The ISPD2005 placement contest and benchmark suite

Conference Paper · January 2005

DOI: 10.1145/1055137.1055182 · Source: DBLP

5 authors, including:

Gi-Joon Nam
IBM
100 PUBLICATIONS **2,697** CITATIONS
SEE PROFILE

C.J. Alpert
IBM
187 PUBLICATIONS **7,109** CITATIONS
SEE PROFILE

Paul Villarrubia
IBM
46 PUBLICATIONS **1,120** CITATIONS
SEE PROFILE

The ISPD2005 Placement Contest and Benchmark Suite

Gi-Joon Nam, Charles J. Alpert, Paul Villarrubia, Bruce Winter and Mehmet Yildiz

IBM Corp.
11400 Burnet Road
Austin TX 78758

{gnam, alpert, pgvillar, bbw, mcan}@us.ibm.com

ABSTRACT

Without the MCNC and ISPD98 benchmarks, it would arguably not have been possible for the academic community to make consistent advances in physical design over the last decade. While still being used extensively in placement and floorplanning research, those benchmarks can no longer be considered representative of today's (and tomorrow's) physical design challenges. In order to drive physical design research over the next few years, a new benchmark suit is being released in conjunction with the ISPD2005 placement contest. These benchmarks are directly derived from industrial ASIC designs, with circuit sizes ranging from 210 thousand to 2.1 million placeable objects. Unlike the ISPD98 benchmarks, the physical structure of these designs is completely preserved, giving realistic challenging designs for today's placement tools. Hopefully, these benchmarks will help accelerate new physical design research in the placement, floorplanning, and routing.

Categories and Subject Descriptors

B.7.2 [Hardware]: INTEGRATED CIRCUITS – *Design Aids*; G.4 [Mathematical Software]: Algorithm Design and Analysis; J.6 [Computer Applications]: COMPUTER-AIDED ENGINEERING – *Computer-Aided Design*.

General Terms

Algorithms, Performance, Design, Experimentation.

Keywords

VLSI Placement, Benchmarks, Physical design.

1. INTRODUCTION

Research in placement algorithms for VLSI circuits has enjoyed a renaissance in recent years. For example, there are several new academic placers that use either top-down cut-based placement [1,10,17], an analytical paradigm [6,7,8,9,15], or simulated-annealing [14]. The amount of research on this topic reflects the importance of the placement as the single most critical component for achieving timing/design in a modern physical synthesis. It is hard to believe that such advances would have been possible without publicly available standard circuit benchmarks such as the MCNC [5] and ISPD98 [2] suites.

Despite these publicly available testcases, it is still difficult to accurately compare one placement tool to another, especially as more and more placement tools are developed [12]. One of the reasons for this difficulty is that each group tends to interpret the existing benchmarks in somewhat different ways, or even modify them according to their capability. For example, a common practice seems to be to change physical sizes of objects in benchmarks so that no macro cell exists. These modifications lead to several variations of the original benchmark and renders fair comparisons very challenging.

The ISPD98 [2] suite is a primary culprit for this confusion since these designs were released without complete physical information. In order to secure permission for their release, the benchmarks had to be doctored: large nets are omitted, the I/O cells are not valid, cell dimensions are not given, and the placement image space is not specified. The Dragon benchmarks (typically known as IBM-PLACE benchmarks) [17] are a good example of an effort to create placement instances from the ISPD98 suite. However, to address the standard cell placement, big macro blocks with attached nets were intentionally removed. Because all I/O objects had connection to these big macros (i.e., no direct nets to standard cells), I/O cells became dangling objects in the Dragon benchmarks. These twists provide a biased perception of the real physical design problems that placers have to face today. Finally, they have been around for several years now, and the size of benchmarks can hardly be considered to reflect modern VLSI designs.

To further aid future advances in placement, this paper announces the release of the ISPD2005 benchmark suite. While primarily intended for placement, they certainly can be used for floorplanning and routing as well. These benchmarks are directly derived from modern industrial ASIC designs and are being used in the first ISPD placement contest. The primary goals of placement contest are:

- to provide new modern placement benchmarks to stimulate new development in placement research,
- to encourage the academic community to expose their placement tools and results to the public in order to make a better quality comparisons,
- and to provide an educational forum on variety of state-of-the-art placement algorithms for future placement researchers

The rest of this paper is organized as follows. Section 2 briefly discusses logistics of placement contest. Section 3 describes the benchmark characteristics and translation process, and Section 4 presents layout figures and analysis on these designs. Concluding remarks follow in Section 5

ISPD'05, April 3-6, 2005, San Francisco, California, USA.
Copyright 2005 ACM 1-59593-021-3/05/0004...$5.00.

