Partitioning Very Large Circuits Using Analytical Placement Techniques

Bernhard M. Riess, Konrad Doll, and Frank M. Johannes
Institute of Electronic Design Automation
Technical University of Munich, 80290 Munich, Germany

Abstract

A new partitioning approach for very large circuits is described. We demonstrate that applying a recently developed analytical placement algorithm, that profits from a linear objective function, significantly improves the partitioning quality compared to the well-known eigenvector approach, which minimizes a quadratic objective function. For the first time, results of benchmark circuits with up to 100,000 cells are presented. The cutsize and the minimum ratio cut is improved up to 90%. The average improvement is about 50%.

1 Introduction

Since high-level synthesis methods become widely accepted, the complexity of electronic systems is rapidly increasing. However, manufacturing technology limits the chip size. Consequently, the entire system has to be partitioned into a set of subsystems. These subsystems may be mapped on a set of ASICs or FPGAs and the system can be realized as a multichip module or an FPGA board. A partitioning algorithm has to divide the whole system into two or more subsystems by minimizing the cutsize and ensuring subsystem sizes within prescribed ranges.

Our intention in this paper is to present a partitioning method that offers the designer a variety of good partitioning solutions, where a small number of nets is cut and the size constraints are met.

Early approaches to the partitioning problem include clustering algorithms [1] and min-cut based methods [2,3]. Clustering is a bottom-up strategy, iteratively constructing strongly connected components. Starting from an initial partition, min-cut iteratively minimizes the cutsize by exchanging or moving cells or groups of cells. The partitioning quality obtained with min-cut methods largely depends on the initial partition, as only local improvements are performed. The success of these two approaches is limited by the lack of a global view. Furthermore, typically the cells are only moved or exchanged if the partition size limits are not violated. This restricts the possibilities of minimizing the cutsize. The ratio cut method [4,5] overcomes this drawback, but the partitioning result still depends on the initial partitioning.

To generate a good partitioning, the use of placement data was proposed [6,7,8]. Hagen and Kahng [9,10] combined a one-dimensional placement method [11] with the ratio cut measure. They calculate a one-dimensional placement by solving an eigenvector problem and determine the best partitioning by computing the ratio of all possible cuts between the cells. This algorithm keeps the global view and produces impressive partitioning quality. It reduces the number of possible two-way partitionings from 2^{n-1} to $n-1$, where n is the number of cells, as only $n-1$ possible cut positions remain between the cells in a one-dimensional placement. This decreases the computational complexity for partitioning from exponential to linear. As the one-dimensional placement method is based on eigenvector computations, which minimize a quadratic objective function, throughout this paper we call it *eigenvector approach*. Recent improvements of this method led to better results for small circuits [12,13,14,15].

Partitioning methods that use analytical placement techniques obtained good results. Therefore, it is promising to choose the best placement techniques to guide the partitioning. In particular, very good placement results have been reported using the analytical placement procedure Gordian [16,17]. It has been demonstrated that minimizing a linear objective function [17] yields a better placement quality in terms of layout area and wire length than minimizing a quadratic objective function. The superiority of a linear objective function in cell placement has also been shown by Hagen and Kahng [18]. Our idea is to use this objective function for partitioning. Reduced wire length leads to a lower probability that a net is cut. Since the linear objective function is superior to the quadratic objective function in terms of wire length, we claim that it will improve the partitioning quality as well. Our experiments show an improvement of up to 90% compared to the eigenvector approach.

Our paper is organized as follows. The next section contains some preliminaries and definitions. Section 3 presents a short review of the eigenvector approach. Our new partitioning approach, which uses a linear objective function, is described in Section 4. In Section 5, results of benchmark circuits with up to 100,000 cells are presented and discussed.

2 Preliminaries

2.1 Modeling the Circuit

A circuit is modeled by a hypergraph $H = (V, E')$ with vertices V representing the cells and hyperedges E' representing the nets. The hypergraph can be transformed to a graph $G = (V, E)$ by mapping each hyperedge in the set E' into a set of binary edges. To perform this mapping, we apply the well-known clique model. Each hyperedge consisting of k vertices is represented by a complete graph with edge weights equal to $1/(k-1)$. Thus, a graph G is obtained that may be described by an $n \times n$ adjacency matrix $\mathbf{A} = [a_{ij}]$, where $n = |V|$. The matrix elements a_{ij} are calculated as the sum of the edge weights of all edges connecting the vertices i and j.

