New Spectral Linear Placement and Clustering Approach *

Jianmin Li, John Lillis, Lung-Tien Liu, and Chung-Kuan Cheng
Dept. of Computer Sci. & Engr.
University of California, San Diego
La Jolla, CA 92093-0114

Abstract

This paper addresses the linear placement problem by using a spectral approach. It has been demonstrated that, by giving a more accurate representation of the linear placement problem, a linear objective function yields better placement quality in terms of wire length than a quadratic objective function as in the eigenvector approach [4][11][6]. On the other hand, the quadratic objective function has an advantage in that it tends to place components more sparsely than the linear objective function, resulting in a continuous solution closer to a physically feasible discrete solution. In this paper, we propose an α-order objective function to capture the strengths of both the linear and quadratic objective functions. We demonstrate that our approach yields improved spectral placements. We also present a bottom-up clustering algorithm which iteratively collapses pairs of nodes in a graph using local and global connectivity information, where the global connectivity information is derived from the clustering property of the eigenvector approach. The effect of our new spectral linear placement and clustering approach is demonstrated on benchmark circuits from MCNC.

1 Introduction

The linear placement problem is a fundamental problem in the field of VLSI design. It can be applied to the 2D placement problem, which is crucial for circuit layout quality. Another important application of linear placement is circuit partitioning.

As system complexity dramaticly increases, the divide-and-conquer concept is usually applied in circuit design, resulting in circuit hierarchy and greater demand for good partitioning methods. Recently, several linear placement based approaches have used the eigenvector method to yield very impressive partitioning quality[5][6][4]. These approaches all assume that a better linear placement result in shorter wire length and therefore reduce the probability for a net to be cut[6][4][11]. This motivates us to further exploit the eigenvector approach for better linear placement in terms of wire length. Our approach introduces a new spectral objective function, which is a tradeoff between linear and quadratic function, and exploits the clustering property of the eigenvector approach.

In [11][4], comparisons between linear and quadratic objective functions were made. It was found that much better placement quality, in terms of area or wire length, is obtained by using a linear function. In [6][4], the use of a linear function for placement also yielded great improvements in circuit partitioning in terms of cut capacity. However, the quadratic objective function appears to have merit in that it tends to result in fewer very long nets than the linear objective function. In other words, the standard deviation of the net lengths is smaller for the quadratic function than for the linear function[11]. This means that the quadratic function tends to place components more sparsely, resulting in fewer components overlapping each other. This advantage must be balanced with that fact that the quadratic function minimizes the squared wire length rather than the linear wire length, and thus does not correspond directly to the goal of the linear placement. In this paper, we present a new spectral objective function, which is a compromise between the linear function and quadratic function; therefore, our objective function can take advantages of both the more sparse placement provided by the quadratic function and more accurate measurement provided by the linear function.

A cluster is a group of "strongly connected" components in a circuit. In placement and partitioning problems, the components of a cluster should be placed closely or in the same subcircuit; therefore, a good clustering method not only can significantly reduces the problem size but also can improve the solution quality of partitioning and placement heuristics [13][12], especially for huge circuits. Furthermore, a clustering approach also has an advantage in the multi-pin net modeling. When a circuit has been clustered, the degree of the hyperedges in the graph is reduced. As a result, the graph models used to approximate the hypergraph (e.g., the star or clique models) in the spectral method become more accurate (and exact in the case where the clustered circuit contains only 2-pin nets).

We observe that the well-known eigenvector approach not only provides a heuristic solution to the linear placement problem but also provides global connectivity information for clustering. For example, if there is a circuit which consists of disconnected subcircuits, the nodes of each subcircuits will merge into a single point in the optimal eigenvector solution. In this extreme case, we can derive the clustering information for each merged set of nodes; but the linear ordering of each merged set of nodes will be arbitrary and derive a suboptimal solution. In this paper, we exploit the clustering property of the eigenvector approach, and use it as a heuristic

*This work was supported in part by Grants from the NSF project MIP-9315794 and California MICRO program.

providing global connectivity information.

