

# A Globally-optimized Co-design Approach for Heterogeneous Systems Using Convex Optimization

Tilman Horst\*, Robert Fischbach† and Jens Lienig‡

Institute of Electromechanical and Electronic Design, Dresden University of Technology  
Dresden, Germany

Email: \*tilman.horst@posteo.de, †robert.fischbach@tu-dresden.de, ‡jens@ieee.org

**Abstract**—Manufacturing methods for heterogeneous systems allow for high-density integration of multiple semiconductor devices in one system or package. Current design tools and approaches do not incorporate the intensified heterogeneity of 3D systems sufficiently. They provide suboptimal design solutions because the overall global solutions cannot be optimized. In this paper, we present a co-design methodology to globally solve and optimize the design problem and so to fully exploit the potential of emerging technologies. We use a design example to show the advantage and potential of a co-design methodology compared to conventional top-down approaches. Our approach should guide further research and tool development in this emerging field of heterogeneous system design.

**Index Terms**—co-design, heterogeneous systems, physical design, convex optimization, EDA

## I. INTRODUCTION

Today’s digitally dominated microelectronic systems are for the most part custom-made, containing only few off-the-shelf components such as input/output (I/O) cells or analog hard intellectual property (IP). Even digital IP cores are usually delivered as synthesizable soft blocks that can be adapted by the designer to fit properly into the system. This freedom enables designers to design competitive systems meeting the most challenging requirements. However, design freedom also stands for large and complex solution spaces that need to be controlled by designers to find satisfying solutions.

Top-down design is a popular approach for design problems with a high degree of customizable components (Fig. 1, left). It starts with the specification and gradually increases the level of detail, defining the sub-structure of the system. With regard to physical design of microelectronic systems, top-down approaches begin by defining the footprint of a module, i.e. its sizes and external pin positions, before proceeding with the “inner parts”. Information and data flow in one direction from the top to the bottom of the design hierarchy not allowing for bidirectional adaptations. While top-down approaches enable best solutions within separate design domains (such as integrated circuit (IC), micromechanical system (MEMS), printed circuit board (PCB)), they do not allow for globally optimal solutions across hierarchy boundaries (aka “design levels”).

With emerging three-dimensional (3D) technologies, which enable the combination of, for example, various dies on an interposer, the complexity of design problems is further growing. At the same time, to justify the high investment costs that come along with 3D technologies, it becomes increasingly important

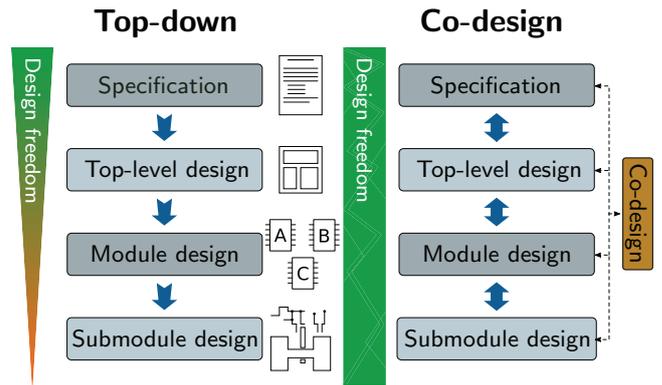


Fig. 1: Classical top down approach versus co-design. Co-design provides unrestricted design freedom during all stages of the design process, leading to better solutions for hierarchical design problems.

to fully exploit the potential revealed by novel technologies. As pointed out before, top-down approaches neglect potentials for design optimization – which is why new and more suitable design approaches are required [1], [2], [3].

In this paper, we propose co-design as an alternative to top-down approaches for the design of heterogeneous systems. In a co-design process, sub-systems (e.g., die, interposer, PCB) are not designed independently, but with the “neighboring” design levels “in mind”. Thereby, design freedom is maintained throughout the design process and globally optimal solutions are enabled (Fig. 1, right).

We present an iterative co-design algorithm, which cycles through the design domains and continuously refines domain-specific design solutions using convex optimization. During this process, each solution is temporary, and altered according to future changes in connected (neighboring) design levels. That is, the design freedom is not restricted prematurely and, in fact, is preserved throughout the design process. In short, our co-design methodology finds best solutions for hierarchical design problems irrespective of early-stage design decisions. As further pointed out in Section IV, it converges quickly, finding good solutions in a short period of time.