The diagonal degree matrix $\mathbf{D} = [d_{ij}]$ is defined by:

$$d_{ij} = \begin{cases} \sum_{j=1}^{n} a_{ij} & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}$$

$\mathbf{D}$ is called degree matrix, because each d_{ii} is equal to the sum of all edge weights incident to vertex i.

Now we are able to compute the matrix

$$\mathbf{B} = \mathbf{D} - \mathbf{A} \tag{1}$$

which is used for the calculations below. $\mathbf{B}$ is usually called *disconnection matrix* [11] or the *Laplacian of G* [19]. The matrix is singular, has a maximum rank of $n-1$, is positive semi-definite, and has at least one zero eigenvalue, while all other eigenvalues are positive. The multiplicity of the zero eigenvalue is equal to the number of connected components of G.

2.2 The Ratio Cut Measure

We apply the ratio of a cut in the same way as Hagen and Kahng [9,10] to determine the best partitioning and to measure the obtained partitioning quality. According to Wei and Cheng [4,5] the ratio of a cut is defined by:

$$RC = \frac{C_{LR}}{|L| \cdot |R|} \tag{2}$$

The set of nodes V of $H = (V, E')$ is divided into two disjoint subsets $L \subset V$ and $R = V - L$ with $L \neq \emptyset$. The cutsize C_{LR} is the number of nets connecting the partitions L and R. The ratio cut favors both of our partitioning goals: Firstly, the numerator minimizes the cutsize and secondly, the denominator avoids uneven partition sizes.

3 The Eigenvector Approach

We present a short outline of the well-known eigenvector approach to be able to compare it to our new algorithm. As proposed by Hall [11] the one-dimensional placement problem may be formulated as a quadratic programming problem with a quadratic constraint

$$\text{QPPQC:} \quad \min_{\mathbf{x} \in \mathbb{R}^n} \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij}(x_i - x_j)^2 \quad \text{s.t.} \sum_{i=1}^{n} x_i^2 = 1,$$

where $\mathbf{x} = [x_1, \ldots, x_i, x_j, \ldots, x_n]^{\mathrm{T}} \in \mathbb{R}^n$ denotes the vector of the cell coordinates. The quadratic objective function to be minimized is the sum of the squared distances between the cells. The quadratic constraint distributes the cells around the origin with variance 1.

Using the matrix $\mathbf{B}$, QPPQC can be rewritten as

$$\min_{\mathbf{x} \in \mathbb{R}^n} \mathbf{x}^{\mathrm{T}}\mathbf{B}\mathbf{x} \qquad \text{s.t.} \quad \mathbf{x}^{\mathrm{T}}\mathbf{x} = 1.$$

To solve this problem, we form the Lagrangian

$$\ell(\mathbf{x}, \lambda) = \mathbf{x}^{\mathrm{T}}\mathbf{B}\mathbf{x} - \lambda(\mathbf{x}^{\mathrm{T}}\mathbf{x} - 1) \tag{3}$$

with the Lagrange multiplier λ. Setting the first partial derivative of $\ell(\mathbf{x}, \lambda)$ with respect to $\mathbf{x}$ to zero leads to the eigenvalue equation

$$(\mathbf{B} - \lambda\mathbf{I})\mathbf{x} = \mathbf{0}, \tag{4}$$

where $\mathbf{I}$ is the identity matrix.

