is likely to be worthwhile; on the other hand, very detailed and systematic investigations are required if meaningful conclusions are to be drawn.

The "litany of variants" also reveals that applying an FM-based algorithm to yet another problem formulation can require varying degrees of innovation. For example, the balance term in the cost function of [144] is obvious: increase the penalty proportionally to the deviation from bisection. On the other hand, certain changes in formulation may appear simple (e.g., partitioning onto an underlying tree structure, by Vijayan [186]), but issues such as efficient gain updating can make the implementation differences non-trivial. Many new problem formulations seem very similar to previous ones (modulo some extra constraints or modified cost functions), hence care must be taken to discern the works which more substantially advance the field through their combination of relevance, non-obvious solution, and experimental methodology.

4 Geometric Representations

A geometric representation of the circuit netlist can provide a useful basis for a partitioning heuristic, since speedups and special "geometric" heuristics become possible. For example, computing a minimum spanning tree of a weighted undirected graph requires $O(n^2)$ time, in general, but only $O(n \log n)$ time for points in 2-dimensional geometric space [150]. Single-source shortest path, all-pairs shortest paths, and matching are other examples of problems that can be solved more efficiently for geometric instances. This section discusses partitioning approaches based on finding a geometric representation of a graph or hypergraph and applying "geometric" algorithms to find a partitioning solution. We focus on three primary types of representations:

- A **1-dimensional representation** or a *linear ordering* is a sequential list of the modules. Generally, modules that are closely connected should lie close to each other in the ordering, so that the ordering can reveal the netlist's structure. Indeed, problems in the sparse matrix computation literature such as finding minimum bandwidth, minimum profile, and minimum fill-in orderings exactly correspond to this type of ordering problem [147]. A linear ordering may also be viewed as a 1-dimensional placement, and vice-versa; consequently, some of the partitioning approaches we discuss below were originally designed for 1-dimensional placement

- A **multi-dimensional representation** is a set of n points in d-dimensional space with $d > 1$, where each point maps to (represents) a unique module. This representation implicitly defines a distance relation between every pair of modules, e.g., the Euclidean distance between their corresponding points. Geometric clustering algorithms may be applied to the set of points, possibly in conjunction with other graph-based algorithms. Such a representation is also common 2-dimensional cell placement.

- A **multi-dimensional vector space** representation can arise in two distinct ways. Using one approach, the vector space consists of indicator n-vectors (corresponding to bipartitioning solutions), and the problem becomes one of finding the direction of the best indicator vector [68]. Using the other approach, the n modules are mapped to n vectors in d-space and the vectors are clustered together to form both a vector partitioning and a module partitioning [12] [6] [37]. A major advantage of the vector space approach is that spectral methods can be used to construct a vector space that optimally captures the netlist information vis-a-vis partitioning, i.e., the optimal indicator vector direction or the optimal vector partitioning solution will map to the optimal graph partitioning solution.

Spectral methods are of primary importance in constructing geometric representations; their discussion requires the following notation. Assume the netlist is represented as a weighted undirected graph $G(V, E)$ with adjacency matrix $A = (a_{ij})$. The $n \times n$ *degree matrix* D is given by $d_{ii} = deg(v_i)$ with $d_{ij} = 0$ if $i \neq j$. The $n \times n$ Laplacian matrix of G is defined as $Q = D - A$. An n-dimensional vector $\vec{\mu}$ is an *eigenvector* of Q with *eigenvalue* λ if and only if $Q\vec{\mu} = \lambda\vec{\mu}$. We denote the set of eigenvectors of Q by $\vec{\mu_1}, \vec{\mu_2}, \ldots, \vec{\mu_n}$ with corresponding eigenvalues $\lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n$. We assume that the eigenvectors are normalized, i.e., for $1 \leq j \leq n$, $\vec{\mu_j}^T\vec{\mu_j} = ||\vec{\mu_j}||^2 = 1$. Let Δ_d denote the $d \times d$ diagonal *eigenvalue matrix* with entries $\lambda_1, \lambda_2, \ldots, \lambda_d$, and let $U_d = (\mu_{ij})$ denote the $n \times d$ *eigenvector matrix* with columns $\vec{\mu_1}, \vec{\mu_2}, \ldots, \vec{\mu_d}$. Notice that $Q = U_n \Delta_n U_n^T$. Some works use the eigenvectors of A instead of Q; however, the Laplacian has becoming more popular because it has the following following desirable properties (the first two due to Q being positive semi-definite) [139]:

- The eigenvectors are mutually orthogonal, and hence form a basis in n-dimensional space.
- Each eigenvalue λ_j of Q is real.
- The smallest eigenvalue $\lambda_1 = 0$ and has a corresponding eigenvector $\vec{\mu_1} = [\frac{1}{\sqrt{n}}, \frac{1}{\sqrt{n}}, \ldots, \frac{1}{\sqrt{n}}]^T$.[8]
- If G is connected then $\lambda_2 > 0$. (In general, the multiplicity of 0 as an eigenvalue of Q is equal to the number of connected components of G.)

