A Scalable In-Context Design and Extraction Flow for Heterogeneous 2.5D Chiplet-Package Co-Optimization

MD Arafat Kabir[1], Dusan Petranovic[2], and Yarui Peng[1]
[1]CSCE Department, University of Arkansas, [2]Mentor Graphics
makabir@uark.edu, yrpeng@uark.edu

***Abstract*—2.5D packages are currently popular choices for designing complex system-in-packages using several chiplets from different technologies. However, no standard CAD flow exists that can design, analyze, and optimize a complete heterogeneous 2.5D system. In this paper, we present a scalable in-context chiplet-package co-design and optimization flow for heterogeneous 2.5D systems. Our experimental studies show the proposed flow can achieve 99% extraction accuracy per-net compared to a holistic extraction flow, while the existing state-of-the-art in-context flow achieves 93% per-net accuracy. Our flow can design and optimize a heterogeneous 2.5D system to achieve the same performance as the holistic flow for homogeneous systems. Moreover, our flow is highly scalable with the number of technologies and chiplets in the system.**

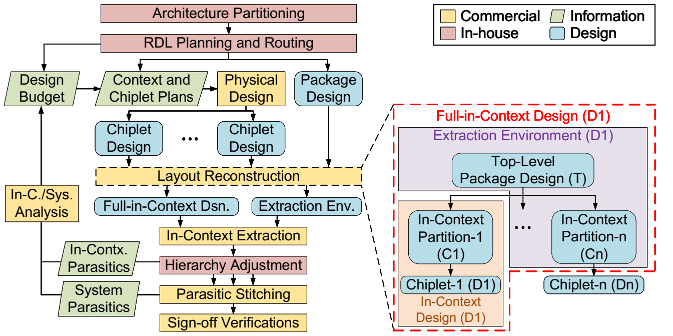
***Keywords*—2.5D Design, Chiplet-Package Co-Optimization, Heterogeneous, In-Context, Parasitic Extraction.**

I. Introduction

In advanced high-density 2.5D packages like TSMC's InFO design [1], chiplet-package interactions become significant. To ensure the highest reliability and performance in such systems, design, analysis, and optimization must be performed at the system level [2]. Recent research investigated the use of chiplet-package co-design for 2.5D systems for different optimization goals [2–4]. System architectures and designs containing tens of chiplets are proposed [5]. In many of these works, the package design is implemented using chip design tools because existing standard package design tools cannot handle the complexity of high-density 2.5D packages. As a result, many advanced package design features, such as variable width wires, any-angle routing, hexagonal pad structure, etc., cannot be utilized. With the development of packaging tools, similar flows can be implemented that include these advanced packaging features.

In most of these 2.5D chiplet-package co-design works [4, 5], chiplet-package interactions are not considered. In a more recent work [2], a holistic flow was proposed that captures the chiplet-package interactions and iteratively improves the system performance. However, this holistic flow cannot accommodate chiplets from heterogeneous technologies. Moreover, such a holistic approach does not scale well when tens of chiplets are involved. An in-context co-design flow that can handle chiplets from heterogeneous technologies was proposed in [6]. However, as evident from Table 5 in [6], the ground and coupling capacitances on package layers are highly overestimated. Such variance is not adequate for highly accurate analysis. Therefore, a chiplet-package co-design and co-optimization CAD flow is in imminent demand, especially for heterogeneous 2.5D systems. It should achieve similar accuracies as the holistic flow and scale well with the number of chiplets and technologies involved.

In this paper, we present a scalable in-context chiplet-package co-optimization flow for heterogeneous 2.5D systems. Our flow achieves highly accurate extraction results and optimizes a heterogeneous system to provide performance comparable to homogeneous systems. Our flow leverages industry-standard tools to perform in-context extraction on heterogeneous 2.5D systems while ensuring the compatibility of extraction results with the industry-standard ASIC CAD flow. We then utilize the holistic-like extraction result to perform cross-boundary analysis and iterative system-level optimization. In this work, we claim the following new contributions: (1) An accurate and scalable extraction strategy to perform in-context extraction and timing optimization of heterogeneous 2.5D systems; (2) A new in-context design flow to perform timing and signal integrity analyses with a complete view of the heterogeneous system; (3) A comparative case study to validate our methodology.