This equation holds for any vector $\mathbf{x}$ and Lagrange multiplier λ if and only if $\mathbf{x}$ is eigenvector and λ the corresponding eigenvalue of $\mathbf{B}$. Multiplying Equation 4 with $\mathbf{x}^{\mathrm{T}}$ leads to

$$\lambda = \mathbf{x}^{\mathrm{T}}\mathbf{B}\mathbf{x}. \tag{5}$$

This shows that the value of the objective function is equal to λ for any eigenvector $\mathbf{x}$. The eigenvector corresponding to the eigenvalue $\lambda = 0$ is no practical solution as all cell coordinates are equal to $1/\sqrt{n}$. Thus, the smallest nonzero eigenvalue and the associated eigenvector yield the best useful solution of QPPQC. The components of this eigenvector are interpreted as cell coordinates yielding a one-dimensional placement. For a small example, such a placement is shown on top of Figure 1. Nets connecting the cells are not drawn for reasons of simplicity.

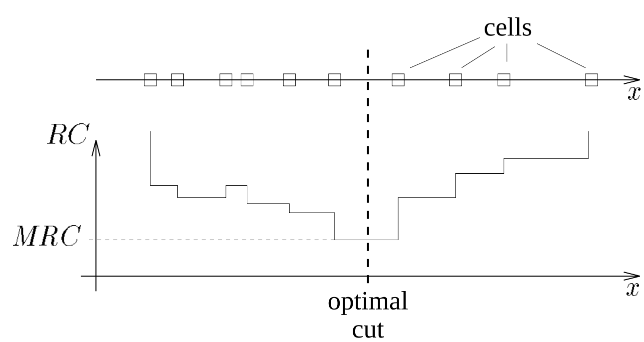

Figure 1: Eigenvector placement and ratio cut diagram

To determine the best partitioning and to assess the quality of the entire approach the ratio of a cut RC is calculated for every possible cut position between two cells according to Equation 2. Thus, we obtain a ratio cut diagram as shown in Figure 1. The set of cells is partitioned at the optimal cut position yielding the minimum ratio cut (MRC). This method was first presented by Hagen and Kahng [9,10], where good partitioning results were reported.

4 Partitioning Using a Linear Objective Function

Our intention is to create a partitioning algorithm obtaining a good partitioning quality by an improved one-dimensional placement. In addition, it must have the ability to deal even with the largest circuits available. The largest design we partitioned up to now had about 100,000 cells.

To solve the placement problem, GORDIAN [16] used the same quadratic objective function which is used in QPPQC, but a linear constraint:

$$\text{QPPLC:}\quad \min_{\mathbf{x}\in\mathbb{R}^n} \sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij}(x_i - x_j)^2 \quad \text{s.t.} \sum_{i=1}^{n} x_i = f$$

In contrary to the quadratic constraint used in QPPQC which distributes the cells, the linear constraint fixes the center of gravity of all cells to the x-coordinate f. This problem formulation yields good placement results which can be improved significantly by using the following problem formulation with a linear objective function:

$$\text{LPPLC:}\quad \min_{\mathbf{x}\in\mathbb{R}^n} \sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij}\,|x_i - x_j| \quad \text{s.t.} \sum_{i=1}^{n} x_i = f$$

This linear programming problem can be rewritten as a quadratic programming problem

$$\min_{\mathbf{x}\in\mathbb{R}^n} \sum_{i=1}^{n}\sum_{j=1}^{n} g_{ij}(x_i - x_j)^2 \qquad \text{s.t.} \sum_{i=1}^{n} x_i = f$$

by weighting the nets with $g_{ij} = \dfrac{a_{ij}}{|x_i - x_j|}$. It is solved in GORDIANL very efficiently by using a conjugate-gradient method [17].

Since the quality of placement-based partitioning approaches largely depends on the quality of the one-dimensional placement, we use GORDIANL with it's linear objective function to generate a high quality one-dimensional placement. We apply the GORDIANL procedure as described in [17] except that only a one-dimensional placement is calculated. As GORDIANL obtains good results in cell placement with the linear constraint we use it for partitioning, too. A better placement quality results in shorter wire length. Reducing the wire length decreases the probability for a net to be cut when partitioning the circuit. Therefore, it is reasonable to assume that a better placement quality in terms of wire length implies a lower cutsize.

Since GORDIANL needs at least one fixed cell, we use the eigenvector placement to assign extreme left and extreme right placed cells to the left and right partition, respectively. We fix the coordinates of these cells and calculate a new placement for all remaining cells with GORDIANL using it's linear objective function and constraint. An improved placement result is shown on top of Figure 2. Fixed cells are shown as filled squares.