[8] This property combined with the orthogonality property implies that the entries of each $\vec{\mu_j}$ with $j > 1$ sum to zero. This does not necessarily hold for the spectra of A, and further the eigenvalues of A will generally be both positive and negative. However, in practice, a diagonal matrix B is often added to A to ensure positive semi-definiteness and the computability of eigenvectors, and also to obtain properties similar to Q (e.g., Donath and Hoffman [59] use $B = -D$, thereby obtaining the spectra of -Q). Currently, the relationship between the spectra of A and Q are not well-understood, though we believe that most theoretical results in the spectral literature can be equivalently derived with either Q or A. To this end, we discuss as many of the works in this section as possible in terms of Q, even if the original paper used A. For example, the discussion of Barnes's work [18] below is given in terms of Q, which allows us to establish the equivalence of lower bounds due to Barnes and to Donath and Hoffman. By "equivalence", we mean that following the theorem derivation of each work using the spectra of Q leads to the identical result.

The remainder of this section is organized as follows. First, we discuss the one-dimensional placement algorithm of Hall [89], and see how the second eigenvector $\vec{\mu_2}$ gives the optimal solution for minimum-squared wirelength placement. A linear ordering of modules is obtained by sorting the entries of $\vec{\mu_2}$, and a heuristic bipartitioning can be derived from splitting the ordering. Next, we describe an extension of Hall's multi-way partitioning approach due to Barnes [18], as well as some other extensions (e.g., [79] [37]). These approaches use k eigenvectors to derive a k-way partitioning; Barnes' approach explicitly tries to map each cluster to a single eigenvector. Third, we discuss an alternative approach that uses multiple eigenvectors to generate a bipartitioning. This *linear probe* technique [68] first constructs a vector space using the eigenvectors as a basis, then generates probes which search the vector space to find good bipartitionings. Fourth, we discuss utilizing this vector space to construct a vector partitioning problem that is equivalent to many multi-way graph partitioning formulations. Finally, we discuss techniques for deriving partitioning solutions from a linear ordering, and show how many of the approaches presented in this section can be used to construct such linear orderings.

4.1 Hall's Quadratic Placement

As defined in Section 2.1, we let $\vec{x}$ denote the $0-1$ indicator vector for a given bipartitioning P^2. The Min-Cut bipartitioning objective ($F(P^2) = E(C_1)$) can now be written as

$$2 \cdot F(P^2) = \vec{x}^T Q \vec{x} = \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij}(x_i - x_j)^2 \qquad (4.1)$$

All the $(x_i - x_j)^2$ terms in the double summation of equation (4.1) are zero, unless v_i and v_j are in different clusters. In this case, $(x_i - x_j)^2 = 1$, and the cost a_{ij} of each cut edge appears twice in the sum, so the double summation evaluates to $2 \cdot F(P^2)$. Also observe that equation (4.1) gives the squared wirelength of the 1-dimensional placement given by $\vec{x}$, even when the coordinates are allowed to be *non-discrete*, i.e., the x_i's are not restricted to integer values. Under the constraint $||x||^2 = \vec{x}^T \vec{x} = 1$, Hall [89] showed that $\vec{x} = \vec{\mu_2}$ gives the optimal nontrivial 1-dimensional placement, with squared wirelength equal to λ_2 (note that $\vec{\mu_1}$ gives the trivial zero-wirelength solution with all modules placed at coordinate $\frac{1}{\sqrt{n}}$).