Our spectral linear placement and clustering approach can be described in three steps. First, compress the circuit by using a bottom-up clustering approach; second, linear placement is done on the clustered circuit by using spectral approach; third, we uncompress these clusters and liner placement is done on the original circuit. Such an approach has the following features:

- By using bottom-up clustering, the problem size is reduced and the eigenvector approach is applied to a much smaller circuit, resulting in less running time.
- With the bottom-up approach, the clusters are iteratively merged, resulting in more accurate net modeling because the net size is reduced during the clustering process.
- By compressing the original circuit, we mainly focus on global linear placement; and by uncompressing the clusters, we focus on local placement with global placement unchanged. This is actually a hierarchical approach, especially suitable for huge circuits.

Experiments on MCNC benchmarks show that our new spectral linear placement and clustering approach yields an improvement of up to 22.5% for linear placement in terms of total wire length, compared to the PANZA approach[4].

The rest of this paper is organized as follows. In the next section, we briefly review the eigenvector approach, and propose our new spectral objective function. In section 3, we exploit the clustering property of the eigenvector approach. Section 4 presents our spectral linear placement and clustering approach. In section 5, we provide the experimental results on the linear placement with MCNC benchmarks. Finally we conclude this paper in section 6.

2 Spectral Linear Placement

2.1 Preliminaries

A circuit is modeled by a hypergraph $G_H = (V_H, E_H)$, where the vertex set V_H represents the components and the hyperedge set E_H represents the nets connecting the components. The linear placement problem of a circuit is then to put these components in equally spaced slots(one component to one slot) such that the total wire length over all nets is minimal. The wire length of a net is defined by the span of its component set. In general, this hypergraph is approximated by a graph $G = (V, E)$, where a hyperedge is represented by a set of edges. In such an approximation, clique and star are the two most common models used for VLSI design problems.

Let $n = |V|$ be the number of vertices and $m = |E|$ be the number of edges in G. Then, this graph G can be described by a $n \times n$ adjacency matrix $A = [a_{ij}]$, where the matrix element a_{ij} is the weight of the connection between vertex i and vertex j.

After transforming a hypergraph into a graph, we can formulate the linear placement problem as below:

$$min \sum_{i>j} \sum_{j} a_{ij} |d_i - d_j| \tag{1}$$

where d_i is the coordinate of the slot for vertex i in the linear placement. The linear placement problem is known to be NP-complete. With a spectral approach, a continuous linear placement, where the restriction on placing vertices at specific slots is released, is usually used as the heuristic to solve the linear placement problem.

2.2 Eigenvector Approach to Linear Placement

Given a weighted graph $G = (V, E)$, represented by the $n \times n$ adjacency matrix $A = [a_{ij}]$, the Laplacian of G is defined as the matrix B where

$$B = \begin{cases} \sum_{j=1}^{j=n} a_{ij} & \text{if } i = j \\ -a_{ij} & \text{otherwise.} \end{cases} \tag{2}$$

With the eigenvector approach, the continuous linear placement problem is formulated as a quadratic programming problem, as shown below:

$$\min \sum_{i>j} \sum_{j} a_{ij}(x_i - x_j)^2 = X^T B X \tag{3}$$

$$\text{s.t. } I^T X = 0, X^T X = 1$$

where x_i is the coordinate of vertex i in the continuous linear placement.

By computing the nonzero smallest eigenvalue and its corresponding eigenvector X of the Laplacian B, we obtain a non-trivial solution to above quadratic programming problem, and the heuristic solution to the linear placement problem is obtained by interpreting the eigenvector as a linear ordering on the vertices V. With such an approach, as we can see, the objective function to be minimized is the squared wire length.