The optimal partitioning is determined in the same way as presented in Section 3. Again the ratio for every possible cut position is calculated and the circuit is partitioned at the optimal cut.

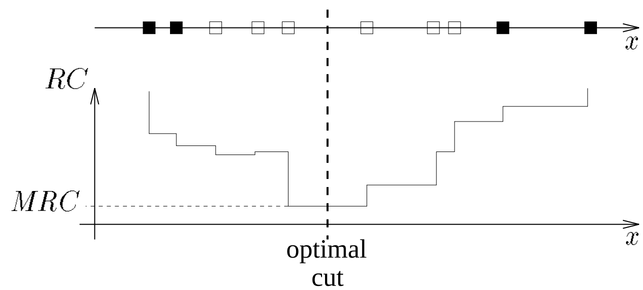

Figure 2: Improved placement and ratio cut diagram applying the linear objective function

5 Experimental Results

The results of our new partitioning method called PARABOLI are compared to the eigenvector approach in terms of the minimum ratio cut (MRC) and the cutsize (C_{LR}). For that purpose, we implemented the EIG1 algorithm of Hagen and Kahng [9,10]. We applied both methods to 19 circuits of the ACM/SIGDA Benchmark Suites [20,21]. The characteristics of the circuits containing approximately 600 to 100,000 cells are summarized in Table 1. Partitioning results for the larger circuits have never been published before.

circuit	#cells	#nets	#pins
s1423	619	538	1 528
sioo	664	408	1 882
s1488	686	667	2 079
balu	801	735	2 697
primary1	833	904	2 941
struct	1 952	1 920	5 471
primary2	3 014	3 029	11 226
s9234	5 866	5 844	14 065
biomed	6 514	5 742	21 040
s13207	8 772	8 651	20 606
s15850	10 470	10 383	24 712
industry2	12 637	13 419	48 404
industry3	15 406	21 924	68 290
s35932	18 148	17 828	48 145
s38584	20 995	20 717	55 203
avq.small	21 918	22 124	76 231
s38417	23 949	23 843	57 613
avq.large	25 178	25 384	82 751
golem	100 312	144 949	338 622

Table 1: Characteristics of benchmark examples

circuit	EIG1			PARABOLI				Improvement	
	C_{LR}	*MRC* [10^{-7}]	cpu [s]	C_{LR}	*MRC* [10^{-7}]	add'l cpu [s]	total cpu [s]	C_{LR}	*MRC*
s1423	23	2 416.2	1.7	6	1 453.9	6.0	7.7	73.9%	39.8%
sioo	30	7 601.1	7.1	34	3 763.4	8.7	15.8	-13.3%	50.5%
s1488	144	13 040.6	7.0	39	4 024.2	10.0	17.0	72.9%	69.1%
balu	85	5 301.6	6.2	32	2 032.6	9.3	15.5	62.4%	61.7%
primary1	15	1 464.1	3.1	14	1 338.9	15.2	18.3	6.7%	8.5%
struct	59	636.9	6.9	40	420.2	28.3	35.2	32.2%	34.0%
primary2	77	457.9	17.6	77	457.9	119.8	137.4	0.0%	0.0%
s9234	9	23.1	24.2	9	23.1	466.1	490.3	0.0%	0.0%
biomed	35	86.1	521.2	42	61.9	189.7	710.9	-20.0%	28.0%
s13207	39	42.2	43.5	10	10.2	2 016.9	2 060.4	74.4%	75.7%
s15850	32	21.6	78.4	7	6.9	2 652.5	2 730.9	78.1%	68.0%
industry2	280	143.5	706.6	106	30.8	660.7	1 367.3	62.1%	78.5%
industry3	136	24.4	195.4	113	20.0	565.3	760.7	16.9%	16.7%
s35932	105	12.8	2 066.6	47	5.8	560.1	2 626.7	55.2%	54.8%
s38584	76	7.0	347.5	55	5.0	6 170.0	6 517.5	27.6%	27.9%
avq.small	241	31.9	3 139.9	27	4.8	959.0	4 098.9	88.8%	84.9%
s38417	121	8.5	281.3	49	3.4	1 760.2	2 041.5	59.5%	59.5%
avq.large	253	25.6	2 995.8	27	3.6	1 139.2	4 135.0	89.3%	86.0%
golem	1 768	19.5	1 893.3	1 581	6.4	8 929.2	10 822.5	10.6%	67.4%
Average								40.9%	47.9%

