R2CTA: Reinforcement Learning and Reservoir Computing based Chiplets TSV Assignment

Xiaomeng Wang and Yang Yi
Bradley Department of Electrical and Computer Engineering
Virginia Tech
Blacksburg, US
wxm@vt.edu, yangyi8@vt.edu

Zhen Zhou
Photonics Research Lab, IDP
Intel Labs
Santa Clara, US
zhen1.zhou@intel.com

***Abstract*—Chiplets based design is gaining more attention in academia because of its success in industry. Its unique structure poses new challenges in the field of physical design optimization. Although many floorplan and TSV assignment algorithms are proposed for dies and 3DICs design optimization, new studies are needed to address the challenges of chiplet systems. With the recent success of artificial intelligence, algorithms should be developed to optimize the design of chiplet systems. This paper proposes a novel approach to the TSV assignment problem in heterogeneous chiplet systems using reinforcement learning (RL) and reservoir computing (RC), coined R2CTA: Reinforcement Learning based Reservoir Computing Chiplet Assignment. R2CTA aims to optimize key metrics of heterogeneous chiplet systems by fine tuning the TSV assignment.**

***Index Terms*—Reinforcement Learning, Reservoir Computing, Echo State Network, Chiplets, TSV Assignment, Heterogeneous Integration, Wirelength Optimization**

I. Introduction

As Moore's law slows down, the integration of heterogeneous chiplets has become a promising approach to enhance system performance with economic benefits for state-of-the-art systems as well as for next generation architectures [1]. To achieve high-performance and low-latency systems, elements such as through-silicon via (TSV) are indispensable. TSV enables the integration of chiplets with different functionalities into a single package, but also introduces significant routing congestion and thermal challenges.

The TSV assignment problem has been studied extensively especially in the field of 3DIC, a few studies use advanced machine learning algorithms like deep reinforcement learning for 3DIC wirelength and thermal optimization. Although 3DIC and certain chiplet systems share the same laminated structure, chiplet systems have additional die placement structures, as well as distinctive characteristics including multiple dies placement on the same level with lateral connections. It should be noted that although research has been done for floorplan and TSV assignment on 3DIC designs, no previous work has been done in the field of chiplet systems using reinforcement learning. With the recent success of reinforcement learning and machine learning algorithms, it is imperative to develop new machine learning based algorithms for chiplet systems design optimization.

For supplementary materials and additional information about the **R2CTA** framework, please visit **`www.wangxm.com/Research/R2CTA`**.

However, the transition to chiplet-based design, while promising, introduces significant challenges in physical design optimization. Among these challenges, TSV assignment and modules floorplan optimization are particularly critical and need to be prioritized, as they directly impact the overall system performance, power efficiency, and manufacturing cost. Other challenges include managing cross-die power and thermal constraints, and ensuring reliable high-speed inter-chiplet communication are also impacted by the high level design optimization.

In this paper, a novel approach named R2CTA(pronounced as "R-two-see-tah"): ***Reinforcement Learning and Reservoir Computing based Chiplets TSV Assignment*** is proposed to optimize the TSV assignment for chiplets based design using reinforcement learning. R2CTA is proposed to capture the dynamic nature of the chiplet systems TSV assignment problemwith the help of reservoir computing, specifically echo state network(ESN) [2].

The rest of the paper is organized as follows: section II introduces the basic concepts of chiplet systems and the challenges they pose; section III describes the proposed R2CTA algorithm in detail, including the multi-agent reinforcement learning framework, echo state network integration, and expanded action space design; section IV presents comprehensive experimental results on various benchmarks, comparing R2CTA with state-of-the-art approaches; section V concludes the paper and discusses potential future research directions.

In summary, this paper presents R2CTA, a novel approach combining reinforcement learning and reservoir computing to optimize TSV assignment in chiplet-based designs. By addressing the unique challenges of chiplet systems through echo state networks and multi-agent reinforcement learning, our method achieves 3x improvement in wirelength reduction compared to existing approaches.