The significance of Hall's result is that it provides the optimal non-discrete solution for Min-Cut Bipartitioning. Although a non-discrete solution for $\vec{x}$ is meaningless, this result suggests heuristically finding the discrete solution "closest" to $\vec{\mu_2}$. Given cluster size constraints $|C_1| = m_1$ and $|C_2| = m_2$, the closest discrete solution is obtained by sorting the coordinates of $\vec{\mu_2}$ – the m_1 modules with the highest coordinates are placed in C_1 and the m_2 lowest modules with the lowest coordinates are placed in C_2. The reverse assignment is also attempted, and one of these solutions is guaranteed to be closest to $\vec{\mu_2}$ as measured by the Frobenius norm. One can view this sorting algorithm as moving the smallest coordinates to C_1 (corresponding to $x_i = 0$) and the largest coordinates to C_1 (corresponding to $x_i = 1$).[9] This approach of finding the partitioning which best approximates the second smallest eigenvector was first used by Barnes [18] and is commonly known as *spectral bisection* (when $m_1 = m_2 = \frac{n}{2}$). This algorithm has also been widely used by the sparse matrix computation community; Pothen et al. [148] have used it as the basis of a vertex separator algorithm and Hendrickson and Leland [96] have extended it to partitioning onto hypercube or mesh architectures. Also, Hagen and Kahng [80] extended spectral bisection to ratio cut bipartitioning by choosing the best ratio cut that results from each possible split of the sorted coordinates of $\vec{\mu_2}$.

We observe that spectral bisection may perform arbitrarily worse than optimal, as illustrated by the following example. Consider the graph in Figure 8 in which each circle represents a clique, so that two $\frac{n}{4}$-cliques A and C are each connected by a single edge to the $\frac{n}{2}$-clique B. Since $\vec{\mu_2}$ gives the optimum 1-dimensional squared wirelength placement, sorting the entries in $\vec{\mu_2}$ yields the coordinates for modules in A that are followed by coordinates for those in B, that are in turn followed by coordinates for those in C. Consequently, spectral bisection will split B into equal halves, cutting $(\frac{n}{4})^2 = \frac{n^2}{16}$ edges. This solution is an $\Omega(n^2)$ factor worse than the optimum bisection cutsize of two edges, obtained by assigning A and C to one cluster and B to the other.

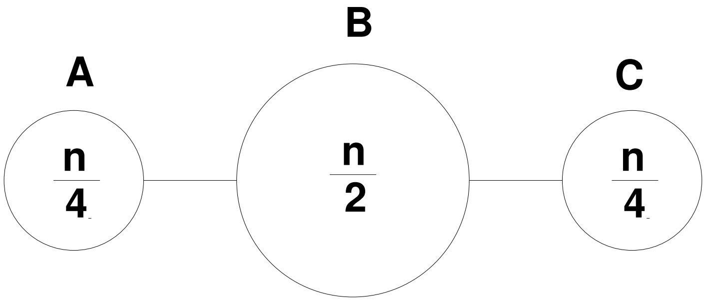


Figure 8: Pathological instance for spectral bisection. Each circle represents a clique of modules.

4.2 Mapping k Clusters to k Eigenvectors

Hall's approach can be extended to multi-way partitioning by associating each cluster with an eigenvector with small eigenvalue. The 1-dimensional placement solution $\vec{\mu_2}$ yields the minimum squared wirelength of λ_2, but $\vec{\mu_3}$ gives the next best solution with cost λ_3, etc. – again subject to the $||\vec{x}||^2 = 1$ constraint. For each of these high-quality 1-dimensional eigenvector placements, the closest indicator

[9]This interpretation is not exactly correct since $\vec{x} = \vec{\mu_2}$ has $\sum_{i=1}^{n} x_i = 0$ while the sum for an indicator vector of a bisection is $\frac{n}{2}$. Furthermore, the indicator vector for a bisection also has $||\vec{x}||^2 = \frac{n}{2}$ as opposed to $||\vec{\mu_2}||^2 = 1$. These inconsistencies are addressed by using coordinates $(-\frac{1}{\sqrt{n}}, \frac{1}{\sqrt{n}})$ instead of $(0,1)$ for the indicator vector. Note that these coordinates correspond to the ratioed assignment matrix R defined in the next subsection.

vector can be found, and the k indicator vectors can be used to construct a k-way partitioning.

Assume that the standard multi-way partitioning objective $F(P^k) = \sum_{h=1}^{k} |E(C_h)|$ applies, and that prescribed cluster sizes $m_1 \geq m_2 \geq \ldots \geq m_k \geq 0$ are given. Let M denote the $k \times k$ diagonal matrix with entries $m_1, m_2, \ldots, m_k$. Although X denotes the assignment matrix for P^k, sometimes it will be advantageous to represent P^k as the $n \times k$ *ratioed assignment matrix* $R = (r_{ih})$ where $r_{ih} = \frac{1}{\sqrt{|C_h|}}$ if $v_i \in C_h$ and $r_{ih} = 0$ otherwise. Notice that $\vec{R_h}$, the h^{th} column of R, has magnitude one, which satisfies the constraint in Hall's 1-dimensional placement formulation. Let $P = (p_{ij})$ be the $n \times n$ *partition matrix* with $p_{ij} = 1$ if v_i and v_j are in the same cluster and $p_{ij} = 0$ otherwise; observe that $P = XX^T = RMR^T$. Finally, define the $n \times n$ *ratioed partition matrix* P^R to be a scaled version of P with ij entry equal to $\frac{1}{|C|}$ if v_i and v_j are in the same cluster C, and 0 otherwise (so $P^R = RR^T$). Barnes noted that