Please cite as:

T. Horst, R. Fischbach, J. Lienig, "A Globally-optimized Co-design Approach for Heterogeneous Systems Using Convex Optimization", Proc. 2020 European Conf. on Circuit Theory and Design (ECCTD), 7-10 Sept. 2020, Sofia, Bulgaria, DOI: 10.1109/ECCTD49232.2020.9218373

The contributions of this paper are:

- introduction of a co-design example to visualize the optimization potential of top-down versus co-design methodologies,
- presentation of a co-design approach based on convex optimization techniques that provides globally optimized and non-overlapping designs, and
- evaluation of potential benefits of co-design compared to conventional top-down design approaches.

## II. PROBLEM FORMULATION

The essence of co-design is that optimizations at multiple hierarchical levels need to be tuned towards the *globally* optimal solution. To illustrate this problem, we define a minimum (artificial) co-design problem as depicted in Fig. 2. Our problem consists of four dies  $d_i$  that are mounted on an interposer  $I$ . Each die contains four circuit blocks  $b_i$  that are interconnected with circuit blocks on the same die and using pins  $p_i$  also with circuit blocks on other dies. We assume the space within the dies, which is not occupied by circuit blocks, is filled up with standard cells.

The goal of our co-optimization is to find non-overlapping positions for all dies  $d_i$ , circuit blocks  $b_i$ , and pins  $p_i$  that *globally* minimize the wire length (WL). That is, during the optimization of design levels, not only the current design level itself, but also adjacent design levels need to be taken into account. If all dies  $d_i$ , circuit blocks  $b_i$  and pins  $p_i$  are referred to as entities  $e_i$ , we can express the objective of our co-design mathematically as follows:

$$\min \sum_{(i,j) \in \Lambda} \|e_i - e_j\|_2^2 \quad (1)$$

In Eq. 1,  $(i, j)$  refers to a link between entities  $i$  and  $j$ . The term  $\Lambda$  refers to the set of links between entities and  $\|e_i - e_j\|_2^2$  refers to the Euclidean norm, i.e. the distance between two entities  $e_i$  and  $e_j$  in  $\mathbf{R}^2$ . We further define that a die must not overlap another die. Circuit blocks (red) have to lie within the dies and must not overlap each other. Package pins (black) have fixed positions, die pins (yellow) can move freely along their assigned edge.

Obviously, to find the optimal placement of blocks on a die, the placement of dies on the interposer ought to be considered, as the latter influences the external pin connections of each die (which, in turn, influence the placement of the blocks on the die). This “chicken-or-egg dilemma” is characteristic of co-design problems. We show that using an appropriate optimization methodology resolves this dilemma and enables to find globally optimal design solutions. By solving this kind of problem (related to NP-complete facility layout problems [4]) globally optimal, the benefits and optimization potential of co-design can then be evaluated.

## III. CO-DESIGN ALGORITHM USING CONVEX OPTIMIZATION

The development of complex heterogeneous systems requires designers from various engineering disciplines and

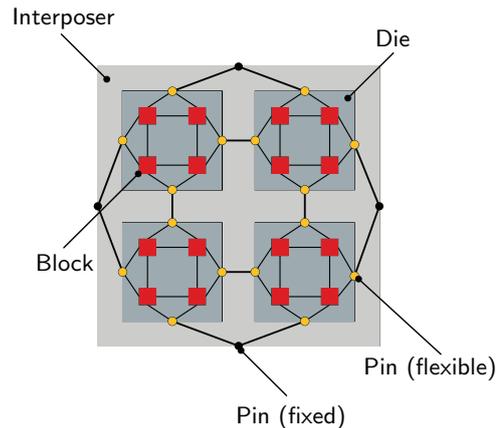


Fig. 2: Illustration of our co-design problem depicting the interposer, dies, circuit blocks and pins.

design tools from different vendors. Co-design should not replace or unify established tools, but orchestrate their activities towards global objectives and across tool and company boundaries. Having these demands in mind, we propose a two-stage approach. The goal of the first stage is to optimize the positions of external pins for each die. The goal of the second stage is to optimize the inner parts of the dies. Since both optimizations depend on each other, we alternate them until a nearly optimal solution is obtained. The overall objective is to minimize wire length across all hierarchy levels. The advantage of separating the local optimization of dies and the global optimization of external pins is that each die can be optimized independently from other dies. This reduces complexity and provides the opportunity to use whichever optimization method is appropriate for a specific module – an important property to master heterogeneity.