II. Background

The semiconductor industry's shift from monolithic designs to chiplet-based architectures introduces new challenges in physical design optimization, particularly in die-to-die communication and thermal management. These challenges

require innovative solutions beyond traditional optimization methods. Recent advances in machine learning, specifically reinforcement learning and reservoir computing, offer promising approaches to tackle these complex optimization problems. Reinforcement learning enables automated decision-making through environment interaction, while reservoir computing provides efficient processing of temporal data with reduced computational overhead, making them particularly suitable for chiplet-based design optimization. Given the success of these algorithms in other domains, they are worth exploring to chiplet system optimization to address its unique challenges and complexity.

A. Chiplets and Advanced Packaging Technology

Chiplets based design relies on advanced packaging technology for integration. It is different from traditional packaging technology in the sense that the interconnection between dies are highly demanded. For advanced packaging technology, it could generally be categorized as follows which is defined in [3]: **2.1D**, **2.3D**, **2.5D**, and **3D**. Each types of packaging technology has its own pros and cons, for the purpose of this paper, we will focus on the 2.5D and 3D technologies since they are the most promising technologies for future chiplet systems.

Through Silicon Vias (TSVs) are one of the most critical components for chiplets based design, they enable the vertical electrical and thermal connections between dies. This indicates that TSVs placement is a critical aspect for chiplets based design optimization. According to [4], TSVs could be classified into three categories: *via-first*, *via-middle*, and *via-last*. In this work, for maximum flexibility, TSVs are considered as via-first for the assignment problem using reinforcement learning.

B. Reinforcement Learning

Reinforcement learning, particularly deep reinforcement learning [5] [6], has recently gained popularity in the machine learning community due to its ability to learn from the environment and make decisions based on the feedback from the environment. It starts to be applied to the field of physical design optimization [7] [8].

In this work, one variant of Multi Agent Deep Deterministic Policy Gradient (MADDPG) [9] is used as the reinforcement learning algorithm, namely ATT-MADDPG [10]. MADDPG is a type of reinforcement learning algorithm that extends the standard deep deterministic policy gradient (DDPG) algorithm [11] to handle multiple agents in a cooperative or competitive environment. MADDPG is designed to address the challenges of multi-agent systems, where the actions of one agent can significantly impact the rewards and observations of other agents. ATT-MADDPG elevates the performance of MADDPG by adding attention mechanism to the policy network. It is suitable for the scenario where the agents that have distinct features collaborate to achieve a common goal. Chiplet based designs fit perfectly into this scenario where each agent is responsible for one die. Dies within the design has features including but not limited to physical size, technology nodes, placement that are more diverse among dies comparing with the traditional 3DIC counterpart.

C. Reservoir Computing

For VLSI especially chiplets based design optimization with a large search space, neural networks including multi-layer perceptron (MLP) requests large amount of computing resources as well as training dataset. Since the training dataset especially for chiplets based design are either not publicly available due to proprietary reasons or synthesized manually that are not representative, it is challenging to train a deep neural network for chiplets based design optimization. Due to the above reasons, alternative machine learning algorithms are needed to overcome the computational burden and scarcity of training dataset.

Reservoir computing [12] [13] is a novel paradigm in machine learning and neural network theory that offers a unique approach to processing temporal data. At its core, reservoir computing utilizes a fixed, randomly connected network of neurons (the "reservoir") to transform input signals into a higher-dimensional space. This reservoir acts as a complex, nonlinear dynamic system that can capture and represent intricate temporal patterns in the input data. The most prominent implementations of reservoir computing include Echo State Networks (ESN) [2] and Liquid State Machines (LSM) [14].

In this work, we incorporate echo state networks (ESN) [2] alongside fully connected neural networks to better capture the dynamic nature of chiplet-based designs. ESN, a type of reservoir computing, processes sequential data efficiently through a reservoir of neurons with fixed random weights and connections. The network output is computed as a linear combination of reservoir states, requiring training only for the output layer. This approach simplifies training while better capturing temporal dynamics compared to traditional fully connected networks. The ESN architecture is illustrated in fig. 1.