$$||Q + P||^2 = ||Q||^2 + 2 \sum_{1 \leq i,j \leq n} q_{ij} p_{ij} + ||P^2|| = ||Q||^2 + 2F(P^k) + \sum_{i=1}^{k} m_i^2 \tag{4.2}$$

Since $||Q||$ and $\sum_{i=1}^{k} m_i^2$ are fixed constants, minimizing $||Q+P||^2$ and minimizing $F(P^k)$ are equivalent partitioning objectives. The Hoffman-Wielandt inequality states that if A and B are real $n \times n$ symmetric matrices with eigenvalues $\alpha_1 \geq \alpha_2 \geq \ldots \geq \alpha_n$ and $\beta_1 \geq \beta_2 \geq \ldots \geq \beta_n$ respectively, then

$$||A - B||^2 \geq \sum_{i=1}^{n} (\alpha_i - \beta_i)^2.$$

It is not hard to show that the eigenvalues of $-P$ are $-m_1, -m_2, \ldots, -m_k, 0, 0, \ldots, 0$, which yields

$$||Q - (-P)||^2 \geq \sum_{i=1}^{k} (\lambda_i - (-m_i))^2 + \sum_{i=k+1}^{n} (\lambda_i - 0)^2 = \sum_{i=1}^{k} (\lambda_i + m_i)^2 + \sum_{i=k+1}^{n} \lambda_i^2 \tag{4.3}$$

Combining equation (4.3) with equation (4.2) yields the equivalent of the famous Donath and Hoffman [59] lower bound: $F(P^k) \geq \sum_{i=1}^{k} \lambda_i m_i$.[10] Actually, the lower bound of [59] is stronger, since it holds for not only the eigenvalues of Q, but also the eigenvalues of $D - A$ as long as the diagonal matrix D has $trace(D) = \sum_{i=1}^{n} deg(v_i)$. It is thus possible to varying D to increase $\sum_{i=1}^{k} \lambda_i m_i$ and improve the lower bound.

Observe that the Donath-Hoffman lower bound also implies that an optimal partitioning solution would be obtained if each $\vec{R_i}$ could be set to $\vec{\mu_i}$. However, setting $R = U_k$ generally does not yield a valid ratioed partition matrix. Barnes [18] justifies his approach to finding a valid R that best

[10] Combining the two equations yields

$$2F(P^k) \geq -||Q||^2 - \sum_{h=1}^{k} m_i^2 + \sum_{i=1}^{k} (\lambda_i + m_i)^2 + \sum_{i=k+1}^{n} \lambda_i^2 = -||Q||^2 + 2\sum_{i=1}^{k} \lambda_i m_i + \sum_{i=1}^{n} \lambda_i^2.$$

It is not difficult to show that $||Q||^2 = ||\Delta_n||^2$ which yields the desired result.

approximates U_k as follows. Since $Q = U_n \Delta_n U_n^T$ and $P = RMR^T$, equation (4.2) implies that the partitioning objective can be written as

$$||Q + P||^2 = ||U_n \Delta_n U_n^T + RMR^T||^2 = ||\Delta_n + U_n^T RM(U_n^T R)^T||^2 \tag{4.4}$$

If R could be chosen such that $U_n^T R = J$, where J is the $n \times k$ matrix having the $k \times k$ identity matrix I_k as its first k rows and all other entries zero, we would have equality in equation (4.3), and R would represent the optimal partitioning solution. Since such a choice of R is generally impossible, Barnes chooses to minimize the error $||U_n^T R - J||^2 = ||U_k - R||^2$ by setting up a transportation problem which can be solved efficiently (see Section 5.3.3). To minimize this error for a given vector $\vec{R_i}$ (representing C_i), $|\vec{\mu_i}^T \vec{R_i}|$ should be maximized; this occurs when the nonzero entries of $\vec{R_i}$ are the m_i largest (or smallest) coordinates of $\vec{\mu_i}$. Of course, it may not be possible to choose each $\vec{R_i}$ in this fashion, because a module might be assigned to more than one cluster; Barnes' transportation formulation optimally resolves this conflict. Notice that for $k = 2$, this formulation reduces to minimizing $||U_2 - R||^2$ and the optimal solution is derived by sorting the coordinates of $\vec{\mu_2}$ as discussed above.

