Multi-Product Optimization for 3D Heterogeneous Integration with D2W Bonding

Zhen Zhuang[1], Kai-Yuan Chao[2], Bei Yu[1], Tsung-Yi Ho[1], Martin D.F. Wong[1]
[1]The Chinese University of Hong Kong
[2]Hong Kong Research Center, Huawei Technology Investment Co., Ltd.

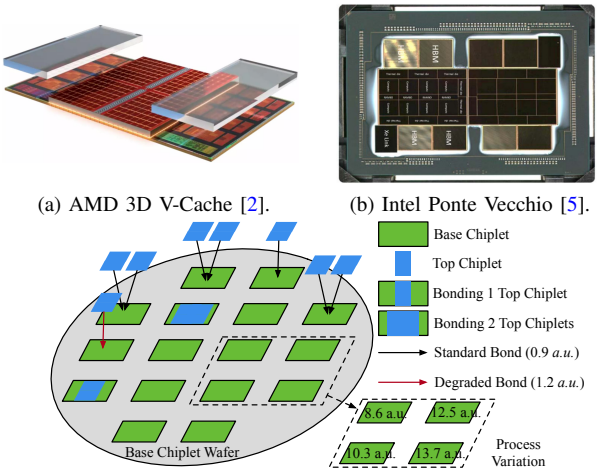
***Abstract*—3D heterogeneous integration enables the integration of multiple heterogeneous chiplets into the same package with the effective reduction of package size and interconnection latency. According to the market requirement, chiplets with robust re-usability and effective cost reduction can be selected from a library to form different package products for enlarging total profit. Since die-to-wafer (D2W) bonding enables the chiplets with different sizes to be bonded in a package, it is a more flexible option for 3D heterogeneous integration compared with the conventional wafer-to-wafer (W2W) bonding. However, this promising technique creates new issues, including 1) flexible chiplet bonding enabling more than one chiplet to be bonded with a base chiplet to construct multiple products and 2) degraded bonding leading to the degradation of performance. In this work, a distributed integer-linear-programming-based (ILP-based) method is proposed to efficiently maximize the profits of multiple package products considering the issues of cost-addition 3D heterogeneous integration with D2W bonding. Compared with the baseline, the distributed ILP-based method can achieve the best profits while achieving a 5.96X speedup. To the best of our knowledge, this is the first work to solve the multi-product optimization problem for 3D heterogeneous integration with D2W bonding.**

I. Introduction

While transistor scaling is still important, the rising cost and complexity have driven the industry to turn to advanced packaging technologies. Nowadays, chiplet-based systems supporting heterogeneous integration have been a tendency for the cost-effective development of high-performance designs. The heterogeneous integration roadmap identifies the challenges and necessity to develop advanced packaging technologies [1]. Since hot applications, such as big data, need large cache capacity, along with sizes in different memory hierarchies, to improve the performance, 3D heterogeneous integration has wide markets for the *memory-bound design (MBD)* [2]. Fig. 1(a) shows the architecture of AMD 3D V-Cache [2]–[4] which bonds a 64MB L3 Cache chiplet and a base chiplet with eight cores. Intel [5] has announced their *high-performance computing (HPC)* chip with eight cache chiplets on top of a base chiplet as shown in Fig. 1(b). With the development of packaging technologies, including fine-pitch hybrid bonding, micro-bump bond, and nanowire bond, new issues are emerging in the advanced chiplet bonding of 3D heterogeneous integration.

Die-to-wafer (D2W) bonding is a promising bonding technique for advanced packages benefiting from high yield and supporting high-density heterogeneous designs [6]. Compared with wafer-to-wafer (W2W) bonding, which is widely implemented by industries, D2W bonding offers more flexibility about chiplet sizes, enabling 1) the mismatch of chiplet footprint and 2) bonding multiple small chiplets to a base chiplet [7]. The *known good die (KGD)* process of D2W bonding and, usually, smaller top dies allows yield improvement in comparison to W2W bonding [6]. Recently, multiple institutions have made major promotions to develop high-quality D2W bonding techniques [6]–[8]. Fig. 1(c) shows the process of D2W bonding. The top chiplets of 3D packages are selected and placed on the base chiplets belonging to the same wafer. The issues of 3D heterogeneous integration with D2W bonding are solved in this work to improve the economic benefits of package products.

Process variation, parametric yield, and product profits are the conventional issues related to the bonding process and solutions [9]–[11]. Process variation is the change of the electrical parameters different from the original intent of designers. As shown in Fig. 1(c), the latency of the base chiplets in the same wafer is different due to the process variation.



(a) AMD 3D V-Cache [2]. (b) Intel Ponte Vecchio [5].

(c) D2W bonding.

Fig. 1 The illustration of 3D heterogeneous integration with D2W bonding. (a) 3D heterogeneous integration: AMD 3D V-Cache. (b) 3D heterogeneous integration: Intel Ponte Vecchio. (c) After the KGD test, wafer probing, and manufacture bonding statistics, those electrical latency variation parameters are available for D2W bonding.

Parametric yield is the number of functional chiplets that meet the required constraints, such as performance. However, the parametric yield is not the ultimate objective of package products since the companies pursue maximized profits over best-sellable volume mixes. Considering the profits of packages are determined in the bonding stage, the objective of this work is to maximize the total profits of multiple high-end-market-segment products, which is one of the most important objectives of today's relatively costly 3D-IC packages.

In addition to the conventional issues, 3D heterogeneous integration with D2W bonding creates new issues, including 1) **the flexible chiplet bonding** of 3D heterogeneous integration enabling more than one top chiplet to be bonded with a base chiplet and 2) **the degraded bonding** leads to the degeneration of performance. To adequately take advantage of the flexibility of 3D heterogeneous integration with D2W bonding, the problem of bonding more than one chiplet to the base chiplet is solved in this work. As shown in Fig. 1(c), one or two top chiplet(s) can be bonded to a base chiplet. For D2W bonding, the deformity of the bond leads to changes in link resistance, which increases the bonding latency [12], [13]. Before D2W bonding, prior individual die electrical measurement, e.g., die or wafer probing, and bonding electrical manufacture statistics, bonding base chiplets close to the edge of the wafer with higher mechanical stress/warpage, could be derived. In this work, high-risk positions of bond deformity, vertical alignment offset, bond strength/RC variation, etc., are identified to better estimate the performance of packages. Our work can be applied to hybrid bonding, micro-bump bonding, or future 3D bonding that brings denser, lower electric parasitics, or cheaper process. The method proposed in this work can provide efficient planning for advanced D2W bonding techniques, including collective D2W bonding and direct placement D2W bonding. Without loss generality, we will use the x86 AMD cache bond example to illustrate. As shown in Fig. 1(c), based on prior statistics, the standard latency of bonds is 0.9 a.u. (arbitrary unit), however, the latency of degraded bonding is 1.2 a.u.