In [11][6][4], the linear objective function, formulated as below:

$$min \sum_{i>j} \sum_{j} a_{ij} |x_i - x_j| \tag{4}$$

$$\text{s.t. } \sum_{i} x_i = f$$

was used to solve the continuous linear placement. Very impressive quality on placement and partitioning was obtained. A better heuristic, the continuous linear placement with the linear objective function, results in the improvement of the quality of the linear placement.

Such a linear objective function can also be rewritten as a quadratic function by modifying the a_{ij} with the distance $|x_i - x_j|$ [11][6][4]:

$$\sum_{i>j} \sum_{j} a_{ij} |x_i - x_j| = \sum_{i>j} \sum_{j} a'_{ij} (x_i - x_j)^2$$

where $a'_{ij} = \frac{a_{ij}}{|x_i - x_j|}$. Therefore, through iterative improvement approach, the quadratic programming method can still be used to solve the continuous linear placement with the linear objective function.

2.3 α-order Objective Function

On the linear placement problem, the linear objective function receives much better quality than the quadratic objective function; that is mainly because the linear function is more accurate measurement for the linear placement problem than the quadratic function. On the other hand, the quadratic function still has its advantages over the linear function. The quadratic function tends to make very long nets shorter than the linear function does, or the standard deviation of the net lengths is smaller for the quadratic function than for the

linear function[11]. This means the quadratic function tends to place vertices more sparsely, resulting in less vertices overlain each other.

Because the linear placement problem is heuristically solved by interpreting the eigenvector as the ordering of vertices, the more sparsely the vertices are placed, the less numerical errors are introduced on the linear placement; therefore, the continuous solution of the linear placement should be sparse enough to be interpreted while the objective function is as accurate as possible.

Based on these observations, we propose the α-order objective function for the continuous linear placement problem, as shown below:

$$min \sum_{i>j}\sum_{j} a_{ij}|x_i - x_j|^{\alpha} = \sum_{i>j}\sum_{j} \frac{a_{ij}}{|x_i - x_j|^{2-\alpha}}(x_i - x_j)^2 \quad (5)$$

$$\text{s.t.} \sum_{i} x_i = f$$

where $1.0 \leq \alpha \leq 2.0$. When $\alpha = 1.0$, the α-order function becomes the linear function; and the α-order function becomes the quadratic function when $\alpha = 2.0$. With the α-order objective function, we hope to increase the sparsity of the solution to the continuous linear placement, while we still maintain accurate enough measurement for the linear placement. Similar to the linearized eigenvector approach in [4], this continuous linear placement with the α-order objective function can be solved iteratively; and the following theorem shows that such an iterative approach is convergent.

Theorem 1 *Using the iterative approach for linearized eigenvector in [4], the wire length, as formulated in (5), is monotonically decreasing.*

Because it is very hard and seems to be impossible to derive theoretical statements about the effects of various α on the solution to the linear placement problem; therefore, we here show the effects by experiments on MCNC benchmarks. We did experiments on MCNC benchmarks with $\alpha = 2.0$(quadratic), $\alpha = 1.5$, $\alpha = 1.2$, and $\alpha = 1.0$(linear) in the α-order objective function. The results are shown in Table 1; and Figure 1 shows the histogram of the vertex distribution of the continuous linear placement with various α in the objective function for the benchmark S38417. In Figure 1, the X axis represents the vertex coordinates of the continuous linear placement; in order to easily draw these histograms, we scale the vertex coordinates up by a factor of 20. The vertical axis represents the percentage of vertices. Note that the norm of these cases are the same. However, we use the star model for net modeling, where for each net we introduce a dummy node, and Figure 1 shows only the real nodes in the circuit. It is easy to see that the nodes are placed more densely in the continuous linear placement when α decreases.

From the experimental results, it is easy to see that α-order objective function with $\alpha = 1.2$ has obtained the best quality on the linear placement in terms of total wire length among these four cases. That is because the α-order($\alpha = 1.2$) objective function has higher sparsity on the continuous linear placement than the linear function does, and it also has more accurate measurement on the linear placement than the quadratic function does, resulting in better performance.