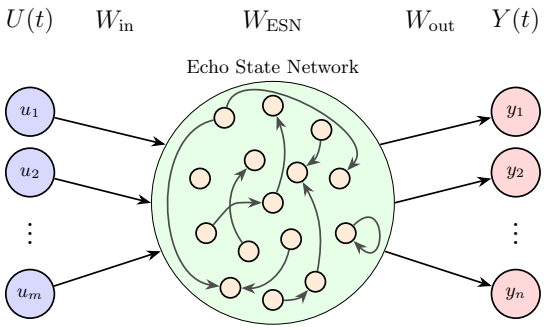

Fig. 1. Echo State Network Architecture [15]

III. Methodology

There are many research works [16] about TSV assignment for dies and 3DICs optimization. Those algorithms typically use techniques like simulated annealing [17], genetic algorithm

[18]. Only one recent research namely ATT-TA(Attention-based Thermal-aware TSV Assignment) [19] uses reinforcement learning which is considered as advanced machine learning algorithm to optimize the TSV assignment, it is considered as the baseline for this research.

However, existing works do not address the unique challenges of chiplet-based designs, which present significantly higher complexity compared to traditional dies and 3DICs. In this work, a novel approach is proposed to optimize the TSV assignment for chiplets based design using reinforcement learning where certain situations are not covered by existing works.

A. Optimization Objective

While chiplet system design optimization encompasses multiple aspects including electrical, thermal, and mechanical considerations, in this paper, we focus specifically on interconnect wirelength minimization as the primary optimization objective. This focused approach allows for clearer comparison with existing methods and benchmarks.

The optimization objective could be written as:

$$\mathcal{O}\left(\mathcal{S}\left(t\right)\right) = \sum_{i=1}^{N} w_i \cdot \mathcal{O}_i\left(\mathcal{S}\left(t\right)\right) \tag{1}$$

where $\mathcal{S}\left(t\right)$ is the complete multi-tiers TSV assignment solution at time step t. Totally N metrics are considered for the optimization in the reinforcement learning framework, $\mathcal{O}_i\left(\mathcal{S}\left(t\right)\right)$ is the metric i to be optimized and w_i is the weight associated with metric i for optimization. $\mathcal{O}\left(\mathcal{S}\left(t\right)\right)$ is the objective function to be optimized, in the TSV assignment problem, the overall optimization objective is the sum of various metrics, the goal is to find the optimal TSV assignment that minimizes the objective function. Here are some of the most commonly used metrics for optimization in the literature:

1) Wirelength Calculation: Wirelength calculation is a crucial metric for chiplet placement optimization. fig. 2 illustrates our wirelength calculation approach using Manhattan distance metrics. In the figure, maroon circles represent module IOs and TSVs within the same net, connected by orange wires. Since IOs and TSVs within the same net are distributed across a die, we employ a minimal spanning tree algorithm [20] [21] to establish connections with minimum total wirelength. In chiplet-based heterogeneous integration, TSVs facilitate cross-die connections, ensuring IOs within the same net are properly connected. During the TSV assignment process, we use Manhattan distance to estimate wirelength and evaluate placement quality, without considering detailed routing constraints to reduce computational complexity.

2) Thermal Calculation: Thermal performance is also one key aspect to be considered in the chiplet based design. Although there are thermal TSVs designed specifically to minimize the hotspot issue, signal TSVs also contribute to mitigate the thermal issue. One of the most widely used open source thermal simulation tool is called *Hotspot* [22]. Although it could provide detailed thermal analysis, it requires long time to run simulation. Another faster thermal simulation tool is called power blurring [23]. It is a simple thermal simulation tool that uses convolution to approximate the temperature distribution with respect to the power assignment.