Our algorithm begins with an initial placement which can, for example, be generated using a conventional top-down approach, followed by the described optimization stages. An overview of our algorithm is depicted in Fig. 3. We use convex optimization techniques to perform the optimizations during each iteration. An important property of convex problems is that any local minimum is also a global minimum. Moreover, convex optimization problems can be solved very efficiently. Problems with hundreds of variables and thousands of constraints can be solved on current desktop computers in a matter of seconds [5].

In the remainder of this section, we discuss the convex optimization technique and constraints utilized during co-design in more detail.

### A. Objective Function

We describe our above discussed problem as an undirected graph with  $N$  nodes. With each node we associate a variable  $x_i \in \mathbf{R}^2$ . Depending on the current stage of the optimization (Fig. 3), the variables represent pin positions or circuit block positions.

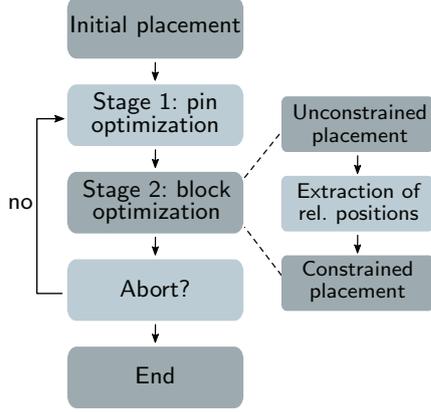


Fig. 3: Overview of the proposed co-design algorithm. Stage 1 and Stage 2 represent the main steps of the optimization.

In Eq. 2, the optimization problem is defined as follows:

$$\min \sum_{(i,j) \in \Lambda} w_{ij} f_{ij}(x_i, x_j) \quad (2)$$

The function  $f_{ij}$  calculates the cost, i.e. a measure of the wire length, of the link between nodes  $i$  and  $j$ . The term  $w_{ij}$  assigns weights to all links between nodes  $i$  and  $j$  ( $w_{ij} \geq 0$ ). Given the functions  $f_{ij}$  are convex, this is a convex optimization problem [5].

We use the Euclidean norm to calculate the cost of a link  $(i, j)$  resulting in a *nonlinear placement* or (more precisely) in a *quadratic placement problem*:

$$f_{ij}(x_i, x_j) = \|x_i - x_j\|_2^2 \quad (3)$$

Since Eq. 3 is an increasing and convex function, the optimization problem Eq. 2 is convex.

### B. Additional Placement Constraints

During optimization of external pins (Stage 1), additional constraints restrict the pins to lie on the edges of respective dies. During optimization of dies (Stage 2), further constraints prevent overlapping of components.

1) *Constraints for Pin Placement*: Since as per definition pins are attached to the border of a die, pins which are assigned to the same die cannot move independently from each other. To reflect this behavior within our optimization and to constrain the relative position of one pin with respect to other pins of the same die, we define additional equality and inequality constraints.

Figure 4 shows a die with four pins (or nodes)  $p_i$  assigned to the edges of the die. In this example, the variable  $x_{i0}$  refers to the x-coordinate of node  $i$  and the variable  $x_{i1}$  refers to the y-coordinate of node  $i$ . The entirety of variables of the example can be represented by a vector  $x$  as Eq. 4:

$$x = (x_{00} \ x_{10} \ x_{20} \ x_{30} \ x_{01} \ x_{11} \ x_{21} \ x_{31})^T \quad (4)$$

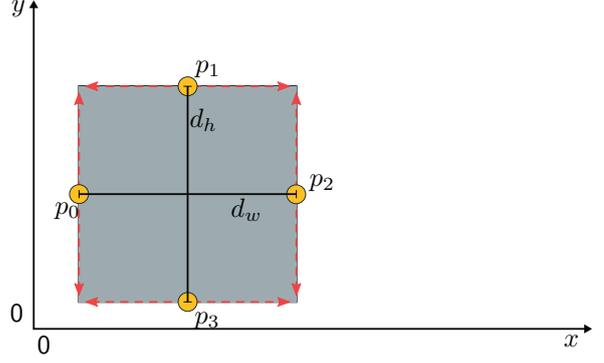


Fig. 4: Pin position constraints of a die with four constraints.

