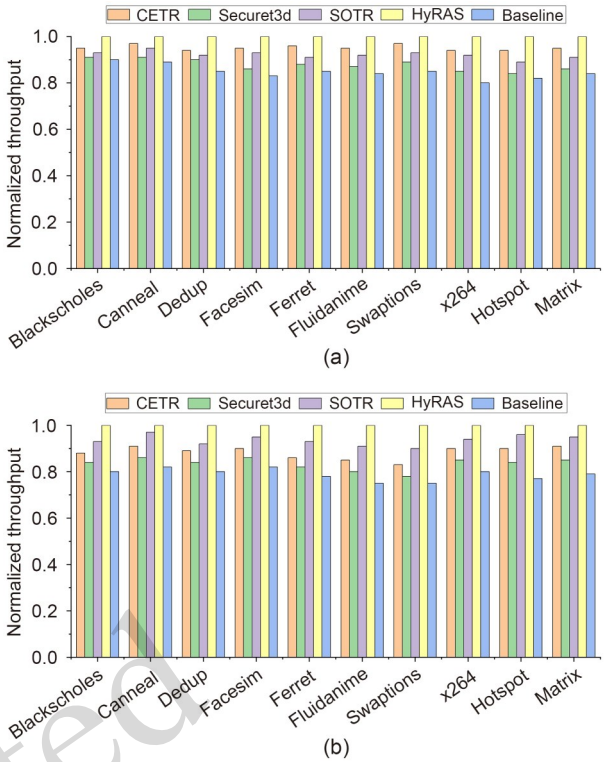

Fig. 14 Evaluation results of HyRAS for PARSEC benchmarks: (a) 5% TSV defect rate; (b) 15% TSV defect rate

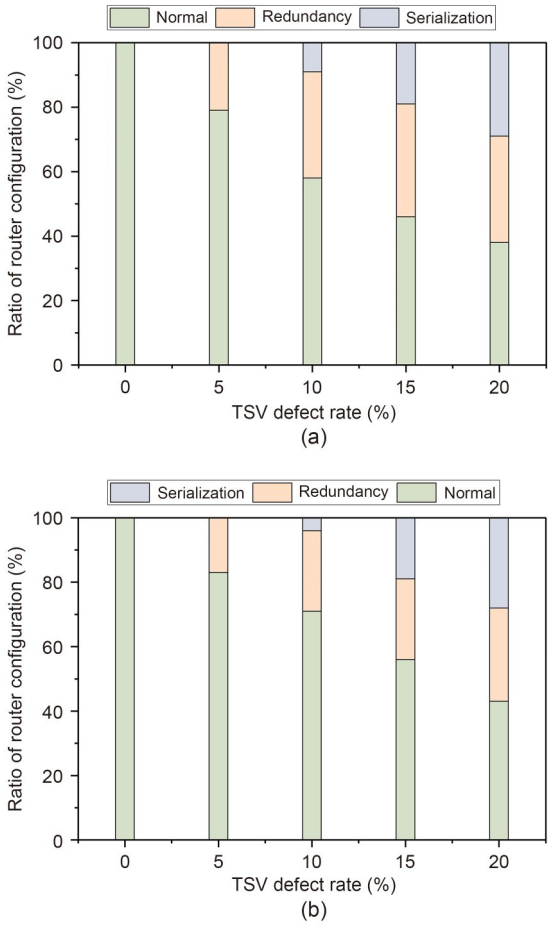

Fig. 15 Evaluation results of defect rate with 0% to 20%: (a) network size of 5×5×5; (b) network size of 8×8×8

enabling adaptable and efficient management of TSV faults across varying severity levels.