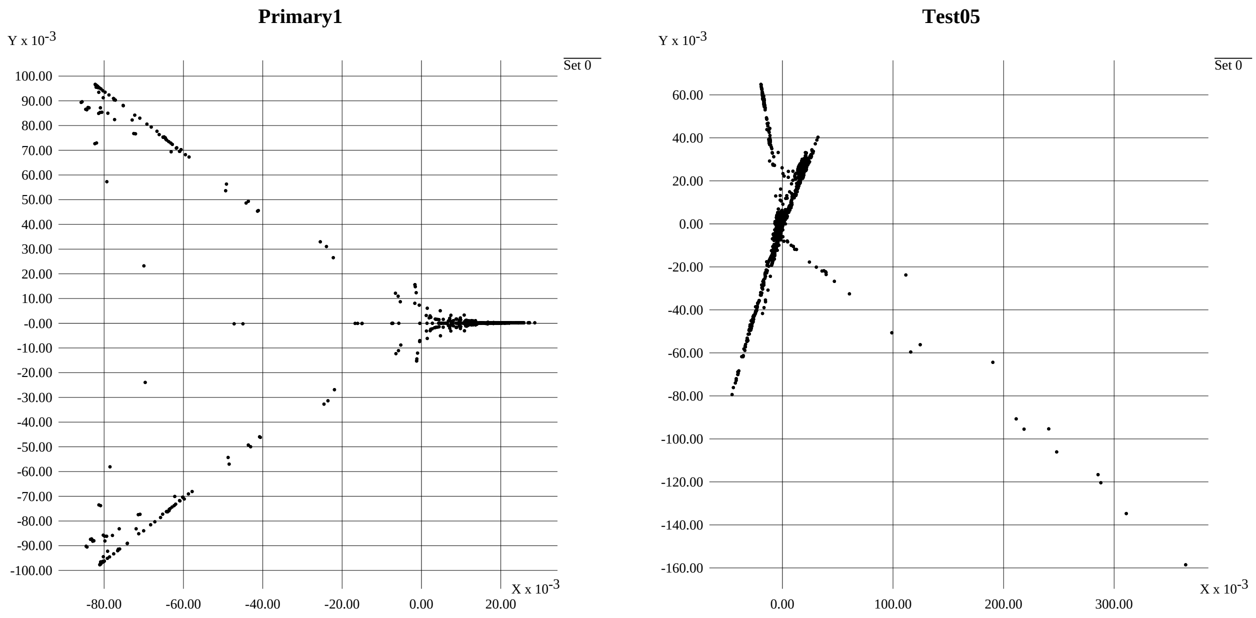
Barnes' algorithm was applied to VLSI circuits by Hadley et al. [79], who also incorporated a new clique net model to obtain a graph representation and FM post-processing. Vannelli and Rowan [185] also applied the algorithm to two variant constructions of the adjacency matrix, which respectively use $a_{ij} = |N(v_i) \backslash N(v_j)| + |N(v_i) \backslash N(v_j)|$ and $a_{ij} = (|N(v_i) \backslash N(v_j)| + |N(v_i) \backslash N(v_j)|)/(|N(v_i)| + |N(v_j)|)$; observe that these graph representations of the hypergraph do not preserve sparsity. Rendl and Wolkowicz [154] observe that Barnes' approach "only makes sense" if U_k is very close to an optimal solution. Consequently, [154] first relaxes the integer constraints of the assignment matrix X to find an X close to U_k, then uses the transportation formulation to find the closest legal solution to X. They also permit the diagonal entries of A to be perturbed before computing the eigenvectors, leading to tighter lower bounds than those given by Donath and Hoffman [59]. Further, they extend the average case bisection bound of Boppana [27] to $k > 2$ (see [133] for a more complete discussion of spectrally derived lower bounds).