The equality constraints Eq. 5 and Eq. 6 constrain the distance between pins  $p_0$  and  $p_2$  in horizontal as well as  $p_1$  and  $p_3$  in vertical direction.  $d_w$  refers to the width and  $d_h$  to the height of the die which contains the pins as follows:

$$x_{20} = x_{00} + d_w \quad (5)$$

$$x_{11} = x_{31} + d_h \quad (6)$$

Equality constraints Eq. 5 and Eq. 6 in standard form can be rewritten in matrix notation as  $Ax = b$  with

$$A = \begin{pmatrix} -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 \end{pmatrix}$$

and

$$b = \begin{pmatrix} d_w \\ d_h \end{pmatrix}.$$

In this notation,  $A$  represents the coefficients of the constraints and  $b$  the values on the right hand side of the equations.

Further constraints are required to restrict the dies to lie within intervals that correspond to the die width  $d_w$  and height  $d_h$ . Equations 7 restrict nodes 1 and 3 to lie within the interval that is restricted by the left and right border of the die. Accordingly, Eqs. 8 restrict nodes 0 and 2 to lie within the interval restricted by the upper and lower bound of the die as follows:

$$\begin{aligned} x_{10} &\geq x_{00} \\ x_{10} &\leq x_{00} + d_w \\ x_{30} &\geq x_{00} \\ x_{30} &\leq x_{00} + d_w \end{aligned} \quad (7)$$

$$\begin{aligned} x_{01} &\leq x_{31} + d_h \\ x_{01} &\geq x_{31} \\ x_{21} &\leq x_{31} + d_h \\ x_{21} &\geq x_{31} \end{aligned} \quad (8)$$

The inequality equations can be expressed in standard form and rewritten in matrix notation as  $Gx \preceq h$  with

$$G = \begin{pmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \end{pmatrix}$$

and

$$h = (0 \quad d_w \quad 0 \quad d_w \quad d_h \quad 0 \quad d_h \quad 0)^\top.$$

The matrix  $G$  represents the coefficients of the inequality constraint equations and the vector  $h$  the right hand side of the equations. The notation  $Gx \preceq h$  denotes componentwise vector inequality, i.e. every entry of the vector  $Gx$  is less than or equal to the corresponding entry of the vector  $h$ .

2) *Non-overlap constraints*: If  $(x_{i0}, x_{i1}), (x_{j0}, x_{j1}), w_i, h_i, w_j$  and  $h_j$  are the coordinates, widths and heights of two rectangular circuit blocks, the non-overlap constraints for each pair can be expressed as Eq. 9 for the separation in x-direction and Eq. 10 for separation in y-direction as follows:

$$\frac{1}{2}(w_i + w_j) - |x_{i0} - x_{j0}| \leq 0 \quad (9)$$

$$\frac{1}{2}(h_i + h_j) - |x_{i1} - x_{j1}| \leq 0 \quad (10)$$

To prevent blocks  $i$  and  $j$  from overlapping at least one of the constraints Eq. 9 or Eq. 10 must be fulfilled. However, the constraints evaluate absolute values  $|x_{i0} - x_{j0}|$  and  $|x_{i1} - x_{j1}|$  and, thus, are non-linear and non-convex. Eliminating the symbols of absolute values results in two disjunctive constraints, i.e. the problem becomes unresolvable.

A possible solution is to define a set of constraints that already reflect relative position of blocks. This allows to eliminate the symbols of absolute values at the cost of loss of generality. In order to find a reasonable set of non-overlapping constraints, we introduce a preliminary optimization step. During this step, we omit all non-overlapping constraints, which results in a placement that contains overlaps, but allows to deduce appropriate relative positions of blocks. For the actual optimization, we convert relative positions obtained from the pre-processing step into non-overlapping constraints. Subsequently, we run the optimization again, but this time considering non-overlapping constraints. The outcome is the optimal non-overlapping placement based on the relative positions determined during pre-processing. Six relative positions to fully describe the placement of blocks in the example of Fig. 5 are:

- block 0 is on the left of block 3
- block 0 is on the left of block 2
- block 1 is on the left of block 3