A. Previous Work

In this section, the problem solved in this work is compared with that of previous work. Furthermore, the drawbacks of applying previous methods for 3D heterogeneous integration with D2W bonding and the necessity of designing a more effective method are illustrated. Fig. 2 shows the comparison of the problems between this work and previous work. Firstly, the problem solved in this work will be introduced. Without loss of generality, five top chiplets tc_i can be bonded with three base chiplets bc_j at different locations of one wafer. Each base chiplet already has a built-in 64 MB L3-Cache and a set of cores. Each top chiplet mainly has a 32 MB L3-Cache. Base chiplets and top chiplets can have different technology nodes. Post-silicon process will program the final capacity. If one top chiplet is bonded with a base chiplet, one of the two 32 MB L3-Cache of the base chiplet is connected vertically to the top chiplet, which means the

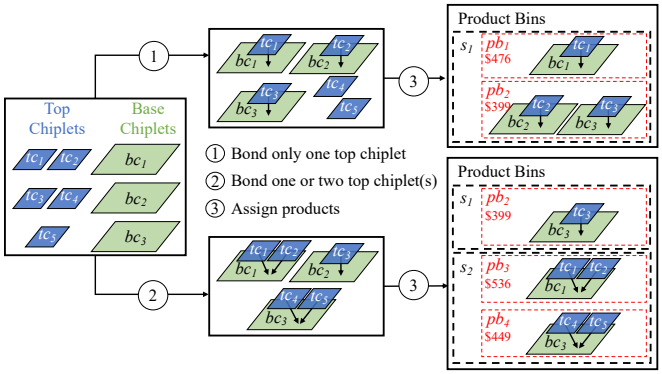
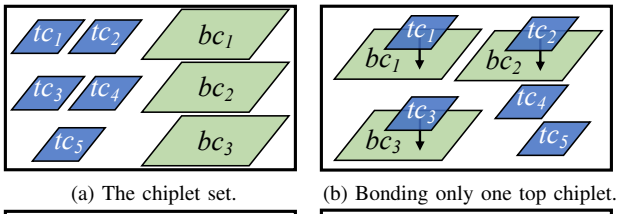


Fig. 2 The illustration of the problem solved in this work and previous work.

package product has 64 MB L3-Cache. If two top chiplets are bonded with a base chiplet, the 64 MB L3-Cache (both 32 MB L3-Cahce components) of the base chiplet is used, which means the package product has 128 MB L3-Cache. The bonded packages should be assigned to different *stock keeping units (SKUs)* s_l, which categorize the packages based on the architectural parameters, such as L3-Cache capacity. Each SKU has different product bins pb_k which limit the value interval of the architectural performance parameters, such as cycles per instruction or frequency, or watt, of the products with marketing required quantity. Cycles per instruction is in use for this paper to illustrate our method. The package products identify the top chiplet(s), base chiplet, and corresponding product bin. Each base chiplet can be bonded with one or two top chiplet(s). The objective is to maximize the total profits of multiple package products. The process of this problem corresponds to ② and ③ in Fig. 2. One or two chiplet(s) can be bonded with each base chiplet. Then, the bonded package products are assigned to proper product bins according to L3-Cache capacity and performance parameters.

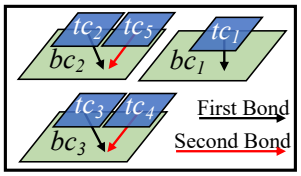
Siddharth *et al.* [9], [10] propose two methods that can be applied to bond only one top die to a base die with the same size for a 2-layer 3D IC design corresponding to ① in Fig. 2. Considering 3D heterogeneous integration, previous work can be used to bond only one top chiplet with one base chiplet as shown in Fig. 3(b). Therefore, package products can only be assigned to the product bins belonging to s_1 which has 64 MB L3-Cache and lower profit compared with s_2. The architectural simulation method used in this work is applied for the examples in Fig. 3. According to simulation results and the parameter intervals of product bins, package products are assigned to the corresponding product bins. The total profits are \$1274 for the case shown in Fig. 3(b).

To tackle the problem solved in this work, the previous work should be extended to incrementally bond the second top chiplet for achieving larger profits since the package products with two top chiplets have 128 MB L3-Cache, and can be assigned to s_2 which has larger profits. In the incremental stage, tc_1, tc_2, and tc_3 have been bonded with

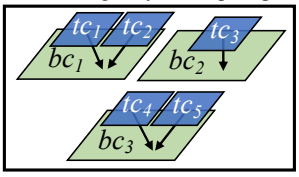


(a) The chiplet set.

(b) Bonding only one top chiplet.



(c) Incrementally bonding the second chiplet.



(d) Multi-product optimization.

Fig. 3 The comparison between previous work and multi-product optimization.

the three base chiplets. Only tc_4 and tc_5 can be selected to be bonded with the initial integrations. The solution is shown in Fig. 3(c). The total profits are \$1374 for the case shown in Fig. 3(c).

The multi-product optimization problems have been studied in many fields, such as physical design [14]. For the multi-product optimization problem in this work, the extension of previous work [9], [10] cannot achieve the global optimal solution. Simultaneously optimizing the bonding of two top chiplets can lead to the global optimal solution as shown in Fig. 3(d). The total profits are \$1384 for the case shown in Fig. 3(d). Since each wafer has many of base chiplets with high cost for the high-end market after extra 3D bonding, the gap between the solutions of previous work and the global optimal solution is much more significant. With the development of technology nodes, the optimality gap will be larger and larger. Therefore, an effective method is necessary for the multi-product optimization problem.