2. ISPD 2005 PLACEMENT CONTEST

The impetus for the debut of the ISPD placement contest came from the steering committee. The ability to secure permission to release the physical footprint of the designs to the public made it come to fruition. Real industrial ASIC designs from IBM and Adaptec[1] are used for the placement contest. Each circuit contains several fixed/movable macros, standard cells and I/O cells. These designs are translated from IBM's internal data format into the GSRC Bookshelf placement format [13]. To make sure the placers handle this complexity that they likely have not seen before, two (*adaptec1* and *adaptec3*) of the eight benchmarks were released to the contestants two months before the contest.

Each contestant downloads the six remaining benchmarks from the contest website and has five days to run their placement tool on them in whatever manner they choose. After five days, the contestants submit their best results for each test case, along with their placement tool binary and run-scripts. The contest coordinators verify that each tool is able to actually generate the submitted result. However, a runtime comparison is not performed. Giving a fixed window of time, but as many CPU resources as the contestants can muster actually reflects the reality of delivering a design as quickly as possible.

The benchmarks do not contain timing information (timing makes the releasing the designs an order of magnitude more complex). The quality of placement is measured by two physical factors:

- *legality, and*
- *wirelength*.

For a placement solution to be legal, it must satisfy the following four conditions:

- all fixed objects are still in their original locations,
- all moveable objects are placed inside the placement region,
- all objects are aligned with row boundaries, and
- all objects do not overlap.

Any solution that violates any of these conditions is considered an illegal solution and is discarded.

Wirelength is measured according to half-perimeter bounding box wirelength (HPWL). HPWL is widely used when comparing results in placement and floorplanning research. HPWL is easy to measure and there is no ambiguity that may cause different research groups to get different results (unlike Steiner trees). In spite of its simplicity, HPWL is still a reasonable first-order estimation for routability (for easy nets). For some difficult nets, steiner-tree wirelength might be necessary to optimize for better routability, but the number of those nets is typically only a small portion. Although there can be some concerns on congestion, one solution to congestion mitigation is to inflate cells to reserve more routing resources around them [4]. After cells are expanded, a placement algorithm must be able to provide very compacted placement solutions. In this scenario, HPWL is still valid metric to measure placement quality. Pin offsets are respected for HPWL measurement. In other words, it is pin-to-pin half perimeter bounding box wirelength, as opposed to center-to-center HPWL.

Independent perl scripts were used for legality checking and calculating HPWL and can be downloaded from the placement contest website: http://www.ispd.cc/contest.htm.

Of course, it would have been better to also assess routability, congestion, timing, and even manufactuability for quality metrics. For routability and congestion, the reality is that, at this time, we lack a standard routers and unified design rules (i.e., the number of metal layers, track pitches etc.) to verify placement solutions, and these metrics are excluded at this time. Recently, however, promising research results have been reported in this area [11,16] and in near future placement contests, it is hope that routability and congestion analysis, at least in the simplified model, will be included.

The Bookshelf placement format does not permit the ability to specify functionality and delay of objects, unit resistance, capacitance for different metal layers that are necessary to extract timing results of placement solutions such as the worst path slack and the number of negative slack paths. After all, timing is meaningless without performing entire physical synthesis flow, and there is no clear way of containing timing within the placement context. Thus, including timing information remains as future work.

We hope that the ISPD 2005 placement contest serves as a forum for better placement comparisons and provides key statistics on placement quality for the future placement research.

3. ISPD2005 BENCHMARK SUITE

Table 1 summarizes the characteristics of circuits released as ISPD2005 benchmark suite. The reported statistics are:

- *#Total Objects*, the total number of objects of design
- *#Mov. Objects*, the number of movable objects
- *#Fixed Objects*, the number of fixed objects
- *#Nets*, the number of nets
- *#Total Pins*, the total number of pins
- *#Pins.Mov.Objs*, the number of pins from movable objects
- *#Pins.Fixed.Objs*, the number of pins from fixed objects
- *Design Density*, the density of design, i.e., the ratio of the area sum of total objects over the area of placement region
- *Design Util.*, the design utilization defined as the area sum of only movable objects divided by available free space (i.e., total placement area – the area sum of fixed objects)
- *Peri. I/Os*, the number of perimeter I/O objects. If this number is 0, a design contains only area-array I/Os.