(a) In-Context Co-optimization Flow (b) Design Hierarchy and Assemblies

Fig. 1. Proposed in-context flow for heterogeneous 2.5D systems.

To our best knowledge, there exists no other tool flow that implements a scalable in-context co-optimization flow for heterogeneous 2.5D systems with holistic-like accuracy and effectiveness in extraction, analysis, and optimization steps.

II. Chiplet-Package Co-Design Flow

A. Top-Level Package Planning

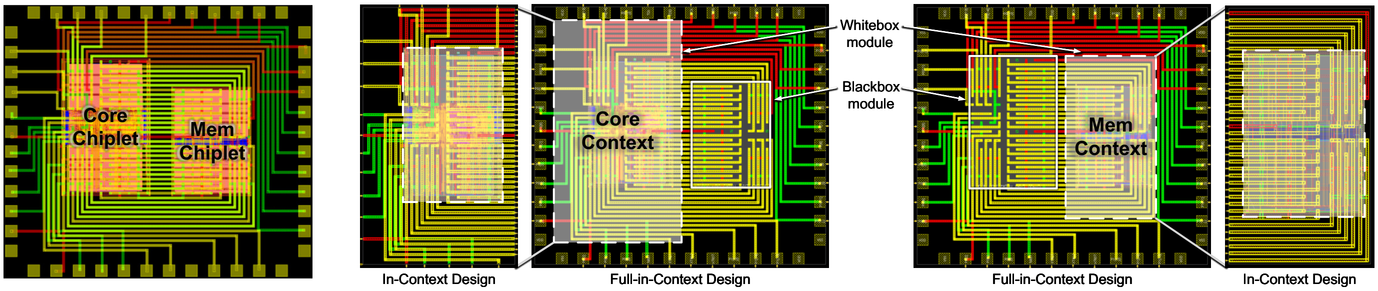
Our proposed flow for heterogeneous 2.5D systems is illustrated in Fig. 1. The top-level planning consists of architecture partitioning and redistribution layer (RDL) planning steps in the figure. Based on system requirements, the gate-level netlist is partitioned into chiplets. These chiplets are converted into sub-designs, treating the top-level design as the package design. As there is no physical design to extract the parasitics in this step, the package wireload is estimated using a model by our in-house RDL planner tool. Based on this plan and estimated wireload, timing budgets are extracted for each chiplet. In-context partitions are defined for each chiplet, which contains the chiplet and part of the package surrounding it. Fig. 2 (b), (c) illustrate such in-context partitions. Hierarchical sub-designs are generated for in-context partitions and chiplets with top-level constraints.

B. Physical Design

The top-level planning step determines the package floorplan, inter-chiplet routing, and signal assignments of chiplets. This plan is

This material is based upon work supported by the National Science Foundation under Grant No. 1755981. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

978-1-6654-4269-5/21/$31.00 © 2021 IEEE



(a) Assembled Full-System GDS [3]

(b) In-Context Designs for Core-Chiplet in 7M3R

(c) In-Context Designs for Mem-Chiplet in 6M3R

Fig. 2. Package and assembled system layouts of the experimental homogeneous and heterogeneous 2.5D systems.

followed in the physical implementation of the package. The physical designs of chiplets are prepared by treating them as individual chips with top-level constraints. The top-level constraints ensure that individual chiplet designs conform to the holistic plan and design budgets. Each chiplet has its own separate plan and can be implemented in any technology, independent of other chiplets.