B. Challenges

In this section, the challenges of multi-product optimization are analyzed. The problem solved by previous work [9], [10] can be regarded as a bipartite matching problem since only one top chiplet can be bonded with a base chiplet for a 2-layer 3D IC design. However, bonding more than one top chiplet to a base chiplet is completely different and more complicated.

Firstly, the performance of a package should be simulated based on the parameters of all chiplets. Therefore, initial integrations, such as the integrations in Fig. 3(b), cannot generate effective architectural simulation results for the potential products with more than one top chiplet by previous works. The initial integrations cannot provide valid guides for the subsequent incremental bonding stage if the previous work is applied to the problem of this work.

Then, architectural simulation results are not linearly changed with the variation of the number of top chiplets. All the combinations of top chiplets should be considered to solve the problem in this work. Therefore, bonding one or two chiplet(s) with a base chiplet is difficult to be solved by previous graph-based methods such as the bipartite matching method of [10], since different combinations with the same top chiplet lead to conflicts in a graph.

Last but not least, considering all the combinations of top chiplets dramatically increases the complexity of the multi-product optimization problem. Hence, an efficient method is necessary to solve the problem.

C. Our Contributions

In this paper, the multi-product optimization problem for 3D heterogeneous integration with D2W bonding is formulated, and an efficient method is designed to maximize the profits of package products. The major contributions:

- To the best of our knowledge, this is the first work to solve the issues of 3D heterogeneous integration with D2W bonding, including the flexible chiplet bonding and the degraded bonding, for the maximization of package profits.
- The flexible chiplet bonding of 3D heterogeneous integration with D2W bonding enables more than one top chiplet to be bonded with a base chiplet to construct products with higher architectural parameters, such as L3-Cache capacity, and larger profits.
- The degraded bonding effect is formulated to better estimate the performance of packages, which enables the packages to be assigned to the correct product bins.
- A distributed ILP-based (**DILP**) method is proposed regarding the nature of the problem. Compared with the baseline, which extends the previous work for the problem solved in this work, DILP can achieve a 5.96X speedup and the largest total profits. Furthermore, DILP can provide effective hints for users.

This paper is organized as follows. Section II presents the architectural simulation method for computing performance parameters, the issues of 3D heterogeneous integration and D2W bonding, and the problem formulation. Section III introduces the technical details of the proposed methods. Section IV and Section V present the experimental analysis and conclusion, respectively.

II. Preliminaries

A. Terminologies and Notations

The following terminologies and notations are used:

- $TC = \{tc_i \mid 1 \leqslant i \leqslant |TC|\}$ is the set of top chiplets. Each top chiplet has the following parameters: latency and the capacity of L3-Cache. Due to process variation, the chiplets have different latencies.
- $BC = \{bc_i \mid 1 \leqslant i \leqslant |BC|\}$ is the set of base chiplets from the same wafer. Each base chiplet has a latency parameter, a set of cores, 64 MB L3-Cache, and is identified as whether it has degraded bonding which affects the bonding latency and chiplet latency. Post-silicon process will program the final capacity. If one top

chiplet is bonded with a base chiplet, one of the two 32 MB L3-Cache of the base chiplet is connected vertically to the top chiplet. If two top chiplets are bonded with a base chiplet, the 64 MB L3-Cache (both 32 MB L3-Cahce components) of the base chiplet is used. Due to process variation, the chiplets have different latencies.

- $S = \{s_i \mid 1 \leqslant i \leqslant |S|\}$ is the set of *stock keeping units (SKUs)*. SKU is a distinct type of item for sale in inventory. All attributes associated with the item type are used to distinguish it from other item types. In this work, each SKU has an L3-Cache capacity attribute to classify package products into different broad categories. Furthermore, each SKU has three product bins.
- $PB = \{pb_i \mid 1 \leqslant i \leqslant |PB|\}$ is the set of product bins. A Product bin belonging to an SKU is used to categorize package products according to performance. In this work, each product bin has the following attributes: corresponding SKU, the maximum count of package products, the lower bound of CPI, the upper bound of CPI, and the profit of each package product.
- $PP = \{pp_i \mid 1 \leqslant i \leqslant |PP|\}$ is the set of package products. Each package product $pp_i = \{(ptc_{i_1}, ptc_{i_2}, pbc_j, ppb_k) \mid ptc_{i_1} \in TC, ptc_{i_2} \in TC \text{ or } ptc_{i_2} = void, pbc_j \in BC, ppb_k \in PB\}$ has the following parameters: latency, CPI, and profit. $void$ means no second top chiplet ptc_{i_2} should be bonded to the base chiplet pbc_j.

B. Architectural Performance Simulation Method

All the parameters of each chiplet are tested before D2W bonding. In this work, *cycles per instruction (CPI)* is simulated using the existing methods from [9], [15], [16]. Based on the following architectural simulation method, the CPI of each package product can be calculated from the known latency of each component. Note that any simulation method and any parameter can be applied in this work, since the parameters of each potential bonding product are calculated before the bonding process. The latency of each package product pp_i is calculated as below [9]:

$$lat_{pp_i} = max(lat_{ptc_{i_1}}, lat_{ptc_{i_2}}) + lat_{pbc_j} + lat_{bond}, \quad (1)$$

where lat_{bond}, $lat_{ptc_{i_1}}$, $lat_{ptc_{i_2}}$, and lat_{pbc_j} represent the latency of bond, ptc_{i_1}, ptc_{i_2}, and pbc_j, respectively. lat_{void} is zero. The CPI of each package product pp_i is calculated as below [15]:

$$CPI_{pp_i} = staCPI + \alpha_1 \times (lat_{pp_i} - stalat), \quad (2)$$

where $staCPI$, $stalat$, and α_1 represent the standard CPI, standard latency, and coefficient generated from the statistic, respectively. The $staCPI$ can be calculated as below [15]:

$$staCPI = \alpha_2 \times MP + \beta_1, \quad (3)$$

where MP, α_2, and β_1 are the penalty of cache miss and coefficients generated from statistics, respectively. Since MP is different for the packages with different L3-Cache capacity, we have the following equation:

$$MP_{ptc_{i_2} \neq void} = CF \times BF \times MP_{ptc_{i_2} = void}, \quad (4)$$

where $MP_{ptc_{i_2} \neq void}$, $MP_{ptc_{i_2} = void}$, CF, and BF represent the penalty of cache miss for the packages with two top chiplets, the penalty of cache miss for the packages with one top chiplet, the factor for large cache capacity, and the factor for degraded bonding effect, respectively.