The height of the object is always an integer multiple of circuit row height so that every object can be placed aligned to standard cell row boundary. We view *Design Util.* as more reflective of design density information because the placement area covered by fixed blocks is not usable for placing movable objects. The quoted *Design Util.* numbers also confirm that in modern ASIC designs, there is abundant free space available to place movable objects [3]. Because raw designs are in IBM internal formats initially, they are read into IBM CPLACE first, and then CPLACE produces designs in the GSRC Bookshelf placement format. The following issues deserve to be discussed further for the translation of designs.

[1] The Adaptec designs were developed inside IBM. Subsequently, the intellectual property rights to them were sold to Adaptec.

3.1 Movable Macros

A movable macro is a movable object whose height is larger than a single standard cell row height. At this time, there are only a few academic placers with the true capability of handling movable macros. Indeed, movable macros can cause serious problems, particularly during legalization. We intentionally changed movable macros, if there are any, into fixed macros by pre-placing them using CPLACE. With this preprocessing, in most of the benchmarks, the only movable objects are standard cells in the ISPD2005 benchmark suite. However, *bigblue3* contains 2485 movable macros which slightly differentiate a placer with this capability. For the future placement research, however, by changing "FIXED" types into movable for any macro objects, the benchmarks can be easily modified to include movable macros. Fixed macros with two or three standard row heights are good targets for this because they are more likely to be latch or clock-related objects with some flexibility of relocations. Extraordinary large objects that are taller than five or more row heights are not recommended for change because these objects are preplaced prior to placement.

3.2 I/O Objects

Area-arrays I/Os, as opposed to perimeter I/Os, are becoming standard in modern ASIC designs. In this benchmark suite, five of the eight designs have area-arrays I/Os while perimeter I/Os only exist in *adaptec1*, *adaptec2* and *bigblue1*, which are rather small designs. In general, area-array I/O cells are allowed to overlap with other fixed/movable objects in a design. This is because I/O objects resides on separate layers from routing metal layers. However, translating this information fully in GSRC Bookshelf placement format was not trivial. Hence, for the translation of released benchmarks, all I/O cells were defined as fixed and treated equally with logic fixed blocks. During this process, a few overlaps naturally occurred among fixed blocks. These overlaps were removed by relocating smaller fixed blocks, regardless of that they are I/O cells or logic cells. The actual relocation is done by treating smaller fixed blocks as movable objects and placing them with CPLACE as post-processing step. In the released benchmarks, there is no overlap among fixed blocks, and some of those fixed blocks are from area-array I/O cells. The number of artificial overlaps created from fixing I/O objects is however negligible and we confirm that, through experiments, this post-processing relocation never perturbs the original design much. In addition, no design contains movable I/O objects.

3.3 Rotation/Mirroring

In reality, some movable objects may be mirrored or rotated while others may not be. However, rotation and mirroring has to be performed carefully without violating any layout constraints. For example, arbitrary rotation/mirroring might cause ill-aligned VDD/GND metal stripes that should be avoided. To prevent any violation of such layout constraints, before the translation is performed, objects are rotated/mirrored in advance with CPLACE in order to make sure they satisfy layout constraints. In original designs, the feasibility of rotation/mirroring is included. But, this information has not been translated into final Bookshelf format for simplification. All directions of objects are defined as "N" (North) in input Bookshelf placement format PL file.

4. PLACEMENT LAYOUT FIGURES

Figure 1 presents figures of benchmarks placed with IBM CPLACE. To produce layout figures, the GSRC Bookshelf placement utility plotter [13] was used. These circuits include many fixed blocks demonstrating current SoC-style VLSI design methodology. Big fixed macro blocks basically dictate the overall placement footprints while placing movable dust logics around them becomes the main task of placement algorithm. It is still a challenging problem because placement algorithm must be able to cope with large number of objects (scalability), satisfy timing constraints (timing closure) and produce routable solutions (routability/congestion). Each design presents slightly different styles of placement problems. For example, *adaptec2* and *adaptec3* have large fixed blocks in the center of placement region, which may cause larger variations in wirelengths depending on which sides of fixed blocks movable cells are placed. In *bigblue1*, placing movable objects at the center region in more compact manner seems to be more critical task. The benchmarks *bigblue2* have relatively large number of pins from regularly placed small fixed blocks. The design density of *bigblue3* is over 85% primarily due to several large fixed blocks whereas design utilization is around 55% with abundant free space available. Two benchmarks *bigblue3* and *bigblue4*, with more than one million placeable objects, are good testcases for testing scalability of placement algorithms.