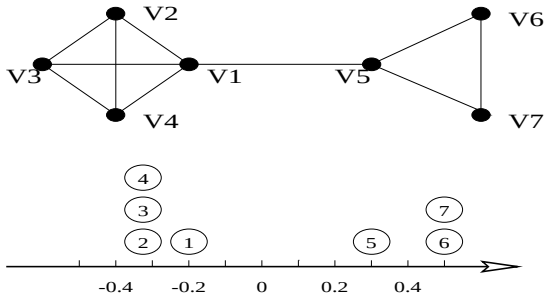

Figure 2: *A Simple Example*

3 Exploiting the Clustering Property

The solution to the continuous linear placement problem by using the eigenvector approach is not only a good heuristic for the linear placement problem but also a good heuristic for clustering problem, where the continuous solution provides the global connectivity information. In this section, we will demonstrate the clustering property of the eigenvector approach.

Figure 2 shows a very simple graph and the continuous linear placement of the eigenvector approach. This simple example intuitively demonstrates the clustering property of the eigenvector approach. In this example, we have seven vertices $V = \{v_i | i = 1, \ldots, 7\}$ and 10 nets, as shown in Figure 2. Then the eigenvector corresponding to the nonzero smallest eigenvalue $\lambda_1 = 0.398321$ for this example is $X = \{x_1 = -0.21422, x_2 = x_3 = x_4 = -0.356037, x_5 = 0.29656, x_6 = x_7 = 0.492886\}$.

From this continuous linear placement, we can roughly divide the vertices V into two groups, one group includes the vertices $(v_i | i = 1, \ldots, 4)$ and another group includes the vertices $(v_i | i = 5, 6, 7)$; because the vertices in each group are overlain each other or placed very closely while the gap between these two groups is quite large. This means that the intra-group connectivity is much higher than the inter-group connectivity, or the probability that some vertices in a group form a cluster is much higher than the probability that some vertices from different groups form a cluster. Therefore, it is quite natural to use the continuous linear placement result as the global connectivity information in a clustering process.

Theorem 2 *Given the Laplacians B_1 and B_2 of the graphs G_1 and G_2, let $\Lambda_1 = \{0, \lambda'_1, \ldots, \lambda'_{k-1}\}$ be the eigenvalues of B_1 and $\Lambda_2 = \{0, \lambda''_1, \ldots, \lambda''_{l-1}\}$ be the eigenvalues of B_2. Then the eigenvalues of B:*

$$B = \begin{pmatrix} B_1 & 0 \\ 0 & B_2 \end{pmatrix}$$

are $\Lambda_1 \cup \Lambda_2 = \{0, 0, \lambda'_1, \ldots, \lambda'_{k-1}, \lambda''_1, \ldots, \lambda''_{l-1}\}$.

Theorem 3 *Given two graphs G_1 and G_2, and their Laplacians B_1 and B_2, then the Laplacian of the graph G=$G_1 \cup G_2$ has two eigenvectors with eigenvalue of 0. One eigenvector is $(1, 1, \ldots, 1)$, where all vertices of G_1 and G_2 are placed at the same location; and the other eigenvector is $(a, \ldots, a, b, \ldots, b)$, where all vertices of G_1 are placed at x=a and all vertices of G_2 are placed at x=b.*

CIRCUITS	Quad	$\alpha = 1.5$	$\alpha = 1.2$	Linear
s1423	12,416	10,257	9,823	9,924
s15850	1,377,295	783,285	733,646	1,072,068
s35932	2,052,699	1,847,369	1,609,146	1,693,484
s38417	3,115,066	1,948,577	1,682,214	2,386,239
s38584	3,915,095	2,721,014	2,655,224	3,480,634