C. The New Issues of Problem

To adequately take advantage of the flexibility of 3D heterogeneous integration, bonding one or two top chiplet(s) to a base chiplet is considered in this work as shown in Fig. 1(c). Since the selection of the number of top chiplets for each base chiplet and assigning packages to appropriate product bins according to the attributes cannot be easily solved within the same graph, the graph-based methods, like [10], are not applied in this work. The ILP-based methods are proposed for the complicated problem.

To better estimate the parameters of D2W bonding, the degraded bonding caused by the bonding deformity, which is a significant problem for promising hybrid bonding, is formulated. Before D2W bonding, prior statistics could be derived such as bonding base chiplets close to the edge of the wafer. The base chiplets on the positions with the high risks of bonding deformity will have large bonding latency lat_{bond} and large bonding factor BF.

D. Problem Formulation

The multi-product optimization for 3D heterogeneous integration with D2W bonding is formulated as below:

- Given a set of top chiplets, a set of bottom chiplets from the same wafer, a set of SKUs, and a set of product bins, one or two top chiplet(s) should be bonded with one base chiplet to construct a package product, and the package products should be assigned to appropriate product bins according to the attributes (the L3-Cache capacity, the interval of CPI, and the maximum count) such that the total profits of package products are maximized.

III. Technical Details

The method proposed in this work, distributed ILP-based method (**DILP**), and the baseline extending previous work to fit the problem of work are introduced in this section.

Firstly, a one-pass ILP (**OPILP**) model is constructed to solve the complete problem. Since the model of OPILP is complicated, OPILP is not efficient enough to generate effective solutions within acceptable runtime and memory. Therefore, DILP, fusing OPILP, is proposed to solve the problem in several batches based on a **fixed-interval extraction method** and a **propagation method** considering the nature of the problem. Then, a baseline extending previous work [9], [10] is proposed for the comparison with DILP to show the efficiency of DILP. According to the analysis of Section I, previous work [9], [10] cannot directly solve the problem of this work. To modify the previous work to tackle the problem of this work, the baseline with two stages, the first stage of bonding one top chiplet and the second stage of incrementally

TABLE I The notations used in the ILP models

Notation	Type	Meaning
Constant		
$p_{i,j,k}$	0-1	identify if i-th component bonded with j-th component can be assigned to pb_k
r_k	real	the profit of pb_k
pn_k	integer	the maximum count of pb_k
Variable		
$t_{i,j}$	0-1	identify if i-th component is bonded to j-th component
$pa_{i,j,k}$	0-1	identify if i-th component bonded with j-th component is assigned to pb_k

bonding the second chiplet, are proposed. The previous work [9], [10] can only solve the problem of the first stage. In the first stage, each base chiplet is bonded with one top chiplet based on an ILP model like previous work [9], [10]. In the second stage, the bonded base chiplets are considered to be bonded with the second top chiplets, and the packages are assigned to proper product bins to maximize the total profits based on an ILP model. Experimental results show that DILP has the best efficiency compared with the baseline.

The technical details of the proposed methods will be shown below. The notations used in the ILP models are shown in TABLE I. "Components" represent different items in different models. For example, "i-th component" and "j-th component" represent the top chiplet and base chiplet in the first stage of the baseline, respectively.

A. One-Pass ILP Method

The multi-product problem is formulated as an ILP problem and solved by an ILP solver. To achieve the integration of one or two top chiplet(s), a set of *top chiplet integration* is defined as $TCI = \{tci_i \mid 1 \leqslant i \leqslant |TCI|\}$. As shown in Fig. 4, TC is transformed into TCI at first. Each tci_i is constructed by two top chiplets. It is regarded as the integration of one top chiplet when the combined two top chiplets are the same top chiplets. For example, tci_1 only includes one top chiplet tc_1. TCI and the combinations of TC have a bijection relationship. For each top chiplet pair tc_i and tc_j, the combination of the pair corresponds to $tci_{\frac{i(i-1)}{2}+j}$. During the D2W bonding, TCI is bonded with BC. Since a top chiplet can only be bonded with one base chiplet, only one item of the subset of TCI, where the items tci_i have the same top chiplets, can be used for bonding. For example, since both tci_2 and tci_4 include top chiplet tc_1, only one of tci_2 and tci_4 can be bonded with base chiplets. Therefore, a conflict set is defined as $CS = \{cs_i \mid 1 \leqslant i \leqslant |CS|\}$ as shown in Fig. 4. Each cs_i is the subset of TCI, i.e., $cs_i \subset TCI$, $\forall cs_i \in CS$. The number of conflicts is the same as the number of top chiplets, i.e., $|CS| = |TC|$.

To realize the desired design purposes, the following constraints should be satisfied. Each tci_i can only be bonded with one base chiplet and thus can be constrained as:

$$\sum_{bc_j \in BC} t_{i,j} \leqslant 1, \ \forall tci_i \in TCI. \tag{5}$$

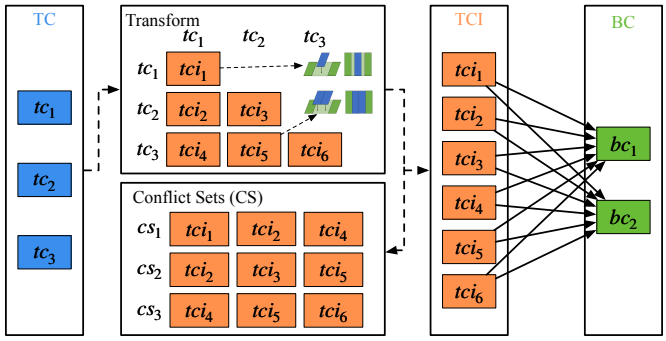


Fig. 4 The illustration of the D2W bonding of OPILP.