To complement synthetic workloads, we perform a parallel evaluation of the HyRAS fault-tolerant architecture using real-world benchmarks, specifically leveraging PARSEC (Dang et al., 2022) traces generated by the GEM5 simulator. Fig. 14 depicts the comparative results at 5% and 15% TSV defect rates. Across these traces, HyRAS consistently demonstrates competitive performance. A notable performance gap between CETR and SOTR is observed at both 5% and 15% TSV fault rates: CETR outperforms SOTR at lower fault rates, while SOTR excels at higher fault rates. At a 5% TSV defect rate, as shown in Fig. 14a, HyRAS achieves up to 19.1% higher throughput than Securet3d, 6.3% higher throughput than CETR, and 12.4% higher throughput than SOTR. At a 15% TSV defect rate, as illustrated in Fig. 14b, HyRAS leads with throughput improvements of up to 28.2% over Securet3d, 20.5% over CETR, and 11.2% over SOTR. The performance divergence between CETR and SOTR stems from the efficacy of RTSV strategies at lower fault counts, which diminishes at higher fault rates, in contrast to the superior effectiveness of serialization-based strategies under elevated fault conditions.

5.3 Defect rate evaluation

Furthermore, to address the scalability of the proposed architecture, Figs. 15a and 15b compare the router state distributions between 5×5×5 and 8×8×8 network sizes. TSVs are grouped as described in

Section 4, with defect rates varying from 0% to 20%. We measure the distribution of three router types within each layer: normal (fault-free or fully repaired), redundancy (using RTSVs for fault tolerance), and serialization (relying on serialization techniques), providing insight into the architecture's adaptability in 3D NoC systems. A critical observation is that the proportion of routers entering the temporal serialization state depends primarily on the fault density, remaining highly stable as the network size scales. Regarding performance scalability, the overhead introduced by serialization does not become prohibitive in larger networks (such as 8×8×8 or beyond) for two main reasons. First, HyRAS implements a fully distributed, link-level repair mechanism; thus, the hardware and control complexity per node remain constant ($O(1)$) irrespective of network dimensions. Second, while serialization inherently increases local transmission latency, the enhanced path diversity in larger NoC topologies allows adaptive routing algorithms to organically bypass heavily serialized and lower-bandwidth elevators. Consequently, the localized latency penalties of serialization are mitigated at the network level, ensuring that HyRAS maintains robust and scalable performance even under severe defect clustering in massive 3D-ICs.

The HyRAS architecture achieves outstanding performance through its robust fault tolerance framework, leveraging RTSVs and serialization techniques to ensure consistent connectivity despite faults. As fault rates increase, the proportion of serialization-based routers rises, while the share of redundancy-based routers levels off, maintaining stable operation. This dynamic adaptability underscores the high reliability of HyRAS architecture.

In conclusion, this evaluation underscores significant improvements in the reliability of the HyRAS architecture. Its resilient design ensures sustained vertical link connectivity across a wide spectrum of fault conditions. HyRAS outperforms the individual RTSV and serialization strategies used by CETR and SOTR, respectively, by integrating both approaches adaptively. Moreover, the evaluation confirms the architecture's ability to maintain efficient scalability. Extensive testing provides compelling evidence of the effectiveness and robustness of the proposed HyRAS approach.

5.4 Hardware Synthesis Analysis

Table 4 provides a comprehensive analysis of the hardware complexity for the HyRAS router (detailing area and power metrics), including static, dynamic, and total power consumption. Compared to the redundancy model of the router, which incorporates the proposed techniques, HyRAS exhibits a modest increase in area and power consumption of 14.52% and 8.86%, respectively. Relative to the baseline model, the HyRAS system approximately doubles both area and power costs. However, the TSV redundancy and serialization modules introduce manageable overheads, accounting for 13.05% and 12.69% of the total router metrics, respectively. Notably, the TSV redundancy module effectively handles a limited number of TSV faults, while the serialization module ensures continuous communication between routers. By integrating redundancy and serialization, HyRAS significantly enhances network reliability while maintaining acceptable area and power consumption overheads. As highlighted earlier, serialization becomes particularly critical in high-defect-rate environments to sustain robust connectivity.