Circuit	#Total Objects	#Mov. Objects	#Fixed Objects	#Nets	#Total Pins	#Pins.Mov. Objs	#Pins.Fixed. Objs	Design Density	Design Util.	Peri. I/Os
adaptec1	211447	210904	543	221142	944053	923513	20540	75.71%	57.34%	480
adaptec2	255023	254457	566	266009	1069482	1045699	23783	78.56%	44.32%	407
adaptec3	451650	450927	723	466758	1875039	1843852	31187	74.53%	33.66%	0
adaptec4	496045	494716	1329	515951	1912420	1876563	35857	62.67%	27.23%	0
bigblue1	278164	277604	560	284479	1144691	1131856	12835	54.19%	44.67%	528
bigblue2	557866	534782	23084	577235	2122282	1979597	142685	61.80%	37.94%	0
bigblue3	1096812	1095519	1293	1123170	3833218	3790107	43111	85.65%	56.68%	0
bigblue4	2177353	2169183	8170	2229886	8900078	8710667	189411	65.30%	44.35%	0

Table 1. ISPD2005 circuit benchmark characteristics.

5. CONCLUSION

ISPD2005 placement contest and its new placement benchmark suite are described for the placement community. All benchmark circuits are directly derived from modern industrial ASIC designs and are expected to contribute to enhancing placement development and research in upcoming years. It is also our hope that the ISPD 2005 placement contest serves as a forum for better placement quality comparisons and provides key statistics on placement quality for the future placement research.

6. ACKNOWLEDGMENTS

Many colleagues at IBM and Adaptec have helped us to release these benchmarks. Thanks goes to Kirk Lamb, Fariba Kasemkhani, Greg Ruhl, Kenneth Klapproth, Mike Flannery, Daniel Moertl, Rudd Herring, Amy Purdy, Leon Stok, Lyle Hanrahan.

7. REFERENCES

[1] S. N. Adya, S. Chaturvedi, J. A. Roy, D. A. Papa and I. L. Markov, “Unification of Partitioning, Placement and Floorplanning,”, in Proc. *International Conference on Computer-Aided Design*, 2004, pp. 550-557.

[2] C. J. Alpert, “The ISPD98 Circuit Benchmark suite,” in Proc. *International Symposium on Physical Design*, 1998, pp. 80-85.

[3] C. J. Alpert, G.-J. Nam and P. Villarrubia, “Effective Free Space Management for Cut-based Placement,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 22(10), pp. 1343-1353, 2003.

[4] U. Brenner and A. Rohe, “An Effective Congestion Driven Placement Framework,” in Proc. *International Symposium on Physical Design*, 2002, pp. 6-11.

[5] F. Brglez, D. Bryan and K. Kozminski, “Combinational Profiles of Sequential Benchmark Circuits”, *International Conference on Circuits and Systems*, 1989, pp. 1929-1934.

[6] T. Chan, J. Cong and K. Sze, “Multilevel Generalized Force-directed Method for Circuit Placement,” in Proc. *International Symposium on Physical Design*, 2005.

[7] H. Eisenmann and F. M. Johannes, “Generic Global Placement and Floorplanning,” in Proc. *Design Automation Conference*, 1998, pp. 425-430.

[8] B. Hu and M. Marek-Sadowska, “FAR: Fixed Point Addition and Relaxation-based Placement,” in Proc. *International Symposium on Physical Design*, 2002, pp. 161-166.

[9] A. B. Kahng and Q. Wang, “Implementation and Extensibility of an Analytic Placer,” in Proc. *International Symposium on Physical Design*, 2004, pp. 18-25.

[10] A. Khatkhate, C. Li, A. R. Agnihotri, M. C. Yildiz, S. Ono, C.-K. Koh and P. H. Madden, “Recursive Bisection Based Mixed Block Placement,” in Proc. *International Symposium on Physical Design*, 2004, pp. 84-89.

[11] J. Lou, S. Krishnamoorthy and H. S. Sheng, “Estimating Routing Congestion using Probabilistic Analysis,” in Proc. *International Symposium on Physical Design*, 2001, pp. 112-117.

[12] P. H. Madden, “Reporting of Standard Cell Placement Results,” in Proc. *International Symposium on Physical Design*, 2001, pp. 30-35.

[13] A.E. Caldwell, A. B. Kahng, I. L. Markov, “Toward CAD-IP Reuse: The MARCO GSRC Bookshelf of Fundamental CAD Algorithms,” *IEEE Design and Test*, May 2002, pp. 72-81, http://vlsicad.eecs.umich.edu/BK/Slots/

[14] W.-J. Sun and C. Sechen, “Efficient and Effective Placement for Very Large Circuits,” *IEEE Trans. Computer-Aided Design*, vol. 14, pp. 349-359, Mar. 1995.