Table 2: Partitioning results for the optimal cut

5.1 Arbitrary Partition Sizes

In Table 2 the partitioning results of PARABOLI and EIG1 are compared. To avoid extreme uneven partition sizes, the extreme left and extreme right 10% of all possible cut positions are not considered. Columns 2 to 4 show the cutsize (C_{LR}), the minimum ratio cut (*MRC*) and the cpu time to calculate the eigenvector in the EIG1 algorithm. All computations were executed on a DEC 3000 Model 500 AXP. The results of PARABOLI are summarized in columns 5 to 7. The cpu-time given in column 7 is needed to calculate a one-dimensional placement with GORDIANL. Column 8 gives the total cpu-time of our approach, which is the sum of the cpu-time for the eigenvector calculation and the improved placement calculated with GORDIANL. Finally, the improvement of PARABOLI compared to EIG1 is presented with respect to the cutsize and the minimum ratio cut.

In order to reduce computational effort, some eigenvector approaches remove nets with a large number of pins [14]. However, our investigations revealed that partitioning results are very sensitive to net elimination. In some cases net elimination improves the partitioning quality, while in other cases the results are worse. Therefore, to present comparable results, we dispense with net elimination as far as possible when calculating the eigenvector of **B**. In our experiments the eigenvectors of the circuits *avq.small* and *avq.large* only are calculated without the 4 biggest nets connecting more than 3,000 pins due to excessive memory requirements.

When applying the GORDIANL placement tool we observed, that the placement results could be improved by eliminating nets with a large number of pins. This improvement has been consistent for all investigated circuits. For this reason, PARABOLI neglects nets with more than 60 pins. This results in relatively small computation times for designs with a large number of nets with more than 60 pins compared to the EIG1 approach. In cases where cpu(EIG1) < cpu(PARABOLI) no or only a few nets were removed by PARABOLI. In cases where cpu(EIG1) > cpu(PARABOLI) PARABOLI eliminated several large nets.

PARABOLI yields up to 86% lower minimum ratio cuts. On the average 48% better results are obtained. The minimum ratio cut of PARABOLI is always better or equal than the minimum ratio cut of the EIG1 algorithm. The cutsize is reduced up to 89% with an average of about 41%. The improvement on the cutsize is sometimes less significant or even negative compared to the improvement of the *MRC*. For the circuits *sioo* and *biomed* PARABOLI creates a higher cutsize, but more even partition sizes outweigh this deterioration such that the *MRC* is improved. This means, that PARABOLI generates more even partition sizes (see Equation 2).

Some of our partitioning results break the lower ratio cut bound $c \geq \lambda/n$ of a minimum ratio cut partition [9,10]. The proof of this bound [10] is based on a clique model with edge weights equal to one which results in counting all edges of the cliques that are cut. However, we propose that a net connecting cells in both partitions causes cut costs equal to one. Therefore, the lower ratio cut bound c may be higher than the actual cost for a net connecting cells in both partitions.