C. In-Context Parasitic Extraction

The proposed extraction flow takes advantage of the industry-standard in-context extraction tool designed for flip-chip package extraction. Flip-chip extraction tools take the chip design as the extraction target and the package routing as the extraction environment. The coupling capacitance between chiplet and package wires is converted to ground capacitance. Although this extraction result is good enough for chip-level timing analysis and optimizations, the chiplet-package interactions are lost. As a result, system-level analysis and optimizations like static timing analysis (STA), signal integrity (SI), and power integrity (PI) analysis are not possible.

Fig. 1 (b) shows the design hierarchy in our flow. In the layout reconstruction step, different levels of the hierarchy are assembled to create layouts for extraction. We refer to the assembly of a chiplet, D1, with its in-context partition of the package as the "in-context design" of D1. The assembly of all in-context partitions, except that of D1, with the top-level package is used as the "extraction environment" of D1. Note the extraction environment does not include chiplet details and treats them as black boxes. A combination of the in-context design and its extraction environment is the "full-in-context design" for the chiplet. Fig. 2 (b), (c) show these layouts for our experimental design. For top-level package T, any chiplet D_i, its in-context partition C_i, and a given chiplet D_x, general mathematical definitions of these designs are given below, where summation represents design-assembly.

$$\begin{aligned} \text{In-Context Design:} \quad & C_x + D_x \\ \text{Extraction Environment:} \quad & \textstyle\sum_{i=1, i\neq x}^{n} C_i + T \\ \text{Full-in-Context Design:} \quad & \textstyle\sum_{i=1}^{n} C_i + T + D_x \end{aligned}$$

In our proposed flow, we use the full-in-context design of a chiplet and its extraction environment with the flip-chip extraction tool. The tool performs extraction on the entire in-context design instead of the chiplet only. As a result, the chiplet-package interactions within the in-context design are preserved in the parasitic netlist.

Since flip-chip extraction tools are not designed for hierarchical extraction, the extracted parasitic netlists cannot be directly used for hierarchical annotation. We use an in-house tool to fix this hierarchy problem, which adjusts the terminal nodes of the inter-chiplet package nets based on the design hierarchy. It also performs some clean-up to remove additional information related to other chiplets that are not part of the extracted in-context design.

D. Optimization through Iterations

The extraction flow creates separate parasitic netlists for each in-context design. As each netlist contains all chiplet-package interactions for a given chiplet, it can be used to perform cross-boundary analysis and optimization at the in-context design level. Moreover, the netlists can be stitched together to create a holistic view of the system to perform system-level analysis. In our study, we use the system-level parasitic netlist to perform full-system STA and create timing contexts for each chiplet. These timing contexts are used to perform iterative optimization of the chiplets to improve overall system performance. Similar iterative optimizations can be performed to improve the package design, SI, and PI of the entire system.

III. Experimental Study

A. Experimental Setup and Designs

For the experimental study, we use an ARM Cortex-M0-based microcontroller system. We partition the system into a core-chiplet and a mem-chiplet, as presented in [6]. The core-chiplet contains all the logic cells and 8KB memory, and the mem-chiplet contains the rest 8KB memory. We modify the Nangate 45nm PDK to create two technology stacks, named 7M3R and 6M3R. The 7M3R stack has the same settings as presented in Table 1 of [6]. The lower seven layers are for chiplet internal routing and the top three layers are for package routing. The top three layers are adjusted to mimic high-density 2.5D package RDLs. The 6M3R stack has six lower layers with the same dimensions as the corresponding layers of 7M3R for chiplet internal routing. The three RDLs are exactly the same as in 7M3R. Although both of these stacks are for 45nm technology, they are heterogeneous from the tool flow perspective.

For a comparative study, we implement a homogeneous system using the holistic flow [2] and our proposed in-context flow. This system is designed using the 7M3R technology and standard cells from the Nangate45nm cell library. Both chiplets are assembled with the package for holistic extraction. In the in-context flow, we create in-context partitions of the package and follow the extraction methodology discussed in Section II. Fig. 2 (a) shows the assembled GDS of the homogeneous system. To study the compatibility and effectiveness of our flow with heterogeneous systems, we implement the microcontroller system using two different PDKs. The core-chiplet is implemented in 7M3R using cells from Nangate45nm cell library, and the mem-chiplet is implemented in 6M3R using cells from the gscl45 cell library, making the design heterogeneous.