Table 1: *Comparison of Different Objective Function.*

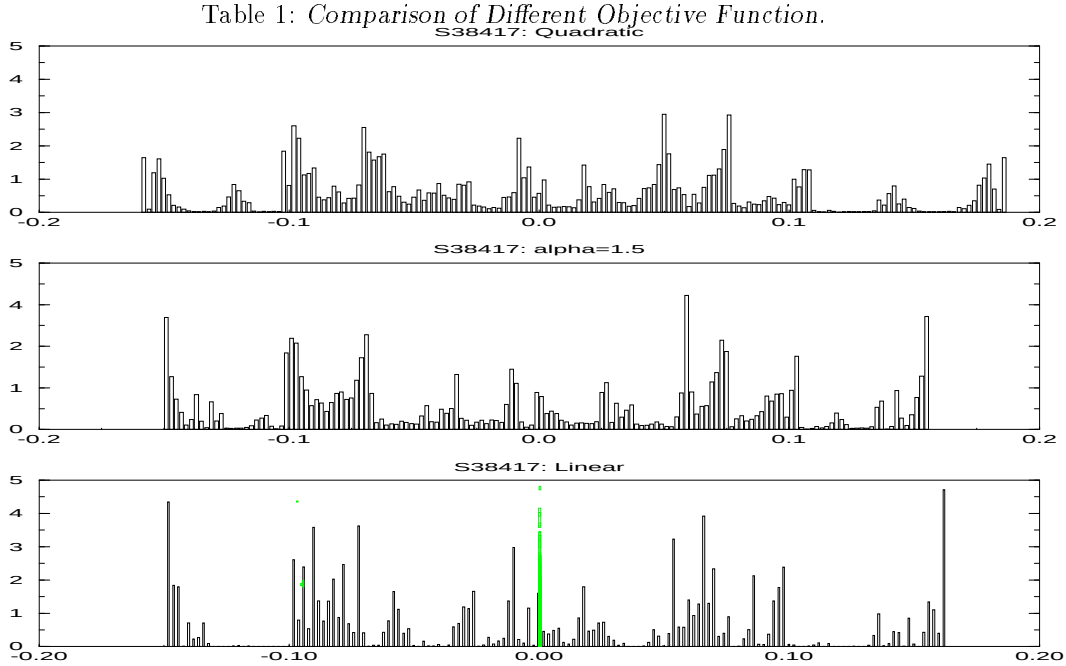

Figure 1: *The Continuous Linear Placement*

Theorem 4 *Given two graphs G_1 and G_2, and their Laplacians B_1 and B_2. Let V_i' be the eigenvector corresponding to the eigenvalue λ_i' of B_1, and let V_i'' be the eigenvector corresponding to the eigenvalue λ_i'' of B_2. Then the eigenvector corresponding to the eigenvalue λ_i' of $B = \begin{pmatrix} B_1 & 0 \\ 0 & B_2 \end{pmatrix}$ is $\begin{pmatrix} V_i' \\ 0 \end{pmatrix}$, where all nodes of G_2 are placed at x=0; and the eigenvector corresponding to the eigenvalue λ_i'' of B is $\begin{pmatrix} 0 \\ V_i'' \end{pmatrix}$, where all nodes of G_1 are placed at x=0.*

The above theorems deal with the Laplacian of a separated graph; however, these theorems also show the clustering property of the eigenvector approach. Intuitively, if the vertices in subgraph $G_i | i = 1, 2$ are "strongly" connected while the connection between G_1 and G_2 is comparatively loose, then continuous linear placement from the eigenvector approach will place the vertices of G_i around one location very closely, forming a natural cluster; and the linear placement based on this derived eigenvector solution will be quite arbitrary because some nodes are overlain each other.

Based on these observation, we use the continuous linear placement as the heuristic for the global connectivity information, while the local connectivity information is derived directly from the netlist. Our clustering algorithm is a bottom-up approach. First, we initialize each cluster which includes only one node of the original circuit. With bottom-up approach, we iteratively merge pairs of clusters which are strongly connected until we reduce the size of the clustered circuit to a predefined threshold T_k.