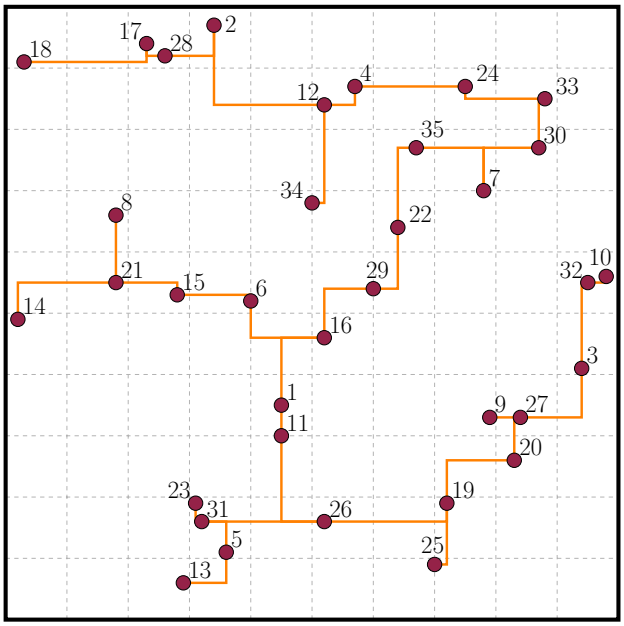

Fig. 2. Illustration of Minimal Spanning Trees (MST) with Manhattan Distance

In this work, only the wirelength metric is considered for the optimization, as this setup could simplify the performance evaluation and the comparison with related research works. In addition, chiplets based design thermal analysis is more complex than traditional 3DIC counterpart, which indicates more research works are needed to be done in this area, thus it is beyond the scope of this paper.

B. Reinforcement Learning based TSV Assignment

Reinforcement learning is used to handle the tsv assignment problem for chiplets based design. The overall algorithm is summarized as follows in algorithm 1.

The remaining of this section will discuss more details about how reinforcement learning is applied to the TSV assignment problem for chiplets based design.

1) Agent: Considering chiplets based design is more complex than 3DIC counterpart, multi-agent reinforcement learning framework needs to be adjusted for the new design methodology. Each agent is responsible for **one tier** within the chiplet system, and the reward function is designed to encourage the agents to collaborate to optimize defined optimization objective.

The ATT-MADDPG framework is used such that multiple agents work together to achieve the global optimization objective, the algorithm follows the centralized training and decentralized execution paradigm such that the global information is available for the agents to make decisions. Each agent's actor is responsible for one tier with its own feature set such that it

Algorithm 1 R2CTA: Reservoir Computing and Reinforcement Learning based Chiplets TSV Assignment

Require: Initial chiplet system with TSV assignment and multi-agent reinforcement learning framework
Ensure: Optimized TSV assignment for chiplet system

```
Initialize TSV Assignment environment, RL framework including Echo State Network
while current episode ⩽ max episode do
    while current step ⩽ max steps per episode do
        for each agent do
            Observe current state from environment
            Select action based on actor network output
        end for
        Execute TSV perturbation options in environment
        Receive reward based on optimization metrics
        Store experience in replay buffer
        for each agent do
            Sample mini-batch from replay buffer
            Update critic networks and actor networks
        end for
        current step ← current step + 1
    end while
    Reset Echo State Network states in critic networks
    if Update Target Network Flag is True then
        Update Target Critic Networks
        Update Target Actor Networks
    end if
    current episode ← current episode + 1
end while
return Optimized policy for TSV assignment
```

is not overwhelmed by the current and historical information from other tiers. For critics, they are responsible for evaluating the overall performance of the chiplet system such that the actors could learn from the global information. To further improve the performance of the algorithm, echo state network is utilized as the input layer such that information from all agents will be projected to a high level dimension with the reservoir. It is able to capture the sequential information better that is provided by the environment including the historical information as features in [19].

2) Environment: The environment is designed to simulate the chiplet system and provide each agent with the necessary information to interact with the environment. The input features contains various hierarchical levels of information, including tier level info, module level info where detailed information for specific tsv are given, finally historical information is also provided.

It's worth mentioning that [19] provides a detailed table of features that are fed into the reinforcement learning architecture, which are similar to the features proposed earlier in [24]. However, those features mentioned explicitly are not sufficient during training, as they cause the objective function to be unstable or oscillating during training phase for the multi-agent reinforcement learning framework. To address this issue, new features are added to overcome the instability of the objective function. The following table shows the features that are used in this work.