B. Analysis and Results

Table I presents the comparison between the extraction result obtained using the holistic methodology, our proposed in-context methodology, and the in-context methodology presented in [6]. We refer to our proposed flow as the "new flow" and the flow in [6] as

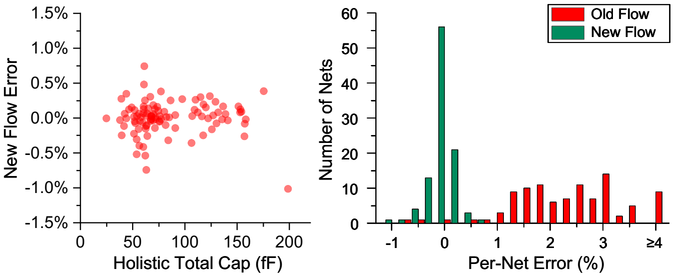


Fig. 3. Comparison of the total capacitance on individual nets of the proposed flow with state-of-the-art in-context flow [6].

TABLE I
COMPARISON OF HOLISTIC (HOLI) VS. IN-CONTEXT (IN-C) COUPLING (CCAP) AND TOTAL (TOTAL CAP) CAPACITANCE EXTRACTION (IN FF)

	Metal	M1-M5	M6	M7	R1	R2	R3
CCAP	Holi	9275	1172	196	1529	2441	1685
	In-C Old	9346	1181	188	1564	2478	1690
	In-C New	8992	1203	193	1517	2390	1640
Total CAP	Holi	31056	3307	498	2547	2669	2209
	In-C Old	31140	3324	489	2661	2749	2251
	Old Err%	**0.27%**	**0.51%**	**-1.79%**	**4.49%**	**3.01%**	**1.91%**
	InC New	31238	3350	495	2591	2654	2192
	New Err%	**0.59%**	**1.31%**	**-0.59%**	**1.74%**	**-0.55%**	**-0.76%**

TABLE II
COMPARISON OF HOLISTIC AND IN-CONTEXT FLOW OPTIMIZATION RESULTS OF THE EXPERIMENTAL DESIGNS

	Design	Homogenous		Heterogeneous
Performance	Iteration	Holistic	In-Context New Flow	
	Initial	288 MHz	287 MHz	278 MHz
	1st iteration	293 MHz	290 MHz	294 MHz
	2nd/final iteration	300 MHz	300 MHz	300 MHz
Power	Power Group	Holistic	In-Context New Flow	
	Wire	4.35 mW	4.37 mW	4.21 mW
	Cell	6.39 mW	6.36 mW	6.20 mW
	Total	10.74 mW	10.73 mW	10.41 mW

the "old flow." As observed from the table, the coupling capacitance between chiplets and the package is preserved with almost holistic-like accuracy, and is comparable to the numbers of the old flow. The coupling numbers for chiplet routing layers (M1-M7) are within +/-3%. This accuracy level is high enough to perform chiplet-level SI analysis, including the impact of RDLs.

Unlike the old flow, the total capacitances on all layers in the new flow are very close to that of the holistic extraction. In the old flow, the total capacitance is highly overestimated on the package layers, due to additional fringe capacitances extracted at the boundary of the in-context partitions. These fringe capacitances are non-existent in the actual design. In our proposed flow, because the extraction tool is aware of the extraction environment while performing in-context extraction, those fringe capacitances are not extracted at the boundary.

In Fig. 3, the scatter plot on the left shows the total capacitance error of each net in the new flow with respect to the holistic extraction. As observed in the scatter plot, the extracted parasitics on each net is as accurate as the holistic flow, with 1% error margin. The histogram on the right organizes the errors into 0.5% bins. The per-net extraction error in the new flow is 0% for most of the nets. However, the parasitics is overestimated in almost all nets in the old flow, with error varying between -1% and 7%. Thus, our proposed flow improves the per-net extraction accuracy from 93% to 99%. As the signal delay depends on the total load capacitance, the parasitic netlist obtained in our flow can be used to perform a highly accurate timing analysis of the system.