For each base chiplet bc_j, it can only be bonded with one top chiplet integration and thus can be constrained as:

$$\sum_{tci_i \in TCI} t_{i,j} \leqslant 1, \ \forall bc_j \in BC. \tag{6}$$

For the top chiplet integrations belonging to the same conflict cs_o, they can only be bonded with one base chiplet and thus can be constrained as:

$$\sum_{tci_i \in cs_o} \sum_{bc_j \in BC} t_{i,j} \leqslant 1, \ \forall cs_o \in CS. \tag{7}$$

For each package product, it should be assigned to the appropriate product bin according to the attributes and thus can be constrained as:

$$\begin{aligned} &pa_{i,j,k} \leqslant p_{i,j,k}, \\ &\forall tci_i \in TCI, \ \forall bc_j \in BC, \ \forall pb_k \in PB. \end{aligned} \tag{8}$$

For each bonding solution, it should be assigned to an appropriate product bin and thus can be constrained as:

$$\sum_{pb_k \in PB} pa_{i,j,k} = t_{i,j}, \ \forall tci_i \in TCI, \ \forall bc_j \in BC. \tag{9}$$

Since the number of products in a product bin pb_k cannot exceed the limited count, it can be constrained as:

$$\sum_{bc_j \in BC} \sum_{tci_i \in TCI} pa_{i,j,k} \leqslant pn_k, \ \forall pb_k \in PB. \tag{10}$$

Finally, the objective can be formulated as:

$$\max \sum_{pb_k \in PB} \sum_{bc_j \in BC} \sum_{tci_i \in TCI} pa_{i,j,k} \times r_k. \tag{11}$$

B. Distributed ILP-Based Method

Since $|TCI| = \frac{|TC|(|TC|+1)}{2}$, the number of TCI is $O(|TC|^2)$, which significantly increases the parameters and constraints in the ILP model of OPILP. In this section, a distributed ILP-based method is proposed to improve the efficiency of OPILP. Therefore, the complexity of ILP models can be effectively reduced.

Algorithm 1 shows the pseudo-code of DILP, and Fig. 5 gives the illustration of DILP. DILP partitions the chiplets into different batches. Each batch has a similar distribution to the original data set by using a fixed-interval extraction method (line 1). In Fig. 5, the data set is partitioned into three batches. Based on the fixed-interval extraction method,

Algorithm 1: Distributed ILP-Based (DILP) Method

Input: batch number bn, propagation limitation pl, top chiplet set TC, base chiplet set BC, product bin set PB, design parameters.
Output: package product set PP.

1 Generate batch set B based on bn ;
2 **for** *each batch* $b_i \in B$ **do**
3 Initialize the parameters and constraints of OPILP model ;
4 Solve OPILP model ;
5 Find the leftover chiplet set LC based on pl ;
6 **if** $i < |B|$ **then**
7 $b_{i+1} \leftarrow LC$;
8 Update the count of PB ;
9 Record current package product set CPP ;
10 $PP \leftarrow CPP$;

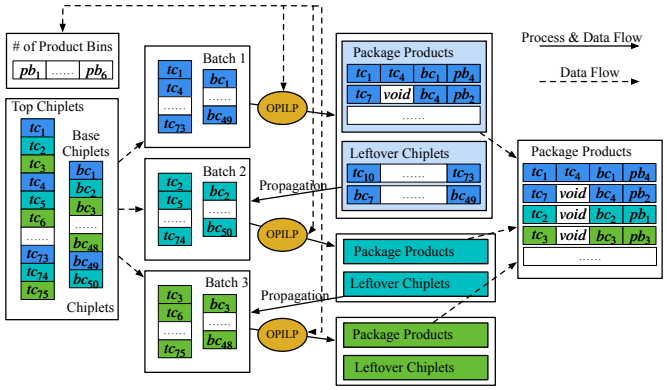


Fig. 5 The illustration of DILP.

the data with the interval of 3, which is the number of batches, is extracted and assigned to the same batch. Since the original data is ordered based on the latency, the data of each batch has a similar distribution to the original data set. In this way, the solution of each batch can approach the solution generated by using OPILP for the original data set.

Then, the data of each batch is processed based on OPILP (lines 3-4). Since the parameters and constraints of the ILP model of each batch are effectively reduced by partitioning the original data set, the complexity of DILP is reduced compared with that of OPILP. The numbers of TCI and BC in each batch are $O((\frac{|TC|}{bn})^2)$ and $O(\frac{|BC|}{bn})$. Furthermore, the quality of the solution of each batch can be guaranteed based on OPILP. To make the solutions of DILP closer to the solutions of OPILP applying for the original data set, a propagation method is proposed. The package products and leftover chiplets are generated after solving the OPILP of each batch. The leftover chiplets with higher performance are selected and propagated to the next batch for improving the quality of global solutions (lines 5-7). The number of selected leftover chiplets, which is pl in Algorithm 1, is defined by users. Finally, the count of the leftover products in each product bin is updated and the solution of each batch is recorded (lines 8-10).

C. Baseline

To modify the previous work to tackle the problem of this work, the baseline with two stages, the first stage of bonding one top chiplet and the second stage of incrementally bonding the second chiplet, are proposed. The previous work [9], [10] can only solve the problem of the first stage. In the first stage, each base chiplet is bonded with one top chiplet based on an ILP model like previous work [9], [10]. In the second stage, the bonded base chiplets are considered to be bonded with the second top chiplets, and the packages are assigned to proper product bins to maximize the total profits based on an ILP model.

In the first stage, each base chiplet is bonded with one top chiplet. The ILP model in this stage has similar formulas with Equations (5) and (6). However, "i-th component" and "j-th component" corresponding to $t_{i,j}$ are related to the top chiplet and base chiplet, respectively. Therefore, the number of the set of "i-th component" is $O(|TC|)$. For each base chiplet, it should be bonded with one top chiplet and thus can be constrained as:

$$\sum_{tc_i \in TC} t_{i,j} = 1, \ \forall bc_j \in BC. \tag{12}$$

For each top chiplet, it can be bounded with one base chiplet or not used and thus can be constrained as:

$$\sum_{bc_j \in BC} t_{i,j} \leqslant 1, \ \forall tc_i \in TC. \tag{13}$$

The objective of the first stage is to minimize the total CPI of the intermediate products. Therefore, the objective can be formulated as:

$$\min \sum_{bc_j \in BC} \sum_{tc_i \in TC} t_{i,j} \times cpi_{i,j}, \tag{14}$$

where $cpi_{i,j}$ represents the CPI of the package bonding bc_j and tc_i. Each intermediate product includes one top chiplet and one base chiplet.