TABLE I
FEATURES FOR THE REINFORCEMENT LEARNING FRAMEWORK

Feature	Range	Description
$\epsilon(s)$	$[-1, 1]$	Current cost
$\epsilon(s')$	$[-1, 1]$	Sampled Neighbours Cost
$\epsilon(s^\star)$	$[-1, 1]$	Lowest Cost by far
$\overline{\epsilon}$	$[-1, 1]$	Average cost
$\overline{\epsilon^\star}$	$[-1, 1]$	Average cost since $s^\star$
$\epsilon'_\star$	$[-1, 1]$	lowest cost of sampled neighbours
n_{nets}	$[0, 1]$	normalized affected nets percentage
t	$[0, 1]$	episode progress
$p(s^\dagger)$	$[0, 1]$	percentile of nominated neighbour location cost in sampled neighbours around selected TSV
$\hat{\mathcal{O}}_i(s)$	$[0, 1]$	normalized metric i of the current state within the die

From table I, features that are used in [19] are shown the first 8 rows, with feature name, range, and description. $p(s^\dagger)$ is the percentile of nominated neighbour location cost in sampled neighbours around a selected TSV, this additional feature is added to address the instability of the objective function during training phase. For ATT-TA, each agent would decide whether to move a TSV to the nominated location or remain in its current position. In addition, for R2CTA, $\hat{\mathcal{O}}_i(s)$ which is the normalized metric i of the current state within the die is added to improve the performance of the reinforcement learning algorithm. In this work, the normalized metric i is the normalized wirelength of a net in which the selected TSV is located within the die.

3) Action Space: Each agent's actor network would generate probability distribution of actions for the TSV assignment, the action space is designed to allow an TSV to move around its neighbor positions to achieve the optimization objective. [19] presents an action space of 2 actions, where the actor network decides whether to move or swap a TSV to the randomly selected nominated location or remain in its current position. The nominated action types include moving a TSV to a neighbouring location or swapping a TSV with another neighbouring TSV. The neighbours are categorized based on the relative distance to the TSV from 1 to 3 units, which is typically in μm.

In this work, one new action space for the TSV assignment is proposed, which provides a TSV with more perturbation options in one step. In addition to move around the neighbour positions with maximum distance of 5 units on x or y axis, moving from one empty space to another within the same die is also available for exploration purpose. This method allows the search space to be expanded and the performance of the algorithm to be improved.

Each step, a randomly selected TSV will be moved to the following locations according to table II based on each agent's actor network decision.

In fig. 3, the visualization of the TSV movement categories is shown, excluding the movement across empty space within a die.

TABLE II
TSV PURTURBATION MOVEMENT CATEGORIES FOR R2CTA

Category	Distance	Description
1	1-2 units	Immediate neighbours
2	3 units	Close proximity movement area
3	4-10 units	Region above square diagonal
4	4-10 units	Region below square diagonal
5	Variable	Move across empty space within a die
6	0	TSV stays the same

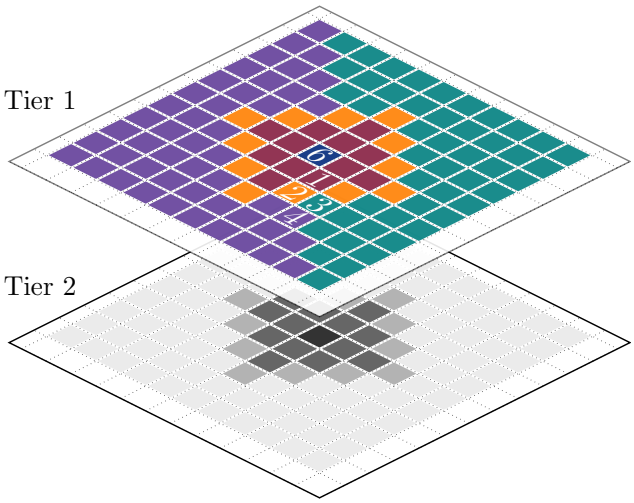

Fig. 3. TSV Perturbation Movement Categories Visualization for R2CTA

2 tiers namely tier 1 and tier 2 are shown in fig. 3. Considering a TSV is selected from tier 1 with its lower left corner located at the blue tile: Category 1 is the immediate neighbours of the randomly selected TSV with Manhattan distance of 1 or 2 units; Category 2 is the close proximity movement area of the randomly selected TSV with Manhattan distance of 3 units; Category 3 and Category 4 are the region above and below the square diagonal of the randomly selected TSV with Manhattan distance of 4 to 10 units. Category 6 is the action that the TSV stays the same.While not shown in the figure, Category 5 is the movement across to another empty space within a die.