Chan et al. [37] extend the Donath and Hoffman bound $F(P^k) \geq \sum_{i=1}^{k} \lambda_i m_i$ to Scaled Cost, proving for this objective that $F(P^k) \geq \sum_{i=1}^{k} \lambda_i$. They argue that since R is an approximation of U_k, than approximation for $P^R = RR^T$ should be given by $U_k U_k^T$, i.e., the matrix having as its ij^{th} entry the dot product of the i^{th} row of U_k with the j^{th} row of U_k. Viewing each module v_i as a vector in k-dimensional space with coordinates given by the i^{th} row of U_k, the ij^{th} entry of $U_k U_k^T$ is the angle or *directional cosine* between modules v_i and v_j. Chan et al. consider the directional cosine between two modules as a distance measure: the larger the directional cosine, the more likely the two modules should be placed in the same cluster. A directional cosine of zero between v_i and v_j implies that the ij^{th} entry in P_R is zero (since P_R should approximate $U_k U_k^T$), meaning that v_i and $v_j$should be assigned to different clusters. The KP algorithm of [37] finds an orthogonal basis with k "prototype vectors" and constructs k clusters by assigning each module to its closest prototype according to the directional cosines measure. Chan et al. [38] have also adapted their KP approach to FPGA partitioning by assigning each module to its closest prototype while observing size and I/O constraints on the clusters (FPGA devices).

We believe that the Barnes and KP approaches may have limited performance potential. Both approaches assume that U_k can be perturbed to give a legal ratioed assignment matrix R: Barnes chooses R to minimize the error $||U_k - R||^2$, and Chan et al. explicitly assume that R is an approximation of U_k. Thus, each $\vec{R_i}$ representing cluster C_i should be close to $\vec{\mu_i}$ in the ideal case, but for real VLSI circuits $\vec{R_i}$ and $\vec{\mu_i}$ will generally be quite different. A given eigenvector will typically have a few outliers – modules with large coordinates – but a large majority of modules will have coordinates very close to zero; these coordinates would have to be perturbed considerably to achieve a valid ratioed assignment matrix. To see this, consider the coordinates for $\vec{\mu_2}$ (on the x-axis) and $\vec{\mu_3}$ (on the y-axis) for the Primary1 and Test05 benchmarks shown in Figure 9. Exactly 603 of the 833 modules for Primary1 have zero (to three decimal places) as their y-coordinate, and these modules are strongly clustered together. The embedding actually admits a very natural 3-way partitioning with sizes $681, 76, 76$ but does not reveal a natural bisection. This phenomenon is even more apparent for Test05, as 12 outliers with x-coordinate larger than 0.1 force a strong agglomeration of points around zero, again leading to a difficult clustering task. We believe that even when minimized, the error $||U_k - R||^2$ will usually be large, i.e., each $\vec{R_i}$ will not be close to its corresponding $\vec{\mu_i}$. This potentially large error may be magnified by squaring U_k: Chan et al. assume that $P^R = RR^T$ is an approximation of $U_k U_k^T$ (but their KP algorithm does not actually square U_k, since computing all n^2 entries of $U_k U_k^T$ would be too expensive).

Hall [89] also proposed a method for multi-way partitioning using the same eigenvector embedding as the authors of[37]. Since $\vec{\mu_2}$ gives the optimal 1-dimensional placement with squared wirelength, Hall reasoned that the other low-cost eigenvector placements could also be utilized. He suggested placing the the coordinates of d eigenvectors orthogonally to yield a d-dimensional geometric embedding of the netlist, as in Figure 9. Hall then proposed to "cluster" the embedding, but did not specify a heuristic. Alpert and Kahng [3] explored Hall's idea by applying geometric clustering algorithms to these spectral embeddings, using Euclidean distance as the dissimilarity measure between pairs of modules. In [5], they constructed a 1-dimensional ordering over the same d-dimensional embedding using a spacefilling curve heuristic to the traveling salesman problem [20]. The ordering was then split into a partitioning using dynamic programming (see Section 4.5).

We believe that neither directional cosines nor Euclidean distance is the proper similarity (or



(a) (b)

Figure 9: The 2-dimensional embedding of the (a) Primary1 and (b) Test05 MCNC benchmarks, with $\vec{\mu_2}$ plotted on the x-axis and $\vec{\mu_3}$ plotted on the y-axis.

dissimiliarity) measure between spectrally embedded modules. The theoretical results described in the next two subsections indicate that the "proper" spectral embedding should be scaled by the eigenvalues, and that the module similarity measure should be a "vector sum".