Algorithm 1 *Bottom-Up Clustering:*
1. *Initialize each cluster with only one node;*
2. *Do { Build new clustered circuit;*
   Calculate connectivity for pairs of clusters;
   Merge pairs of clusters with high connectity;
   } Repeat until sizeof(circuit)$\leq T_k$

We use the following metric to measure the connectivity for a pair of clusters i and j.

$$C = \frac{\sum_k c_{ij}^k}{W_i \times W_j} \times g(i,j)$$

where c_{ij}^k denotes the connective contribution from the kth net. If the kth net connects clusters i and j, $c_{ij}^k = \frac{1}{size(net_k)-1}$; otherwise, $c_{ij}^k = 0$. W_i denotes the size of the cluster i, this term can be viewed as the inertia of the cluster i; a cluster with large size is discouraged from participaring in additional merging operations in the clustering process. Finally, the g(i,j) is the weight from the continuous linear placement with the eigenvector approach, acting as the global connectivity information.

Although the eigenvector solution provides heuristic for the global connectivity information, it is quite costly to solve the continuous linear placement for a large circuit. Furthermore, it is very likely to introduce more

errors in the numerical computation for a large circuit. Due to these reasons, in our clustering implementation, when a circuit is very large, we only use local connectivity information in the clustering process; after the size of the clustered circuit becomes moderate, the global connectivity information from the continuous linear placement is used.

4 New Spectral Linear Placement

It is well known that clustering approach not only can significantly reduce the size of circuits but also can improve the performance of partitioning and placement. In this section, we combine the bottom-up clustering approach with our new spectral approach with α-order objective function for the linear placement problem. The new spectral linear placement algorithm, as shown below, can be viewed as consisting of three phases: 1) by using bottom-up clustering approach, groups of nodes with high connectivity are formed into clusters; 2) linear placement is done on this clustered circuit; 3) linear placement is done on the original circuit. In the last phrase, we first uncompress the clusters and locally do the linear placement for the nodes of each cluster; then we further improve the quality of the linear placement by using the decomposition algorithm[4], which use the recursive max-flow min-cut approach.

Algorithm 2 *New Spectral Linear Placement:*

1. *Call Alg. 1 to merge clusters;*
   Let $\Theta = \{C_i | i = 1, \ldots, K\}$ *be cluster set;*
2. *Build new clustered circuit for* Θ*;*
   Solve placement with α*-order function;*
3. *Let* $O = (C_1, \ldots, C_K)$ *be the cluster ordering;*

   i=2;
   Do { *Build a shrunken resistive network*
   $(s, v_{j1}, \ldots, v_{jl}, t)$ *for* C_i,
   where $v_{ji} (i = 1 \ldots l)$ *is component of* C_i*;*
   Solve this shrunken resistive network;
   Replace C_i *with ordering* $(v_{i1}, \ldots, v_{il})$ *in O;*
   i=i+1;
   } *while* $(i \leq K - 1)$.

The bottom-up clustering approach has been discussed in the last section, where we exploit the clustering property of the eigenvector approach as the global connectivity information. The main objective of the clustering is to reduce the circuit size without sacrificing the quality of partitioning and placement. The first phrase produces the clustered circuit. The spectral approach with α-order objective function, which has also been discussed in the second section, can be applied for either clustered circuit or original circuit on the linear placement. In this section, we mainly focus on the discussion of the third phrase of our new spectral linear placement algorithm.

In [3][17], the resistive network concept was introduced to solve the placement problem. By analogizing an electric network, where the conductance between node i and node j is equal to $-b_{ij}$ of the Laplacian B defined by Equation 2, it has been demonstrated that the placement problem is equivalent to that of choosing the node voltages of the electric network for which the power dissipation is a minimum. The node voltages of the electric network are analogy to the node coordinates of the placement problem[3][17]. However, different from the eigenvector approach, the resistive network can include the I/O pad specifications. Therefore, by modeling the I/O pads as fixed voltage sources applied to the electric network, we can fix some nodes at specific coordinates and the coordinates of the rest nodes are then the node voltages to be determined.