4) Reward Function: In the Multi-Agent Reinforcement Learning framework, the standard centralized training and decentralized execution is applied. The reward function is the overall raw wirelength difference between current step and the next step: positive reward is given when the wirelength is reduced, otherwise negative reward is given. This global reward function value is fed into the critic networks to evaluate the performance of the chiplet system. After that, the actor network will update its policy based on the critic feedback.

Previous works [19], [24] implemented additional reward penalties when the next state performs worse than the best sampled neighbor state. While this approach can be effective in single-agent reinforcement learning [24], we remove such penalties in our multi-agent framework since local optimizations may conflict with achieving global optimum.

IV. EXPERIMENTS

In this section, the detailed experiments and results for R2CTA are presented. It is mainly compared with ATT-TA [19] on how echo state network could improve the multi-agent reinforcement learning framework.

Well known benchmarks for VLSI floorplanning including MCNC [25], [26] and GSRC [27] are used to evaluate the performance of R2CTA. As the experiments are conducted on multi-tier chiplets based designs which has its own unique characteristics comparing with the traditional 3DIC counterpart, the datasets are preprocessed for the chiplets based design by considering each tier contains only one die. The end goal for the reinforcement learning algorithm is to optimize wirelength by fine tuning the TSV assignment. All experiments are conducted on a Mac Mini built with M4 Apple Silicon chip with 16GB unified memory in Python 3 using PyTorch.

A. Experimental Setup

A generated dataset with similar characteristics to GSRC's n100 is used to train R2CTA as well as ATT-TA. It needs to go through a preprocessing phase, which contains multiple steps of allocating modules to dies, floorplanning of each die, and an initial TSV assignment. Corblivar [28] is used to perform the floorplanning for the 3DIC design so that such design can be utilized for both R2CTA and ATT-TA. It enables the floorplanninng optimization for metrics like half-perimeter wirelength, area, wirelength, power, congestion, thermal performance, etc.

The following table III shows the training parameters for both R2CTA and ATT-TA. To demonstrate the effectiveness of ESN, R2CTA without ESN and ATT-TA with ESN are trained with the parameters that generate the best performance. The key differences between the configurations lie in their exploration strategies (Gumbel-Softmax vs Epsilon Greedy), action space dimensionality (2 vs 6 actions), and target network update mechanisms (hard vs soft updates). For ESN-enabled variants, the reservoir sizes and dynamics are carefully tuned to balance computational complexity with performance.

B. Experimental Results

After the training phase, the trained model is used to make predictions on the testing dataset. The testing dataset is generated using benchmark circuits including MCNC's ami33, ami49 and GSRC's n100, n200, n300. The following table IV shows the experimental results comparing R2CTA and ATT-TA across four different configurations: vanilla ATT-TA, ATT-TA with ESN, R2CTA without ESN, and vanilla R2CTA. The main metric considered is wirelength reduction achieved through TSV assignment optimization. Each result represents the average of multiple independent runs.

C. Discussion

The experimental results demonstrate several key findings:

- **Overall Performance**: The vanilla R2CTA consistently outperforms all other configurations across all benchmarks, achieving wirelength reductions ranging from

TABLE III
TRAINING PARAMETERS FOR ATT-TA AND R2CTA

Parameter	ATT-TA		R2CTA	
	vanilla	with ESN	without ESN	vanilla
Actor Network	2 Hidden Layers with 32 Neurons Per Layer			
Critic Network Base	4 Attention Heads, One Encoder Layer, One Decoder Layer and One Output Layer, with 32 Neurons Per Layer			
Critic Network ESN	N/A	Neurons: 128, Spectral Radius: 0.7, Leak Rate: 0.7	N/A	Neurons: 84, Spectral Radius: 0.7, Leak Rate: 0.7
Exploration	Gumbel-Softmax with Temperature Annealing		Epsilon Greedy	
RL Parameters	Episodes: 200, Steps Per Episode: 2000, Buffer Size: 100000, Batch Size: 128, Sampled Neighbours Per Category: 8 Discount Factor $\gamma = 0.95$, Actor Learning Rate: 0.001, Critic Learning Rate: 0.01, Optimizer: Adam			
Action Space	2		6	
RL Parameters	Overall Raw Wirelength Reduction Between $\mathcal{S}(t)$ And $\mathcal{S}(t+1)$			
Update Target Network	Hard, Update Frequency: 10, $\tau = 0.001$		Soft, Update Frequency: 2, $\tau = 0.001$	