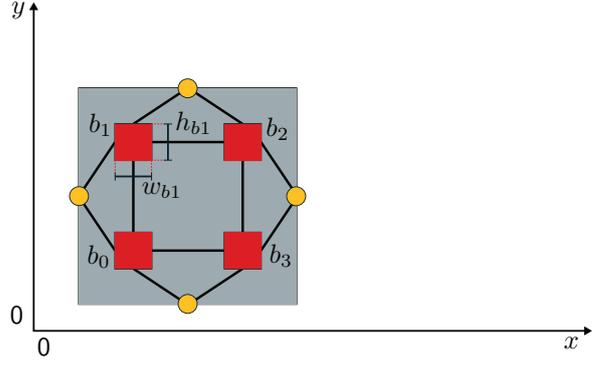


Fig. 5: Visualization of block position constraints of four blocks placed on a die.

- block 1 is on the left of block 2
- block 0 is below block 1
- block 3 is below block 2

Based on these predefined relative positions, we define linear and convex non-overlapping constraints listed in Eq. 11 and consider them during the convex optimization of block positions.

$$\begin{aligned} x_{00} + \frac{1}{2}w_{b0} &\leq x_{30} - \frac{1}{2}w_{b3} \\ x_{00} + \frac{1}{2}w_{b0} &\leq x_{20} - \frac{1}{2}w_{b2} \\ x_{10} + \frac{1}{2}w_{b1} &\leq x_{30} - \frac{1}{2}w_{b3} \\ x_{10} + \frac{1}{2}w_{b1} &\leq x_{20} - \frac{1}{2}w_{b2} \\ x_{01} + \frac{1}{2}h_{b0} &\leq x_{11} - \frac{1}{2}h_{b1} \\ x_{31} + \frac{1}{2}h_{b3} &\leq x_{21} - \frac{1}{2}h_{b2} \end{aligned} \quad (11)$$

The inequalities listed in Eq. 11 can be expressed in matrix notation as:

$$G = \begin{pmatrix} 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \end{pmatrix}$$

and

$$h = \begin{pmatrix} -\frac{1}{2}(w_{b3} + w_{b0}) \\ -\frac{1}{2}(w_{b2} + w_{b0}) \\ -\frac{1}{2}(w_{b3} + w_{b1}) \\ -\frac{1}{2}(w_{b2} + w_{b1}) \\ -\frac{1}{2}(h_{b1} + h_{b0}) \\ -\frac{1}{2}(h_{b2} + h_{b3}) \end{pmatrix}.$$

### C. Convex Optimization Solver

To solve Eq. 3 subject to the constraints described in Subsection III-B, we convert the problem into the standard form as shown in Eq. 12:

$$\begin{aligned} \min \quad & \frac{1}{2}x^\top Px + q^\top x \\ \text{s.t.} \quad & Ax = b, \\ & Gx \preceq h \end{aligned} \quad (12)$$

All inequalities, links and positions of fixed pins are collapsed into matrices  $P, q, A, b, G$  and  $h$  and as such can be solved using the solver for quadratic programs provided by [6]. The matrices containing the constraint coefficients  $A, G$  as well as the vectors  $b, h$  are generated as pointed out in Subsection III-B.

## IV. EXPERIMENTAL RESULTS

The goal of our experimental analysis was to find out to what extent co-design outperforms a straightforward top-down approach, using the example of global wire length (WL). For the experiments we applied physical dimensions for the dies, package and blocks as specified in Table I. The parameters *die spacing* and *block spacing* refer to the space that separates adjacent blocks and dies.

TABLE I: Physical Dimensions of Components

	x [mm]	y [mm]
<b>package size</b>	65	65
<b>die size</b>	24	24
<b>block size</b>	6	6
<b>die spacing</b>	1	1
<b>block spacing</b>	0.5	0.5

### A. Results Using a Top-down Approach

We implemented a top-down approach consisting of three steps (Fig. 6). The goal of the first step is to find optimal positions for all dies on the interposer. Then, once the die positions are known, each pin is placed at an appropriate pin position. Lastly, once the positions of external pins are known, the circuit blocks within the dies are placed.