In the second stage, the leftover top chiplets are considered to be bonded with the intermediate products. The ILP model in this stage has similar constraints with Equations (5), (6) and (8) to (10). Equation (7) is not necessary since bonding the second top chiplet incrementally has no conflict with the top chiplets. However, the "i-th component" and "j-th component" of all parameters are related to the leftover top chiplet and intermediate product, respectively. The number of the set of "i-th component" is $O(|TC| - |BC|)$. Since the intermediate product can have no second top chiplet, a component "void" means no top chiplet belongs to "i-th component", and it does not have the constraints of Equation (5). The set of leftover top chiplets can be denoted as $LTC = \{ltc_i \mid tc_i \in TC \ is \ not \ used \ in \ the \ first \ stage\} + \{void\}$. The set of intermediate products can be denoted as $IP = \{ip_j \mid 1 \leqslant j \leqslant |BC|\}$. Each $ltc_i \in LTC$, $ltc_i \neq void$, can only be bonded with one intermediate product and thus can be constrained as:

$$\sum_{ip_j \in IP} t_{i,j} \leqslant 1, \ ltc_i \in LTC, \ ltc_i \neq void. \tag{15}$$

TABLE II The setting of product bins

pbid	sid	Price ($)	Count Ratio	lb_{cpi}	ub_{cpi}
0	0	476	0.125	0.000	0.189
1	0	399	0.250	0.189	0.192
2	0	322	0.125	0.192	0.200
3	1	536	0.125	0.000	0.149
4	1	449	0.250	0.149	0.152
5	1	362	0.125	0.152	0.160

The constraint Equation (15) is not set for $ltc_i = void$. For each intermediate product ip_j, it can be bonded with one leftover top chiplet or $void$ and thus can be constrained as:

$$\sum_{ltc_i \in LTC} t_{i,j} \leqslant 1, \ \forall ip_j \in IP. \tag{16}$$

The objective of the second stage is also to maximize the total profit similar to Equation (11). It can be formulated as:

$$\max \sum_{pb_k \in PB} \sum_{\substack{ltc_i \in LTC \\ ltc_i \neq void}} \sum_{ip_j \in IP} pa_{i,j,k} \times r_k. \tag{17}$$

However, since it is impossible to maximize profits in the first stage as it is not determined whether to bond the second top chiplets, the quality of the baseline is significantly reduced. Experimental results show that DILP can achieve the largest total profit compared with the baseline.

IV. Experimental Results

The proposed methods in this work are implemented in C++ language on a Linux server with 64 GB memory. The Gurobi optimizer [17] is adopted in this work to solve the ILP models. Since the problem can be regarded as bonding chiplets for a base wafer, the testcase is generated for the wafer with 300 mm diameter. The mainstream wafer size is 300 mm. The number of base chiplets in a wafer is calculated based on [18]–[20]:

$$cpw = \lfloor (\frac{\pi w_d^2}{4a}) e^{-\frac{2\sqrt{a}}{w_d}} \rfloor, \tag{18}$$

where cpw, w_d, a represent the number of chiplet per wafer, the diameter of the wafer, and the area of a chiplet, respectively. The number of base chiplets in a wafer is 538. The number of top chiplets is set to 1.5X of the number of base chiplets. As a result, the number of top chiplets is 807.

A. Experimental Setup

In our implementation, there are two SKUs for 64 MB L3-Cache designs, i.e., the package products with one top chiplet, and 128 MB L3-Cache designs, i.e., the package products with two top chiplets. Each SKU has three product bins with different CPI intervals. The CPI of each package product is calculated from the latency of chiplets and bonds based on the architectural simulation method introduced in Section II-B. The setting of product bins is shown in TABLE II. "pbid", "sid", "Price", "Count Ratio", "lb_{cpi}", and "ub_{cpi}" represent the index of product bin, the index of SKU, the price of each product in the bin, the number of package products in each bin to the number of total products, the lower bound of CPI of each product bin, and the upper bound of CPI of each product bin, respectively. The package products belonging to the SKU, whose id is 0, have only one top chiplet. The package products belonging to the SKU, whose id is 1, have two top chiplets. The prices of TABLE II are generated based on the data of [21].

TABLE III The setting of parameters

Parameter	Value	Ref	Parameter	Value	Ref
lat_{bc}	11.49 a.u.	[22]	lat_{bond}	0.90 a.u.	[23]
$pplat_{1tc}$	13.84 a.u.	[22]	$pplat_{2tc}$	12.50 a.u.	[15]
lat_{1tc}	1.45 a.u.	*	lat_{2tc}	0.11 a.u.	*
$area_{bc}$	122 mm^2	[2]	$area_{tc}$	41 mm^2	[2]
w_d	300 mm	[24]	$ppcpi_{1tc}$	0.19	[25]
$ppcpi_{2tc}$	0.15	[15]	MP_{2tc}	0.24	[15]
MP_{1tc}	0.43	[15]	CF	0.50	#
BF	1.10	#	α_1	0.0005	[15]
α_2	0.18	[15]	β_1	0.11	[15]

* Calculated based on the simulation method in the Section II-B.
\# Setting based on the experience.