Therefore, the resistive network approach can be used to determine the coordinates of the nodes in a cluster, resulting in uncompressing the clustered circuit. Given the cluster ordering $O = (C_1, C_2, \ldots, C_K)$, when we try uncompressing the cluster C_i, we hope to determine the coordinates of the nodes in C_i while the ordering of clusters is not changed. With the resistive network approach, we shrink all the clusters to the left of C_i in O into a I/O pad s with 0 voltage, replace the cluster C_i with its components, and shrink all the clusters to the right of C_i in O into a I/O pad t with 1 voltage. Then the coordinates of the nodes in C_i are determined by the voltages of this shrunken resistive network.

Theorem 5 *In the above shrunken resistive network, the voltage of a movable node is between 0 and 1.*

Based on the above theorem, the nodes in C_i will be placed between two shrunken nodes s and t. Therefore, the uncompressing process doesn't break the cluster ordering defined by O.

After uncompressing the cluster $C_i | i = 2, \ldots, K-1$ in the ordering O, we obtain a solution to the linear placement problem for the original circuit. As shown in the experimental results of next section, such clustering and uncompressing approach obtains much better placement quality in terms of total wire length than the linearized eigenvector approach on original circuits. Using this better linear placement solution as the heuristic for the seed selection in the decomposition algorithm[4], the quality of the linear placement can be further improved.

5 Experiments

We implemented our new spectral linear placement and clustering algorithm(SLPC) for the linear placement problem, and tested it with a set of large benchmarks from MCNC. These benchmarks were also used in PANZA in [4].

We compare our SLPC to the lineared eigenvector approach(EIG2) and PANZA[4] in terms of total wire length. In comparison to EIG2, in order to be fair comparison, the recursive max-flow min-cut method is not applied in our SPLC method, we denote this version of SPLC as SPLC1. In comparison to PANZA, the recursive max-flow min-cut method is applied after SLPC1, and we denote this version of SPLC as SPLC2. For all these approaches, we use the star model for netlist modeling, and the net length is measured by the span of its node set.

In our SPLC approach, we set the predefined threshold for controlling the size of clustered circuits as $T_k = 100$ in Alg. 1, and we use $\alpha = 1.2$ in the α-order objective function.

The experimental results are summerized in Table 2. The results of SLPC are superior to the EIG2 and PANZA in terms of the total wire length. On the average, SLPC1 yields improvements of 18.5% over EIG2[4] and SLPC2 yields improvements of 8.8% over PANZA[4]. It is also important to note that SLPC approach yields much higher improvements for large circuits than for small circuits.

CIRCUITS	WIRE LENGTH				%improv over	
	EIG2	SLPC1	PANZA	SLPC2	EIG2	PANZA
s1423	9,924	9,508	9,060	9,254	4.2	-2.1
s9234	290,472	289,031	266,617	248,999	0.5	6.6
s13207	608,820	604,538	451,210	465,214	0.7	-3.1
s15850	1,072,068	725,042	728,103	591,372	32.4	18.8
s35932	1,693,484	1,265,986	920,607	871,937	25.3	5.3
s38417	2,386,239	1,799,575	1,711,139	1,325,547	24.6	22.5
s38584	3,480,634	2,015,480	1,728,237	1,487,277	42.1	13.9
Average					18.5	8.8

Table 2: *Linear Placement Results.*

6 Conclusion

In this paper, We develop a new spectral linear placement and clustering algorithm(SLPC). First we compare the spectral approach with various objective functions with experiments on MCNC benchmarks. We observe that the solution to the linear placement problem depends on both the sparsity of the continuous linear placement and the accurate measurement of the linear placement goal. The α-order objective function, which takes the sparsity and accurate measurement of the continuous linear placement into consideration at same time, has better performance on the linear placement than both the linear and quadratic objective functions. We also exploit the natural clustering property of the eigenvector approach, which we use as the global connectivity information in the bottom-up clustering process. This simple but efficient bottom-up clustering method not only significantly reduce the problem size but also improves the performance of the linear placement. The efficient resistive network approach is applied to uncompress clustered circuits. Therefore, our approach for the linear placement problem is first bottom-up approach to clustered circuits then top-down approach back to original circuits. Finally the decomposition algorithm is applied to further improve the quality of the linear placement. Our experiments on MCNC benchmarks demonstrate that our spectral linear placement and clustering approach receives much better performance than the previous ones.