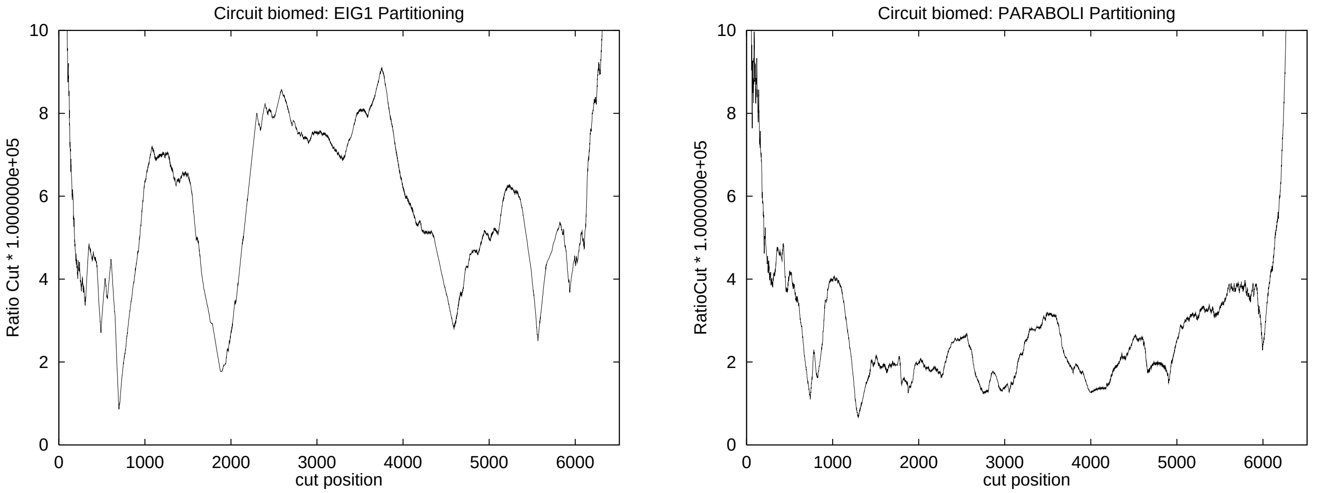

Figure 3: Ratio cut diagram: Ratio cut versus cut position for the circuit *biomed*

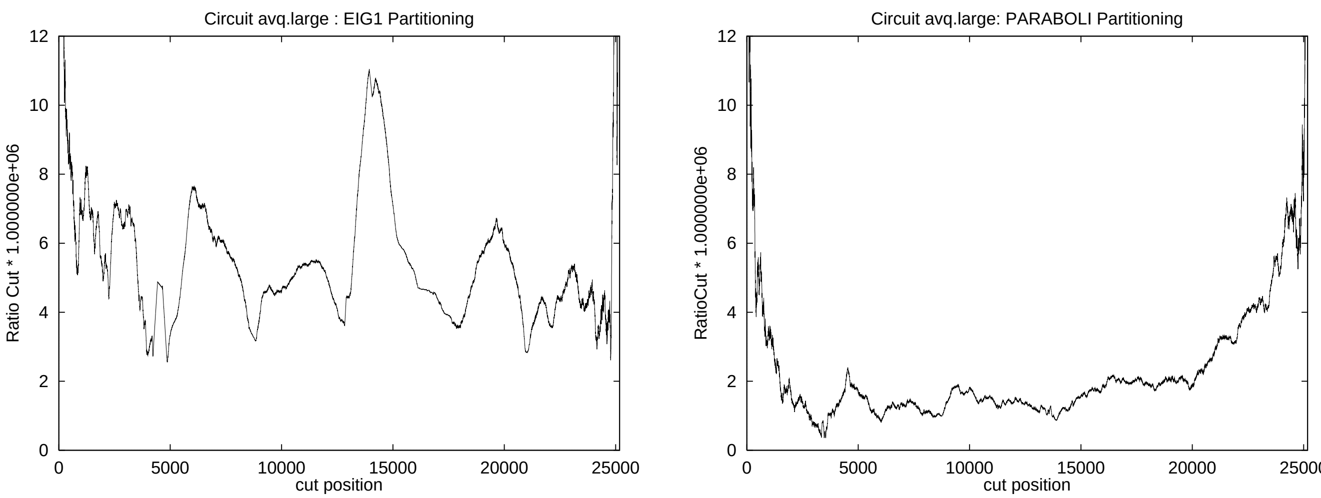

Figure 4: Ratio cut diagram: Ratio cut versus cut position for the circuit *avq.large*

5.2 A Reduced Ratio Cut Level

A closer look at the ratio cut diagrams reveals that not only the minimum ratio cut should be considered to assess the quality of our approach, but also the entire ratio cut diagram should be taken into consideration. Figures 3 and 4 give the ratio cut diagram for the circuits *biomed* and *avq.large* for EIG1 and PARABOLI. PARABOLI reduces the ratio cut level over a wide range of cut positions. Compared to the eigenvector approach with only a few minima, PARABOLI gives a great variety of good cut positions, offering the designer a lot of partitioning solutions with a small ratio cut. This advantage of our approach is essential for multiway partitioning. It clearly shows the significant improvement compared to the eigenvector approach.