The parameters used for architectural simulation are shown in TABLE III. The first column and the fourth column show the parameters used by the architectural simulation method. The second column and the fifth column show the values of the parameters. The third column and the sixth column show the sources of values. lat_{bc} and $pplat_{1tc}$ represent the latency of the base chiplet and the package with only one top chiplet, respectively. The values of lat_{bc} and $pplat_{1tc}$ are estimated based on the test results of AMD 3D V-Cache released in [22]. lat_{bond} represents the latency of bonds which is estimated based on the latency of related interconnection components [23]. $pplat_{2tc}$ represents the latency of the package with two top chiplets. The value of $pplat_{2tc}$ is calculated based on the statistics of [15] and $pplat_{1tc}$. lat_{1tc} and lat_{2tc} represent the latency of one top chiplet and the latency of the integration of two top chiplets, respectively. The values of lat_{1tc} and lat_{2tc} are calculated based on Equation (1) and the values of lat_{bc}, lat_{bond}, $pplat_{1tc}$, and $pplat_{2tc}$. $area_{bc}$ and $area_{tc}$ represent the area of a base chiplet and a top chiplet, respectively. The values of $area_{bc}$ and $area_{tc}$ are estimated based on the area of the chiplets of AMD 3D V-Cache [2]. w_d represents the mainstream wafer size [24]. $ppcpi_{1tc}$ represents the CPI of the package product with only one top chiplet which is estimated based on the test results released in [25]. $ppcpi_{2tc}$ represents the CPI of the package product with two top chiplets which is calculated based on the statistics of [15] and $ppcpi_{1tc}$. MP_{1tc}, MP_{2tc}, CF, and BF represent the penalty of the cache miss of the packages with one top chiplet, the penalty of the cache miss of the packages with two top chiplets, the factor for large cache capacity, and the factor for degraded bonding effect, respectively. MP_{1tc} and MP_{2tc} are estimated based on the statistics of [15]. α_1, α_2, and β_1 represent the coefficients of architectural simulation, which are generated based on the statistics of [15].

lat_{bc}, lat_{1tc}, and lat_{2tc}, which are the latency of base chiplet and top chiplet(s) shown in TABLE III, are the standard latency generated from the statistics [15], [22]. The

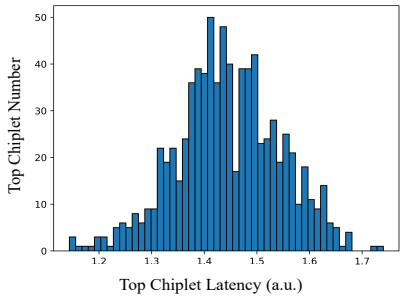


Fig. 6 The distribution of Top chiplet latency.

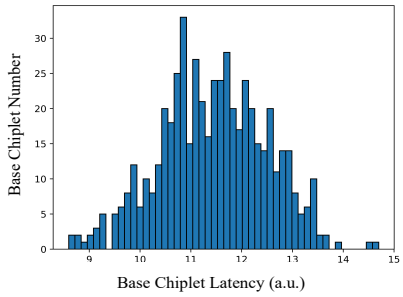


Fig. 7 The distribution of base chiplet latency.

TABLE IV The comparison between DILP and the baseline

Method	Profits ($)	Runtime (s)
Baseline	133412	9813.19
DILP	135874	1646.00
Ratio	1.02	5.96

method introduced in [10], which creates process variation by Gaussian distribution, is used in this work to generate the testcase based on the standard latency values. The latency distribution of top chiplets is shown in Fig. 6. The latency distribution of the base chiplets from a 300 mm wafer is shown in Fig. 7.

B. The Comparison between the baseline and DILP

The experimental results of the baseline and DILP are shown in TABLE IV. The baseline, which is introduced in Section III-C, extends the previous work [9], [10] to fit the problem of this work. Compared with the baseline, DILP can achieve the largest total profits with a 5.96X speedup.

The product distribution of the baseline solution is shown in Fig. 8. The product distribution of the DILP solution is shown in Fig. 9. The first three product bins belong to the SKU with 64 MB L3-Cache. The last three product bins belong to the SKU with 128 MB L3-Cache. The product bins with lower CPI have larger profits. "Max" represents the maximum count of product bins. "Solution" represents the number of the products of the solution in each product bin. Fig. 8 and Fig. 9 show that both the baseline and DILP prefer to construct the products with the medial performance intervals to maximize total profits. However, DILP has the better ability to assemble chiplets for all product bins to achieve larger total profits.

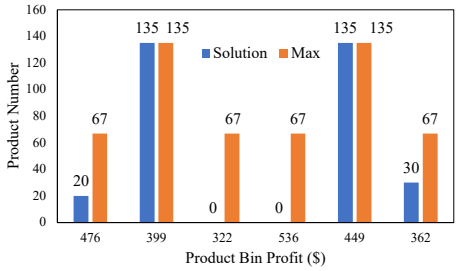


Fig. 8 The distribution of the products of the baseline.

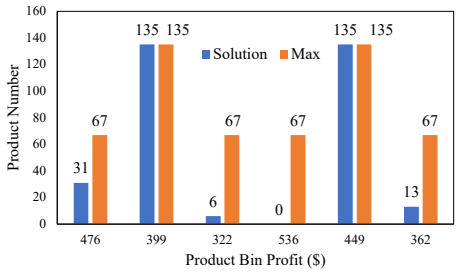


Fig. 9 The distribution of the products of DILP.

C. Experimental Analysis

Based on the distribution of the products of DILP, the following design rules can be summarized:

- Experimental results show that the product bins with medial performance intervals can achieve better total profits. Based on the solutions of DILP, users can change the maximum counts or the performance intervals of product bins to maximize the total profits.
- In this work, the performance interval of each product bin is strict, which leads to the leftover chiplets are not used for any product bin. Based on the solutions of DILP, users can think about strategies to properly use the leftover chiplets to form new products.

V. Conclusion and Future Work

The promising tendency, 3D heterogeneous integration with D2W bonding, creates new challenges. To the best of our knowledge, this is the first work to overcome the new issues of the multi-product optimization problem including flexible chiplet bonding and degraded bonding. A distributed ILP-based method is proposed to efficiently maximize profits and provide hints for users.

In the future, we will generalize this work for multi-layer integration with n top chiplet and consider defect-recoverable banks with less than unit cache size for more SKUs to increase overall profits if marketable segments exist.

Acknowledgment

The research work described in this paper was conducted in the JC STEM Lab of Intelligent Design Automation funded by The Hong Kong Jockey Club Charities Trust.