References

[1] D.Adolphson, and T.C. Hu, "Optimal Linear Ordering," SIAM J. Appl. Math. Vol.25(3), Nov. 1973, pp403-23.

[2] C.K. Cheng, "Linear Placement Algorithms and Applications to VLSI Design," NETWORKS, Vol.17, 1987, pp439-64.

[3] C.K. Cheng, and E.S. Kuh, "Module Placement Based on Resistive Network Optimization," IEEE Trans. on Computer-Aided Design, Vol.CAD-3(3), Jul. 1984, pp218-25.

[4] J. Li, J. Lillis and C.K. Cheng, "Linear Decomposition Algorithm for VLSI Design Applications," Proc. IEEE Int. Conf. on Computer-Aided Design, 1995, pp223-8.

[5] L. Hagen, and A.B. Kahng, "Fast Spectral Methods for Ratio Cut Partitioning and Clustering," Proc. IEEE Int. Conf. on Computer-Aided Design, 1991, pp10-3.

[6] B.M. Riess, K. Doll, and F.M. Johannes, "Partitioning Very Large Circuits Using Analytical Placement Techniques," Proc. 31th ACM/IEEE Design Automation Conference, 1994, pp646-51.

[7] T.C. Hu, and K. Moerder, "Multiterminal Flows in a Hypergraph," Hu and Kuh eds., IEEE Press, 1985.

[8] 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(3), Mar. 1991, pp356-65.

[9] K.M. Hall, "An r-Dimensional Quadratic Placement Algorithm, Management Science," Vol.17(3), Nov. 1970, pp219-29.

[10] R.S. Tsay, and E.S. Kuh, "A unified approach to partitioning and placement (VLSI layout)," IEEE Trans. on Circuits and Systems, Vol.38(5), May 1991, pp521-33.

[11] G. Sigl, K. Doll, and F.M. Johanness, "Analytical Placement: A Linear or a Quadratic Objective Function?" Proc. 28th ACM/IEEE Design Automation Conference, 1991, pp427-32.

[12] C.J. Alpert, and A.B. Kahng, "A General Framework for Vertex Orderings, With Applications to Netlist Clustering", Proc. IEEE Int. Conf. on Computer-Aided Design, 1994, pp63-7.

[13] J. Cong, and M. Smith, "A Parallel Bottom-up Clustering Algorithm with Applications to Circuit Partitioning in VLSI Design", Proc. 30th ACM/IEEE Design Automation Conference, 1993, pp755-60.

[14] J. Cong, L. Hagen, and A.B. Kahng, "Random Walks for Circuits Clustering", Proc. IEEE Intl. Conf. on ASIC, Jun. 1991, pp14.2.1-4.

[15] L. Hagen, and A.B. Hahng, "New Spectral Methods for Ratio Cut Partitioning and Clustering", IEEE Trans. on Computer Aided Design, Vol.11(9), Sep. 1992, pp1074-85.

[16] J. Garbers, H.J. Promel, and A. Steger, "Finding clusters in VLSI circuits", Proc. IEEE Int. Conf. on Computer-Aided Design, 1990, pp520-3.

[17] R.S. Tsay, E.S. Kuh, and C.P. Hsu, "PROUD: A Fast Sea-of-Gates Placement Algorithm", Proc. 25th ACM/IEEE Design Automation Conference, 1988, pp318-23.