TABLE IV
EXPERIMENTAL RESULTS COMPARISON BETWEEN R2CTA AND ATT-TA

Benchmark	Tiers	ATT-TA				R2CTA			
		vanilla		with ESN		without ESN		vanilla	
		WL Red. (μm)	Speedup	WL Red. (μm)	Speedup	WL Red. (μm)	Speedup	WL Red. (μm)	Speedup
ami33	3	1801	1.0x	3488	1.94x	3035	1.68x	6522	3.62x
ami49	3	1766	1.0x	3549	2.01x	3131	1.77x	6430	3.64x
n100	3	2429	1.0x	4508	1.86x	5323	2.19x	7938	3.27x
n200	3	2315	1.0x	4259	1.84x	4223	1.82x	6890	2.98x
n300	3	2354	1.0x	4274	1.82x	4367	1.86x	6892	2.93x

2.93x to 3.64x compared to the baseline vanilla ATT-TA. This significant improvement validates the effectiveness of our integrated approach.

- **ESN Impact**: The addition of ESN to both ATT-TA and R2CTA shows clear benefits. ATT-TA with ESN achieves approximately 1.8-2.0x improvement over vanilla ATT-TA, while R2CTA's performance is further enhanced by the ESN integration.
- **Scalability**: The performance advantage of R2CTA remains robust across different benchmark sizes, from smaller circuits (ami33, ami49) to larger ones (n100, n200, n300). This demonstrates good scalability of our approach.
- **Consistency**: The relatively small variation in speedup factors across different benchmarks (ranging from 2.93x to 3.64x for vanilla R2CTA) indicates the stability and reliability of our method.

These results suggest that the combination of reinforcement learning with reservoir computing provides a powerful framework for TSV assignment optimization. The ESN component appears to enhance the learning capability significantly, while the RL framework enables effective exploration of the solution space.

V. CONCLUSION

In this paper, we presented a novel approach to optimize TSV assignment in chiplet-based designs using reinforcement learning enhanced with reservoir computing. Our method addresses the increased complexity of chiplet-based designs that are distinctive from traditional 3DICs and dies designs. Experimental results demonstrate that our approach based on echo state network achieves superior results compared to existing reinforcement learning solutions with respect to wirelength optimization. The proposed method shows particular promise in handling the increased complexity of chiplet-based designs while maintaining computational efficiency.

REFERENCES

[1] X. Wang, Z. Zhou, and Y. Yi, "Transforming ai landscape with neuromorphic computing and chiplets," in *Energy-Efficient Devices and Circuits for Neuromorphic Computing*. New York, NY: Elsevier, 2025, ch. 15.

[2] H. Jaeger, "Echo state network," *scholarpedia*, vol. 2, no. 9, p. 2330, 2007.

[3] J. H. Lau, *Chiplet design and heterogeneous integration packaging*. Springer, 2023.

[4] J. Knechtel, "Interconnect planning for physical design of 3d integrated circuits," Ph.D. dissertation, Saechsische Landesbibliothek-Staats-und Universitaetsbibliothek Dresden, 2014.

[5] V. Mnih, "Playing atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013.

[6] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[7] D. Vashisht, H. Rampal, H. Liao, Y. Lu, D. Shanbhag, E. Fallon, and L. B. Kara, "Placement in integrated circuits using cyclic reinforcement learning and simulated annealing," *arXiv preprint arXiv:2011.07577*, 2020.

[8] V. B. Pawar, "Application of machine learning to physical design," *San Francisco State University*, 2022.

[9] R. Lowe, Y. I. Wu, A. Tamar, J. Harb, O. Pieter Abbeel, and I. Mordatch, "Multi-agent actor-critic for mixed cooperative-competitive environments," *Advances in neural information processing systems*, vol. 30, 2017.