4.3 Probes in Multi-Dimensional Vector Space

We have noted that a potential difficulty with the approaches of [18] [37], namely, that each cluster is associated with a single eigenvector, but the cluster's indicator vector may be far from its eigenvector. Blanks [24] also shared this intuition and illustrated (for placement, though it also holds for bipartitioning) that the closest legal solution to the eigenvector solution $\vec{\mu_2}$, i.e., the indicator vector that maximally projects onto $\vec{\mu_2}$, will generally not be optimum. Figure 10 gives an abstract visualization of this concept. The solution space consists of all n-dimensional vectors $\vec{x}$, the plane represents the subspace of all legal indicator vectors, and the optimum non-discrete solution for min-cut partitioning is given by the point $\vec{x} = \vec{\mu_2}$. The ellipsoids correspond to constant cost degradation, i.e., the smallest ellipsoid is the set of solutions with cost $\lambda_2 + \epsilon$, the second smallest ellipsoid contains solutions with cost $\lambda_2 + 2\epsilon$, etc; Blanks proved that these cost surfaces are indeed ellipsoidal. The optimum legal bipartitioning solution B is the intersection of the subspace of legal solutions with the smallest possible ellipsoid (such that this intersection is nonempty). The indicator vector $\vec{x}$ that maximally projects onto $\vec{\mu_2}$, labeled with A in the Figure, certainly does not have to be identical to B.

An alternative approach to finding the closest legal solution might be to combine several high-quality eigenvector solutions into a new solution that is closer to the subspace of legal solutions. The same figure shows a combined solution whose closest legal solution, C, is closer to optimal than A. Note that all linear combinations of the first d eigenvector solutions lie in the d-dimensional subspace spanned by these eigenvectors. This observation is the genesis behind the probing approach Frankle and Karp [68] [67]: they search this entire subspace for a good solution rather than associating a solution with a single eigenvector.

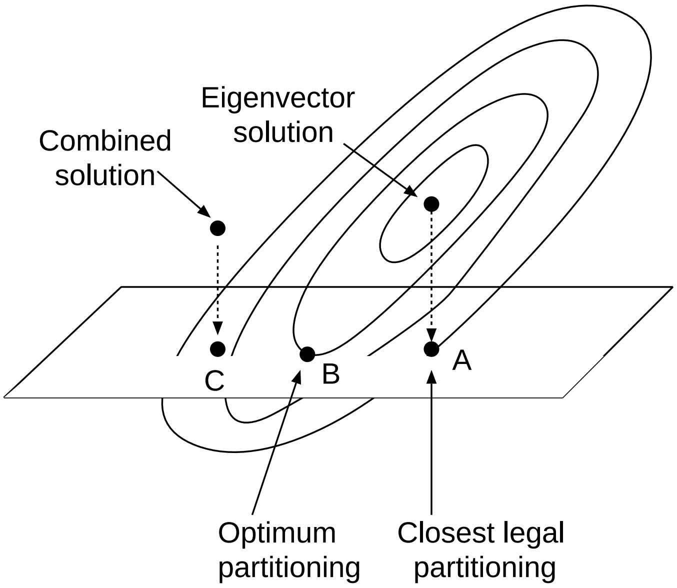


Figure 10: Abstraction of Hall's eigenvector approach as shown in [24].

Frankle and Karp [68] developed the following probing technique for finding such non-discrete combined solutions in the multi-dimensional eigenspace. Since the n eigenvectors of Q form an orthogonal n-dimensional basis, any indicator vector can be expressed as a sum of projections onto each of the eigenvectors, i.e., $\vec{x} = \sum_{j=1}^{n} \alpha_j(\vec{x})\vec{\mu_j}$ where $\alpha_j(\vec{x}) = \vec{x}^T\vec{\mu_j}$. Substituting this value for $\vec{x}$ in equation (4.1) yields

$$F(P^2) = \sum_{j=1}^{n} \alpha_j(\vec{x})^2 \lambda_j \qquad (4.5)$$

If the cluster sizes $|C_1| = m_1$ and $|C_2| = m_2$ are fixed, then $\sum_{j=1}^{n} \alpha_j(\vec{x})^2 = m_2$ is a constant (since $\vec{x}$ is the indicator vector for C_2). This observation, along with equation (4.5), implies that the goal is to find an indicator vector $\vec{x}$ that projects maximally onto the eigenvectors with small eigenvalues: The eigenvalue λ_i is the coefficient for the $\alpha_i(\vec{x})$ term; hence, when λ_i is large $\alpha_i(\vec{x})$ should be small, and vice versa. If the large eigenvalues are ignored and only the first d terms in the sum are considered, the problem becomes one of finding the indicator vector with maximal projection onto the subspace spanned by the first d eigenvectors. Frankle and Karp express equation (4.5) as the following maximization problem:

$$\text{Maximize:} \quad Hm_2 - F(P^2) = H\sum_{j=1}^{n}\alpha_j(\vec{x})^2 - \sum_{j=1}^{n}\alpha_j(\vec{x})^2\lambda_j = \sum_{j=1}^{n}\alpha_j(\vec{x})^2(H-\lambda_j) \tag{4.6}$$