The iterative timing optimization results are shown in Table II. As observed from the "Homogeneous" column, the timing optimization results from the in-context flow very closely match holistic flow. As the heterogeneous design incorporates different PDKs and cell libraries, the first implementation with RDL wireload would not match with the holistic designs. However, the optimization results of the following iterations closely match. Similar results are observed in the power comparison table. Both homogeneous designs have almost the same power numbers in the final iteration. As the heterogeneous design uses a different cell library in the mem-chiplet, the power numbers slightly differ from that of the homogeneous design. These results validate that our in-context flow for heterogeneous systems achieves the accuracy and optimization results comparable to the state-of-the-art holistic flow for homogeneous designs.

However, unlike the existing in-context flows for heterogeneous systems, our flow is highly scalable in terms of the number of technologies and chiplets. As the in-context parasitic netlist contains chiplet-package interactions within the in-context partition, cross-boundary analysis and optimizations can be performed on each chiplet independently. In the end, a system-level holistic view can be created through hierarchical annotation of the in-context parasitics to perform full-system analysis and verification. For a 2.5D system with multiple chiplet technologies, the initial plans can be distributed to several design houses with the package in-context partitions. They can perform cross-boundary analysis and optimizations in their own contexts without worrying about others parts of the package. This way, multiple design houses can collaborate on a large-scale heterogeneous 2.5D system, containing hundreds of chiplets in tens of heterogeneous technologies, yet maintain cross-boundary analysis, system-level optimization, and verification.

IV. CONCLUSIONS

In this paper, we present a scalable in-context design, extraction, analysis, and optimization flow for heterogeneous 2.5D systems. Through a comparative study between two implementations of a homogeneous system, we show that our in-context methodology can achieve 99% extraction accuracy w.r.t holistic method. With a 45nm heterogeneous system using two different PDKs, we demonstrate that our flow can perform holistic-flow-like optimizations on heterogeneous systems to improve system-level performance. Unlike existing flows, our flow is highly scalable in terms of the number of chiplets and heterogeneous technologies in the system. It enables parallelism and speed up through independent in-context partitions yet offers accurate system-level analysis, optimization, and verification.

REFERENCES

[1] D. Yu, "A new integration technology platform: Integrated fan-out wafer-level-packaging for mobile applications," in *Symposium on VLSI Technology*, June 2015, pp. T46–T47.

[2] M. A. Kabir and Y. Peng, "Holistic Chiplet-Package Co-Optimization for Agile Custom 2.5D Design," *IEEE Transactions on Components, Packaging, and Manufacturing Technology*, vol. 11, no. 5, pp. 715–726, 2021.

[3] M. A. Kabir and Y. Peng, "Chiplet-Package Co-Design For 2.5D Systems Using Standard ASIC CAD Tools," in *Asia and South Pacific Design Automation Conference*, Jan. 2020, pp. 351–356.

[4] H.-T. Wen, Y.-J. Cai, Y. Hsu, and Y.-W. Chang, "Via-based Redistribution Layer Routing for InFO Packages with Irregular Pad Structures," in *Design Automation Conference*, Jul. 2020, pp. 1–6.

[5] J. Kim, G. Murali, H. Park *et al.*, "Architecture, Chip, and Package Codesign Flow for Interposer-Based 2.5-D Chiplet Integration Enabling Heterogeneous IP Reuse," *IEEE Transactions on Very Large Scale Integration Systems*, vol. 28, no. 11, pp. 2424–2437, 2020.

[6] M. A. Kabir, D. Petranovic, and Y. Peng, "Coupling Extraction and Optimization for Heterogeneous 2.5D Chiplet-Package Co-Design," in *International Conference on Computer-Aided Design*, Nov. 2020, pp. 1–8.