5.3 Specified Partition Sizes

System design often requires approximately even partitions. Thus, we compared both approaches allowing each partition to have up to 10% more or less than the equipartitioned number of cells. Table 3 presents the minimum ratio cut and the cutsize for the optimal cut in this interval. PARABOLI yields an improvement between 25% and 80% in terms of the cutsize and the minimum ratio cut compared to the eigenvector approach. On the average, about 55% improvement is obtained.

The results in Tables 2 and 3 show an average improvement of 49.7%. This significant improvement outweighs the on the average moderate additional computational effort. It is important to notice that these excellent results are produced without performing any local improvement.

In our prototype implementation we first calculate the eigenvector, assign extremely placed cells to distinct partitions, and finally compute an improved placement with the linear objective function. It may be a part of future research to unify both approaches by using net weights during the eigenvector calculation to linearize the objective function. This will result in a lower computational cost.

circuit	EIG1		PARABOLI		Improvem.	
	C_{LR}	MRC $[10^{-7}]$	C_{LR}	MRC $[10^{-7}]$	C_{LR} [%]	MRC [%]
s1423	23	2 416.2	16	1 670.4	30.4	30.9
sioo	128	11 619.0	45	4 123.3	64.8	64.5
s1488	158	13 485.3	50	4 259.1	68.4	68.4
balu	85	5 301.6	41	2 579.9	51.8	51.3
primary1	81	4 671.7	53	3 057.6	34.6	34.6
struct	102	1 081.7	40	420.2	60.8	61.2
primary2	197	876.0	146	646.6	25.9	26.2
s9234	227	265.6	74	86.0	67.4	67.6
biomed	729	687.3	135	127.8	81.5	81.4
s13207	241	125.6	91	47.6	62.2	62.1
s15850	215	78.7	91	33.2	57.6	57.8
industry2	620	155.4	193	48.7	68.9	68.6
industry3	399	67.4	267	45.3	33.1	32.7
s35932	105	12.8	62	7.6	41.0	40.5
s38584	76	7.0	55	5.0	27.6	28.0
avq.small	598	49.8	224	18.7	62.5	62.5
s38417	121	8.5	49	3.4	59.5	59.5
avq.large	571	36.0	139	8.9	75.6	75.4
golem	5 379	21.5	1 629	6.6	69.7	69.5
Average					54.9	54.9

Table 3: Partitioning results allowing up to 10% deviation from bisection

6 Conclusions

We developed an efficient partitioning method for two- and multiway partitioning. The partitioning is based on a cell placement which is calculated efficiently by applying analytical placement techniques. The main conclusions of our research are:

- A linear objective function is superior to the quadratic objective function for partitioning as well as for placement.
- Our approach dramatically improves both, the cut-size and the minimum ratio cut.
- The ratio cut level is reduced over a wide range of cut positions offering the designer a high degree of freedom to choose an appropriate partitioning for specified partition sizes.
- Excellent results are shown even on the largest available Benchmark circuits with up to 100,000 cells.

Acknowledgment

The authors would like to thank Prof. K. Antreich for his continuous advice and support and Prof. J. Cohoon, Univ. of Virginia, for his helpful suggestions during his sabbatical in Munich, Germany. This work was supported in part by the German National Science Foundation (DFG) under Grant An 125/11-1.

References

[1] D. M. Schuler and E. G. Ulrich, "Clustering and Linear Placement," *Proceedings 9th Design Automation Workshop*, pp. 50–56, 1972.

[2] B. W. Kernighan and S. Lin, "An Efficient Heuristic Procedure for Partitioning Graphs," *Bell Systems Technical Journal*, pp. 291–307, 1970.