Figure 7 shows the placement that results when the top-down approach is applied to the design example described in Section II. After die placement, the the red pins are located as close as possible to the package pins in order to minimize WL. Finally, given the positions of the red pins, the blocks are placed optimally within the dies. In our example, using a top-down approach, the quadratic global WL is  $5516 \text{ mm}^2$ .

### B. Results Using Co-design

We used the top-down placement in Fig. 7 as initial placement for the co-design flow according to Fig. 3. After five iterations, the total squared WL was reduced to  $4676.68 \text{ mm}^2$ , which equals a reduction of 15% compared to the top-down

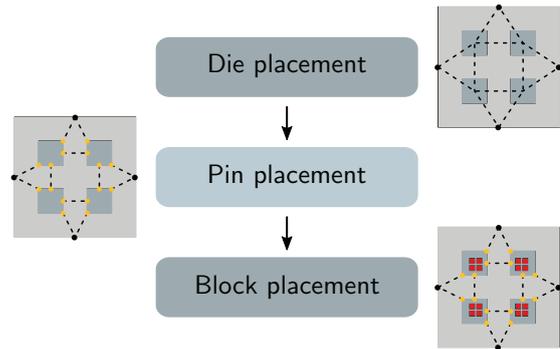


Fig. 6: Proposed top-down flow.

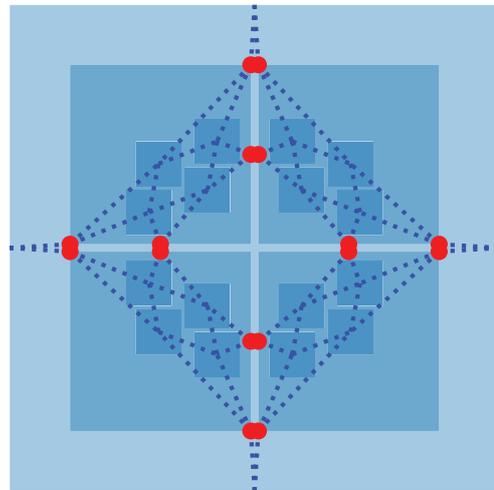


Fig. 7: Top-down placement. The red pins are placed in the die corners to minimize package-level wire length.

placement. Figure 8 shows the initial placement and the placement after one, two and five iterations. Comparing the initial placement in Fig. 8a with the others, we observe that the positions of red pins have changed. As a result, the inter-die wire length (blue triangles) has increased. However, the placement of blocks within the dies benefits from the changed pin positions, resulting in a reduction of the *total* wire length. That is, our algorithm sacrificed the optimality of (local) interposer-level routing to achieve a shorter global WL.

## V. CONCLUSION

In this paper, we introduced a minimum (artificial) design example in order to show that a co-design methodology achieves better results than a conventional top-down approach. We could significantly reduce the global wire length in hierarchical system compositions in our example.

We presented an algorithm that effectively solves co-design problems that arise if multiple components are designed in parallel. Our algorithm results in a wire-length reduction of 15% compared to a layout generated via a rather conventional top-down approach. It enables the globally-optimized

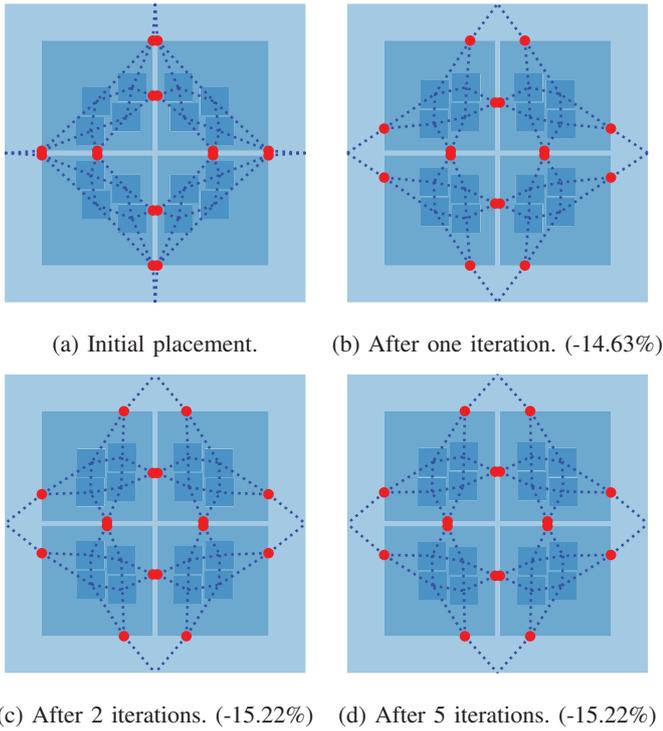


Fig. 8: Initial placement and placements after 1, 2 and 5 iterations. The numbers in parentheses indicate the reduction in wire length compared to the initial placement after each number of iterations.