REFERENCES

[1] Heterogeneous Integration Roadmap (HIR) 2021 Edition, IEEE. Available: https://eps.ieee.org/technology/heterogeneous-integration-roadmap/2021-edition.html.

[2] AMD 3D V-Cache. https://www.amd.com/zh-hans/technologies/3d-v-cache.

[3] J. Wuu, R. Agarwal, M. Ciraula, C. Dietz, B. Johnson, D. Johnson, R. Schreiber, R. Swaminathan, W. Walker, and S. Naffziger, "3D V-Cache: the Implementation of a Hybrid-Bonded 64MB Stacked Cache for a 7nm x86-64 CPU," in *Proceedings of IEEE International Solid-State Circuits Conference (ISSCC)*, vol. 65, pp. 428–429, 2022.

[4] T. Burd, *et al.*, "Zen3: The AMD 2nd-Generation 7nm x86-64 Microprocessor Core," in *Proceedings of IEEE International Solid-State Circuits Conference (ISSCC)*, vol. 65, pp. 1–3, 2022.

[5] W. Gomes, A. Koker, P. Stover, D. Ingerly, S. Siers, S. Venkataraman, C. Pelto, T. Shah, A. Rao, F. O'Mahony and E. Karl, "Ponte Vecchio: A Multi-Tile 3D Stacked Processor for Exascale Computing," in *Proceedings of IEEE International Solid-State Circuits Conference (ISSCC)*, vol. 65, pp. 42–44, 2022.

[6] A. Bond, E. Bourjot, S. Borel, T. Enot, P. Montméat, L. Sanchez, F. Fournel, and J. Swan, "Collective Die-to-Wafer Self-Assembly for High Alignment Accuracy and High Throughput 3D Integration," in *Proceedings of IEEE Electronic Components and Technology Conference (ECTC)*, pp. 168–176, 2022.

[7] G. Gao, L. Mirkarimi, G. Fountain, D. Suwito, J. Theil, T. Workman, C. Uzoh, B. Lee, K. Bang, and G. Guevara, "Die to Wafer Hybrid Bonding for Chiplet and Heterogeneous Integration: Die Size Effects Evaluation-Small Die Applications," in *Proceedings of IEEE Electronic Components and Technology Conference (ECTC)*, pp. 1975–1981, 2022.

[8] News. https://compoundsemiconductor.net/article/115085/EVG_Achieves_Die-to-wafer_Fusion_And.

[9] G. Siddharth and D. Marculescu, "System-Level Process Variability Analysis and Mitigation for 3D MPSoCs," in *Proceedings of IEEE Design, Automation & Test in Europe Conference (DATE)*, pp. 604–609, 2009.

[10] C. Ferri, S. Reda, and R. I. Bahar, "Strategies for Improving the Parametric Yield and Profits of 3D ICs," in *Proceedings of IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, pp. 220–226, 2007.

[11] D.-C. Juan, S. Garg, and D. Marculescu, "Impact of Manufacturing Process Variations on Performance and Thermal Characteristics of 3D ICs: Emerging Challenges and New Solutions," in *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 541–544, 2013.

[12] J. A. Theil, T. Workman, D. Suwito, L. Mirkarimi, G. Fountain, K. Bang, G. Gao, B. Lee, P. Mrozek, C. Uzoh, M. Huynh, and O. Zhao, "Analysis of Die Edge Bond Pads in Hybrid Bonded Multi-Die Stacks," in *Proceedings of IEEE Electronic Components and Technology Conference (ECTC)*, pp. 130–136, 2022.

[13] K. Kennes, A. Phommahaxay, A. Guerrero, S. Suhard, P. Bex, S. Brems, X. Liu, S. Tussing, G. Beyer, E. Beyne, "Carrier Systems for Collective Die-to-Wafer Bonding," in *Proceedings of IEEE Electronic Components and Technology Conference (ECTC)*, pp. 2058–2063, 2022.

[14] Q. Ma, M. D.F. Wong, and K.-Y. Chao, "Configurable Multi-Product Floorplanning," in *Proceedings of Asia and South Pacific Design Automation Conference (ASP-DAC)*, pp. 549–554, 2010.

[15] R. Clapp, M. Dimitrov, K. Kumar, V. Viswanathan, and T. Willhalm, "Quantifying the Performance Impact of Memory Latency and Bandwidth for Big Data Workloads," in *Proceedings of IEEE International Symposium on Workload Characterization*, pp. 213–224, 2015.

[16] Y. Chou, B. Fahs, and S. Abraham, "Microarchitecture Optimizations for Exploiting Memory-Level Parallelism," in *Proceedings of 31st Annual International Symposium on Computer Architecture (ISCA)*, pp. 76–87, 2004.

[17] Gurobi Optimizer. Available: https://www.gurobi.com/.

[18] A. V. Ferris-Prabhu, "An Algebraic Expression to Count the Number of Chips on a Wafer," in *IEEE Circuits and Devices Magazine*, vol. 5, no. 1, pp. 37–39, 1989.

[19] D. K. de Vries, "Investigation of Gross Die Per Wafer Formulas," in *IEEE Transactions on Semiconductor Manufacturing*, vol. 18, no. 1, pp. 136–139, 2005.

[20] A. Agnesina, M. Brunion, J. Kim, A. Garcia-Ortiz, D. Milojevic, F. Catthoor, M. Perumkunnil, and S. K. Lim, "Power, Performance, Area and Cost Analysis of Memory-on-Logic Face-to-Face Bonded 3D Processor Designs," in *Proceedings of IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED)*, pp. 1–6, 2021.

[21] News. https://www.guru3d.com/news-story/amd-ryzen-7-5800x3d-priced-450-other-cpus-to-follow-in-mid-april.html.

[22] News. https://www.tomshardware.com/reviews/amd-ryzen-7-5800x3d-review/2.

[23] U. Pasupulety, B. Halavar, and B. Talawar, "Accurate Power and Latency Analysis of a Through-Silicon Via (TSV)," in *Proceedings of International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pp. 688–694, 2018.

[24] News. https://anysilicon.com/does-size-matter-understanding-wafer-size/.

[25] News. https://hothardware.com/news/amd-claims-zen-2-could-offer-a-combined-fpu-and-integer-ipc-boost-of-29-percent.