Table 4 Hardware complexity breakdown of a single router

Model		Area (μm^2)	Power (mW)		
			Static	Dynamic	Total
Baseline router		28 873	7.62	1.34	8.96
Proposal	Router	30 052	8.42	1.55	9.97
	Redundancy	2151	0.58	0.11	0.69
	Serialization	2361	0.64	0.12	0.76
	Total	34 564	9.64	1.78	11.42

6 Conclusion

In this paper, we present HyRAS, a fault-tolerant 3D NoC architecture engineered to address TSV faults during both manufacturing and runtime phases. The core innovation of HyRAS lies in its hybrid fault-tolerant approach, which seamlessly combines RTSV allocation with serialization-based techniques to adapt dynamically to diverse fault rates while maintaining deadlock-free operation. Our evaluations demonstrate that HyRAS delivers superior performance. Simulation results validate that HyRAS ensures zero packet loss across TSV fault rates from 0% to 20%. Additionally, HyRAS's flexible TSV clustering configuration imposes no positional restrictions, providing designers with the versatility to optimize link placement tailored to specific application requirements.

In conclusion, HyRAS represents a significant leap forward in reliable and scalable 3D NoC design, delivering outstanding fault tolerance and adaptable performance for mission-critical applications.

Acknowledgments

This work was supported by the Doctoral Foundation of Fuyang Normal University (No. 2023KYQD0053), the University Natural Science Research Project of Anhui Province (No. 2024AH051464), the Open Fund of the Anhui Engineering Research Center for Intelligent Computing and Information Innovation (No. ICII202506), and the Key Research Project of Fuyang Normal University (No. 2025FSKJ48).

Author contributions

Chenglong SUN designed the research. Yanqing ZHOU and Yan ZHANG processed the data. Qi WANG drafted the paper. Chenglong SUN helped organize the paper. Chenglong SUN revised and finalized the paper..

Conflict of interest

All the authors declare that they have no conflict of interest.

Data availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

Declaration on the use of generative AI tools

During the preparation of this work, the authors used Gemini to improve language. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the published article.

References

Agarwal S, Goel K, Sinha M, et al., 2025. Mitigation of phase transitions in self-organizing NoC for stable queueing dynamics. *IEEE Trans Comput*, 74(2):623-636. https://doi.org/10.1109/tc.2024.3500373

Akbari S, Shafiee A, Fathy M, et al., 2012. AFRA: a low cost high performance reliable routing for 3D mesh NoCs. Proc Design, Automation & Test in Europe Conf & Exhibition, p.332-337. https://doi.org/10.1109/DATE.2012.6176490

Asadboland M, Mehranzadeh A, Mosleh M, 2025. CTWR: a congestion, temperature and wear-aware routing algorithm for partially-connected 3D network-on-chip. *Comput Electr Eng*, 124:110421. https://doi.org/10.1016/j.compeleceng.2025.110421

Bienia C, 2011. Benchmarking Modern Multiprocessors. PhD Dissemination, Princeton University, Princeton, USA.

Binkert N, Beckmann B, Black G, et al., 2011. The gem5 simulator. *ACM SIGARCH Comput Archit News*, 39(2):1-7. https://doi.org/10.1145/2024716.2024718

Bose A, Ghosal P, 2020. A low latency energy efficient BFT based 3D NoC design with zone based routing strategy. *J Syst Archit*, 108:101738. https://doi.org/10.1016/j.sysarc.2020.101738

Catania V, Mineo A, Monteleone S, et al., 2017. Cycle-accurate network on chip simulation with noxim. *ACM Trans Model Comput Simul*, 27(1):4. https://doi.org/10.1145/2953878

Chen S, Xu Q, Yu B, 2019. Adaptive 3D-IC TSV fault tolerance structure generation. *IEEE Trans Comput-Aided Des Integr Circ Syst*, 38(5):949-960. https://doi.org/10.1109/TCAD.2018.2824284

da Silva AA, Nogueira L, Coelho A, et al., 2025. Securet3d: an adaptive, secure, and fault-tolerant aware routing algorithm for vertically–partially connected 3D-NoC. *IEEE Trans Very Large Scale Integr Syst*, 33(1):275-287. https://doi.org/10.1109/tvlsi.2024.3500575

Dang KN, Ahmed AB, Okuyama Y, et al., 2020a. Scalable design methodology and online algorithm for TSV-cluster defects recovery in highly reliable 3D-NoC systems. *IEEE Trans Emerg Top Comput*, 8(3):577-590. https://doi.org/10.1109/TETC.2017.2762407