[3] C. M. Fiduccia and R. M. Mattheyses, "A Linear-Time Heuristic for Improving Network Partitions," *Proc. 19th ACM/IEEE Design Automation Conf.*, pp. 175–181, 1982.

[4] Y.-C. Wei and C.-K. Cheng, "Towards Efficient Hierarchical Designs by Ratio Cut Partitioning," *IEEE/ACM Int. Conf. on Computer-Aided Design*, pp. 298–301, 1989.

[5] Y.-C. Wei and C.-K. Cheng, "Ratio Cut Partitioning for Hierarchical Designs," *IEEE Transactions on Computer-Aided Design*, vol. 10, no. 7, pp. 911–921, 1991.

[6] E. R. Barnes, "An Algorithm for Partitioning the Nodes of a Graph," *SIAM J. ALG. DISC. METH.*, vol. 3, no. 4, pp. 541–550, 1982.

[7] J. Frankle and R. M. Karp, "Circuit Placements and Cost Bounds by Eigenvector Decomposition," *IEEE/ACM Int. Conf. on Computer-Aided Design*, pp. 414–417, 1986.

[8] R. B. Boppana, "Eigenvalues and Graph Bisecion: An Average-case Analysis," *IEEE Symp. on Foundations of Computer Science*, pp. 280–285, 1987.

[9] L. Hagen and A. B. Kahng, "Fast Spectral Methods for Ratio Cut Partitioning and Clustering," *IEEE/ACM Int. Conf. on Computer-Aided Design*, pp. 10–13, 1991.

[10] L. Hagen and A. B. Kahng, "Fast Spectral Methods for Ratio Cut Partitioning and Clustering," *CS Departement Tech. Report, University of California, Los Angeles*, 1991.

[11] K. M. Hall, "An r-Dimensional Quadratic Placement Algorithm," *Management Science*, pp. 219–229, 1970.

[12] J. Cong, L. Hagen, and A. Kahng, "Net Partitions Yield Better Module Partitions," *Proceedings 29th ACM/IEEE Design Automation Conference*, pp. 47–52, 1992.

[13] C. J. Alpert and A. B. Kahng, "Geometric Embeddings for Faster and Better Multi-Way Netlist Partitioning," *Proc. 30th ACM/IEEE DAC*, pp. 743–748, 1993.

[14] S. W. Hadley, B. L. Mark, and A. Vanelli, "An Efficient Eigenvector Approach for Finding Netlist Partitions," *IEEE Trans. on CAD*, pp. 885–892, 1992.

[15] S. Areibi and A. Vanelli, "A Combined Eigenvector Tabu Search Approach for Circuit Partitioning," *IEEE Proc. Custom Integrated Circuits Conf.*, pp. 9.7.1 – 9.7.4, 1993.

[16] J. M. Kleinhans, G. Sigl, F. M. Johannes, and K. J. Antreich, "GORDIAN: VLSI Placement by Quadratic Programming and Slicing Optimization," *IEEE Trans. on Computer-Aided Design*, vol. 10, no. 3, pp. 356–365, 1991.

[17] G. Sigl, K. Doll, and F. M. Johannes, "Analytical Placement: A Linear or a Quadratic Objective Function?," *Proc. 28th ACM/IEEE DAC*, pp. 427–432, 1991.

[18] L. Hagen and A. B. Kahng, "Improving the Quadratic Objective Function in Module Placement," *Fifth Annual IEEE Int. ASIC Conference and Exhibit*, pp. 42–45, 1992.

[19] P. K. Chan, D. F. Schlag, and J. Y. Zien, "Spectral K-Way Ratio Cut Partitioning and Clustering," *Proc. 30th ACM/IEEE Design Automation Conf.*, pp. 749–754, 1993.

[20] K. Kozminski, "Benchmarks for Layout Synthesis - Evolution and Current Status," *Proceedings 28th ACM/IEEE Design Automation Conference*, pp. 265–270, 1991.

[21] F. Brglez, D. Bryan, and K. Kozminski, "Combinational Profiles of Sequential Benchmark Circuits," *Proc. IEEE Int. Symp. on CAS*, vol. 3, no. 3, pp. 1929–1934, 1989.