placement of pins, circuit blocks and dies across multiple hierarchical levels and provides opportunities to incorporate various additional constraints. Each step of the algorithm can also be accomplished by external design tools and algorithms. The output of external tools can easily be fed back and considered by our iterative co-design methodology. The approach is based on convex optimization techniques and can be extended to efficiently solve larger problems with hundreds of variables (Fig. 10).

Our example showed the advantages and potential of a co-design methodology compared to conventional (top-down) approaches that do not take “neighboring” design levels into account. Our presented methodology should guide further research and tool development in the emerging field of heterogeneous systems design.

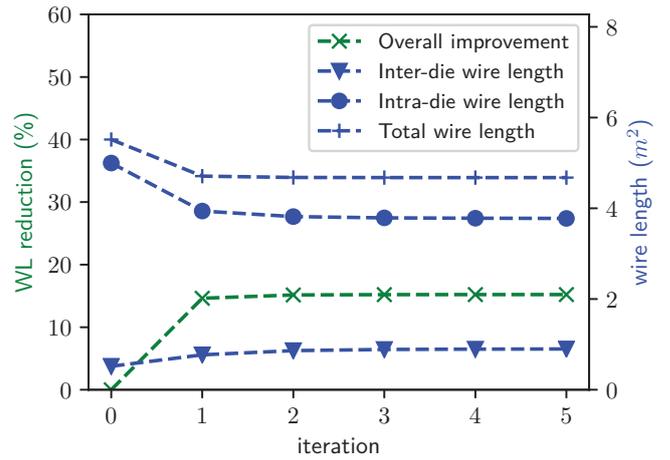


Fig. 9: Improvement over the initial top-down placement after 1 to 5 iterations. While the inter-die wire length increases, the total wire length (including inter- and intra-die connections) decreases.

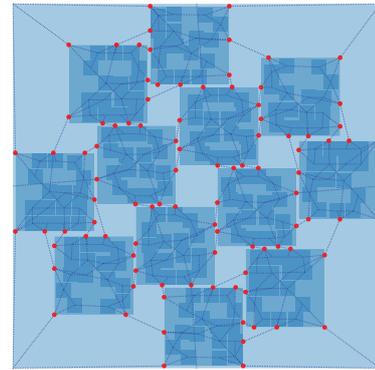


Fig. 10: Placement results with 12 dies and 16 circuit blocks on each die. The resulting optimization problem consists of 492 optimization variables and about 500 constraints. This solution was reached on a Intel(R) Xeon(R) CPU @ 3.00GHz after five iterations in about 10 seconds.

## REFERENCES

- [1] J. Knechtel, O. Sinanoglu, I. M. Elfadel, J. Lienig, and C. C. N. Sze, “Large-scale 3D chips: Challenges and solutions for design automation, testing, and trustworthy integration,” *IPSI Transactions on System LSI Design Methodolog*, vol. 10, pp. 45–62, Aug. 2017.
- [2] J. Knechtel, E. F. Y. Young, and J. Lienig, “Planning massive interconnects in 3-D chips,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 34, no. 11, pp. 1808–1821, Nov. 2015.
- [3] R. Fischbach, J. Lienig, and T. Meister, “From 3D circuit technologies and data structures to interconnect prediction,” in *Proc. of 2009 Int. Workshop on System Level Interconnect Pred. (SLIP)*, 2009.
- [4] S. S. Heragu and A. Kusiak, “Efficient models for the facility layout problem,” *European Journal of Operational Research*, 1991.
- [5] S. Boyd and L. Vandenberghe, “Convex optimization.” Cambridge University Press, 2009.
- [6] M. S. Andersen, J. Dahl, and L. Vandenberghe, “Cvxopt: A python package for convex optimization, version 1.2,” 2019. [Online]. Available: cvxopt.org