Dang KN, Ahmed AB, Abdallah AB, et al., 2020b. TSV-OCT: a scalable online multiple-TSV defects localization for real-time 3-D-IC systems. *IEEE Trans Very Large Scale Integr Syst*, 28(3):672-685. https://doi.org/10.1109/tvlsi.2019.2948878

Dang KN, Ahmed AB, Abdallah AB, et al., 2022. HotCluster: a thermal-aware defect recovery method for through-silicon-vias toward reliable 3-D ICs systems. *IEEE Trans Comput-Aided Des Integr Circ Syst*, 41(4):799-812. https://doi.org/10.1109/tcad.2021.3069370

Dubois F, Sheibanyrad A, Petrot F, et al., 2013. Elevator-first: a deadlock-free distributed routing algorithm for vertically partially connected 3D-NoCs. *IEEE Trans Comput*, 62(3):609-615. https://doi.org/10.1109/tc.2011.239

Flich J, Duato J, 2008. Logic-based distributed routing for NoCs. *IEEE Comput Archit Lett*, 7(1):13-16. https://doi.org/10.1109/L-CA.2007.16

Fu YX, Zhang C, Song WQ, et al., 2021. Optimizing vertical link placement and congestion aware dynamic elevator assignment for partially connected 3D-NoCs. *IEEE Trans Comput-Aided Des Integr Circ Syst*, 40(10):1957-1970. https://doi.org/10.1109/tcad.2020.3038338

Hou KH, Fan ZW, Zhang SF, et al., 2024. Dimension influence on the interface fatigue characteristics of three-dimensional TSV array: a fully coupled thermal-electrical-structural analysis. *IEEE Trans Dev Mater Reliab*, 24(4):571-583. https://doi.org/10.1109/TDMR.2024.3453923

Hsieh AC, Hwang T, 2012. TSV redundancy: architecture and design issues in 3-D IC. *IEEE Trans Very Large Scale Integr Syst*, 20(4):711-722. https://doi.org/10.1109/tvlsi.2011.2107924

Jiang L, Xu Q, Eklow B, 2013. On effective through-silicon via repair for 3-D-stacked ICs. *IEEE Trans Comput-Aided Des Integr Circ Syst*, 32(4):559-571. https://doi.org/10.1109/TCAD.2012.2228742

Kang U, Chung HJ, Heo S, et al., 2010. 8 Gb 3-D DDR3 DRAM using through-silicon-via technology. *IEEE J Solid-State Circ*, 45(1):111-119. https://doi.org/10.1109/JSSC.2009.2034408

Khalil K, Eldash O, Kumar A, et al., 2021. Self-healing router approach for high-performance network-on-chip. *IEEE Open J Circ Syst*, 2:485-496. https://doi.org/10.1109/OJCAS.2021.3095000

Khalil K, Kumar A, Bayoumi M, 2024. Dynamic fault tolerance approach for network-on-chip architecture. *IEEE J Emerg Sel Top Circ Syst*, 14(3):384-394. https://doi.org/10.1109/jetcas.2024.3438250

Kirtonia P, Williams S, Akter S, et al., 2026. A novel TSV model with fault characterization for high-frequency transmission in 3D ICs. *IEEE Trans Circ Syst I Reg Pap*, 73(3):1742-1755. https://doi.org/10.1109/TCSI.2025.3628723

Lee H, Shin SH, Yoo Y, et al., 2023. TRUST: through-silicon via repair using switch matrix topology. *IEEE Trans Comput-Aided Des Integr Circ Syst*, 42(7):2377-2390. https://doi.org/10.1109/tcad.2022.3220711

Liu C, Chu C, Xu DW, et al., 2022. HyCA: a hybrid computing architecture for fault-tolerant deep learning. *IEEE Trans Comput-Aided Des Integr Circ Syst*, 41(10):3400-3413. https://doi.org/10.1109/TCAD.2021.3124763

Lung CL, Chien JH, Shi YY, et al., 2011. TSV fault-tolerant mechanisms with application to 3D clock networks. Proc Int SoC Design Conf, p.127-130. https://doi.org/10.1109/ISOCC.2011.6138663