[10] H. Mao, Z. Zhang, Z. Xiao, and Z. Gong, "Modelling the dynamic joint policy of teammates with attention multi-agent ddpg," *arXiv preprint arXiv:1811.07029*, 2018.

[11] T. Lillicrap, "Continuous control with deep reinforcement learning," *arXiv preprint arXiv:1509.02971*, 2015.

[12] B. Schrauwen, D. Verstraeten, and J. Van Campenhout, "An overview of reservoir computing: theory, applications and implementations," in *Proceedings of the 15th european symposium on artificial neural networks. p. 471-482 2007*, 2007, pp. 471–482.

[13] G. Tanaka, T. Yamane, J. B. Héroux, R. Nakane, N. Kanazawa, S. Takeda, H. Numata, D. Nakano, and A. Hirose, "Recent advances in physical reservoir computing: A review," *Neural Networks*, vol. 115, pp. 100–123, 2019.

[14] W. Maass, "Liquid state machines: motivation, theory, and applications," *Computability in context: computation and logic in the real world*, pp. 275–296, 2011.

[15] X. Wang, Z. Zhou, and Y. Yi, "Tikz code generator for reservoir computing illustrations," https://www.wangxm.com/research/rctikz, 2024, online tool, Accessed: 2025-01-01.

[16] Y. Zhao, C. Hao, and T. Yoshimura, "Thermal and wirelength optimization with tsv assignment for 3-d-ic," *IEEE Transactions On Electron Devices*, vol. 66, no. 1, pp. 625–632, 2018.

[17] J. Ao, S. Dong, S. Chen, and S. Goto, "Through-silicon-via assignment for 3d ics," in *2011 9th IEEE International Conference on ASIC*. IEEE, 2011, pp. 353–356.

[18] D. Saha and S. Sur-Kolay, "Guided ga-based multiobjective optimization of placement and assignment of tsvs in 3-d ics," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 27, no. 8, pp. 1742–1750, 2019.

[19] W. Guan, X. Tang, H. Lu, Y. Zhang, and Y. Zhang, "Att-ta: A cooperative multiagent deep reinforcement learning approach for tsv assignment in 3-d ics," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 2023.

[20] C. Dussert, G. Rasigni, M. Rasigni, J. Palmari, and A. Llebaria, "Minimal spanning tree: A new approach for studying order and disorder," *Physical Review B*, vol. 34, no. 5, p. 3528, 1986.

[21] R. L. Graham and P. Hell, "On the history of the minimum spanning tree problem," *Annals of the History of Computing*, vol. 7, no. 1, pp. 43–57, 1985.

[22] J.-H. Han, X. Guo, K. Skadron, and M. R. Stan, "From 2.5 d to 3d chiplet systems: Investigation of thermal implications with hotspot 7.0," in *2022 21st IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (iTherm)*. IEEE, 2022, pp. 1–6.

[23] A. Ziabari, J.-H. Park, E. K. Ardestani, J. Renau, S.-M. Kang, and A. Shakouri, "Power blurring: Fast static and transient thermal analysis method for packaged integrated circuits and power devices," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 22, no. 11, pp. 2366–2379, 2014.

[24] Z. He, Y. Ma, L. Zhang, P. Liao, N. Wong, B. Yu, and M. D. Wong, "Learn to floorplan through acquisition of effective local search heuristics," in *2020 IEEE 38th International Conference on Computer Design (ICCD)*. IEEE, 2020, pp. 324–331.

[25] University of Michigan, "Mcnc benchmark circuits," 2024, accessed: 2024-09-20. [Online]. Available: http://vlsicad.eecs.umich.edu/BK/MCNCbench/

[26] K. Koźmiński, "Benchmarks for layout synthesis—evolution and current status," in *Proceedings of the 28th ACM/IEEE Design Automation Conference*, 1991, pp. 265–270.

[27] University of Michigan, "Gsrc benchmark circuits," 2024, accessed: 2024-09-20. [Online]. Available: http://vlsicad.eecs.umich.edu/BK/GSRCbench/

[28] J. Knechtel, J. Lienig, and I. A. M. Elfadel, "Multi-objective 3d floorplanning with integrated voltage assignment," *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, vol. 23, no. 2, pp. 1–27, 2017.