[15] N. Viswanathan and C. C.-N. Chu, “Fastplace: Efficient Analytical Placement Using Cell Shifting, Iterative Local Refinement and a Hybrid Net Model,” in Proc. *International Symposium on Physical Design*, 2004, pp. 26-33.

[16] J. Westra, C. Bartels and P. Groeneveld, “Probabilistic Congestion Prediction,” in Proc. *International Symposium on Physical Design*, 2004, pp. 204-208.

[17] X. Yang, B. K. Choi and M. Sarrafzadeh, “Routability-Driven White Space Allocation for Fixed-Die Standard-Cell Placement,” in Proc. *International Symposium on Physical Design*, 2002, pp. 42-47.

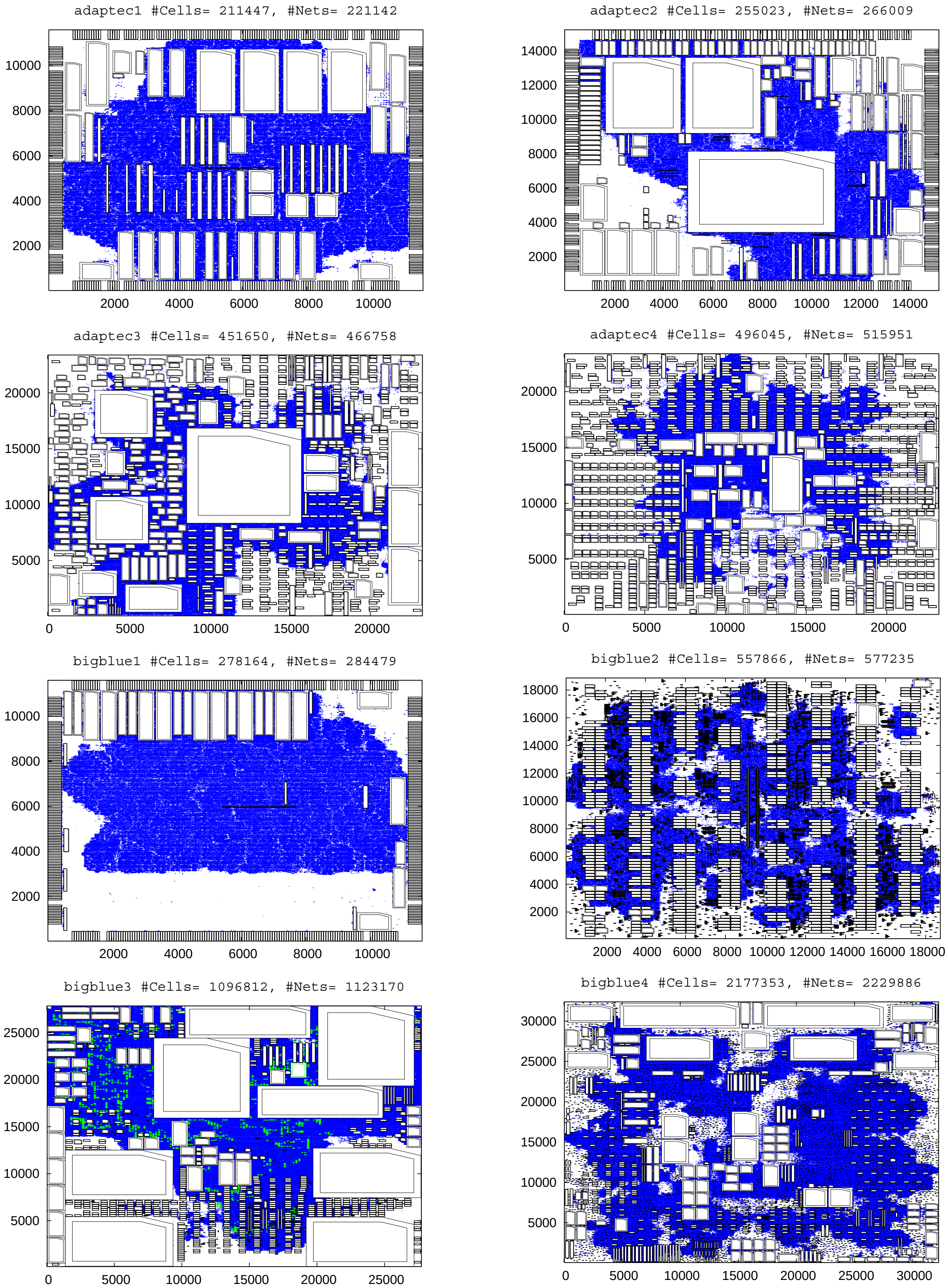

Figure 1. Layout figures of ISPD 2005 benchmark suite.