Maity DK, Roy SK, Giri C, 2021. TSV-cluster defect tolerance using tree-based redundancy for yield improvement of 3-D ICs. *IEEE Trans Comput-Aided Des Integr Circ Syst*, 40(8):1500-1510. https://doi.org/10.1109/tcad.2020.3021341

Mercier R, Killian C, Kritikakou A, et al., 2022. BiSuT: a NoC-based bit-shuffling technique for multiple permanent faults mitigation. *IEEE Trans Comput-Aided Des Integr Circ Syst*, 41(7):2276-2289. https://doi.org/10.1109/tcad.2021.3101406

Ni TM, Liu DS, Xu Q, et al., 2020. Architecture of cobweb-based redundant TSV for clustered faults. *IEEE Trans Very Large Scale Integr Syst*, 28(7):1736-1739. https://doi.org/10.1109/TVLSI.2020.2995094

Ni TM, Xu Q, Huang ZF, et al., 2021. A cost-effective TSV repair architecture for clustered faults in 3-D IC. *IEEE Trans Comput-Aided Des Integr Circ Syst*, 40(9):1952-1956. https://doi.org/10.1109/TCAD.2020.3025169

Niazmand B, Azad SP, Flich J, et al., 2016. Logic-based implementation of fault-tolerant routing in 3D network-on-chips. Proc 10th IEEE/ACM Int Symp on Networks-on-Chip, p.1-8. https://doi.org/10.1109/NOCS.2016.7579317

Ouyang YM, Zhang TB, Li JH, et al., 2024. Fault-tolerant routing for reliable packet transmission in on-chip networks. *Microelectr J*, 153:106425. https://doi.org/10.1016/j.mejo.2024.106425

Papaphilippou P, Van Chu T, 2024. Efficient deadlock avoidance for 2-D mesh NoCs that use OQ or VOQ routers. *IEEE Trans Comput*, 73(5):1414-1426. https://doi.org/10.1109/TC.2024.3365954

Reddy RP, Acharyya A, Khursheed S, 2017. A cost-effective fault tolerance technique for functional TSV in 3-D ICs. *IEEE Trans Very Large Scale Integr Syst*, 25(7):2071-2080. https://doi.org/10.1109/TVLSI.2017.2681703

Song RH, Zhang JQ, Zhu ZQ, et al., 2024. Fault and self-repair for high reliability in die-to-die interconnection of 2.5D/3D IC. *Microelectr Reliab*, 158:115429. https://doi.org/10.1016/j.microrel.2024.115429

Taheri E, Kim RG, Nikdast M, 2023. AdEle+: an adaptive congestion-and-energy-aware elevator selection for partially connected 3D networks-on-chip. *IEEE Trans Comput*, 72(8):2278-2292. https://doi.org/10.1109/TC.2023.3248260

Wang SC, Chakrabarty K, Tahoori MB, 2019. Defect clustering-aware spare-TSV allocation in 3-D ICs for yield enhancement. *IEEE Trans Comput-Aided Des Integr Circ Syst*, 38(10):1928-1941. https://doi.org/10.1109/TCAD.2018.2864291

Wei C, Cui XL, Cui XX, 2024. Dy-MFNS-CAC: an encoding mechanism to suppress the crosstalk and repair the hard faults in rectangular TSV arrays. *IEEE Trans Reliab*, 73(1):622-636. https://doi.org/10.1109/tr.2023.3303189

Xiong RT, Ren W, Zhang CZ, et al., 2025. A sampling-based acceleration method for heterogeneous chiplet NoC simulations. *Future Gener Comput Syst*, 166:107643. https://doi.org/10.1016/j.future.2024.107643

Xu Q, Geng H, Ni TM, et al., 2022. Fortune: a new fault-tolerance TSV configuration in router-based redundancy structure. *IEEE Trans Comput-Aided Des Integr Circ Syst*, 41(10):3182-3187. https://doi.org/10.1109/tcad.2021.3133484

Zhang Y, Jing ZW, Yang QH, et al., 2025. A survey on routing algorithm and router microarchitecture of three-dimensional network-on-chip. *J Syst Archit*, 164:103429. https://doi.org/10.1016/j.sysarc.2025.103429