for some $H \geq \lambda_n$. Let V_d denote the $n \times d$ *scaled eigenvector matrix* with column j equal to $\vec{\mu_j}\sqrt{H-\lambda_j}$. The right hand side of equation (4.6) can now be expressed as $||\vec{x}^T V_n||$. Given any n-dimensional vector $\vec{y}$, i.e., a *probe direction*, Frankle and Karp show that the legal indicator vector $\vec{x}$ that maximizes $\vec{x}^T V_n \vec{y}$ can be found efficiently by sorting and splitting the entries of $V_n\vec{y}$, as in spectral bisection. Here, the vector $\vec{y}$ serves as the combined solution point in Figure 10. Figure 11 illustrates this approach for two dimensions. The vector $\vec{x}^T V_2$ that projects maximally onto $\vec{y}$ gives the optimal solution vector for probe direction $\vec{y}$.

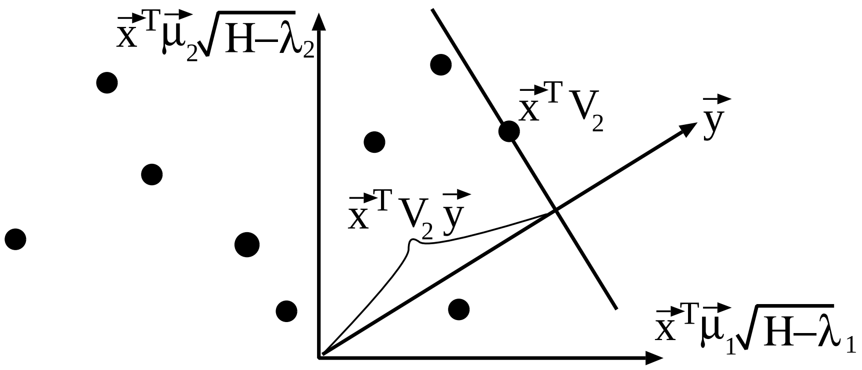


Figure 11: Illustration of the probing approach of [68].

Frankle and Karp heuristically search over various d-dimensional vector probes $\vec{y}$, and find for each the indicator vector $\vec{x}$ that maximizes $\vec{x}^T V_d \vec{y}$. Both randomized and iterated probing techniques are used to construct candidate probe vectors. By restricting $\vec{y}$ to d dimensions, only vectors that lie in the subspace spanned by the first d eigenvectors are considered. We believe that this is a sound strategy, given the equation (4.5) intuition that a good solution $\vec{x}$ should strongly project onto the first few eigenvectors. The problem reduces to finding a probe direction $\vec{y}$ that is as close as possible to a legal solution. Note that for $d = 2$, any probe vector will return the solution obtained by sorting and splitting the entries of $\vec{\mu_2}$.

4.4 Vector Partitioning

The probe approach can also be viewed another way [6]: for each module v_i, observe that the d-dimensional vector corresponding to the i^{th} row of V_d is actually the indicator vector for the single-module cluster $\{v_i\}$ after projecting onto the subspace spanned by the first d eigenvectors, and scaling the j^{th} coordinate by $\sqrt{H-\lambda_j}$, $1 \leq j \leq d$. If $\vec{x}$ is the 0-1 indicator vector for $\{v_i\}$, then its projection onto the scaled eigenspace is

$$\vec{y_i} = [\vec{x}^T\vec{\mu_1}\sqrt{H-\lambda_1}, \vec{x}^T\vec{\mu_2}\sqrt{H-\lambda_2}, \ldots, \vec{x}^T\vec{\mu_d}\sqrt{H-\lambda_d}]^T = [\mu_{i1}\sqrt{H-\lambda_1}, \mu_{i2}\sqrt{H-\lambda_2}, \ldots, \mu_{id}\sqrt{H-\lambda_d}]^T$$

i.e., the i^{th} row of V_d. Observe that if $\vec{Y} = \sum_{i=1}^{n} x_i \cdot \vec{y_i}$, then $\vec{Y} = \vec{x}^T V_d$. Recall that Frankle and Karp showed bipartitioning equivalent to maximizing $||\vec{x}^T V_n||^2$; we now have that bipartitioning is also equivalent to maximizing $||Y||^2$ when $d = n$. We therefore seek a subset of vectors $\vec{y_i}$ (corresponding to $x_i = 1$) whose sum has the largest possible magnitude. Graph bipartitioning exactly reduces to this vector subset problem for $d = n$ and $H = \lambda_n$.

Arun and Rao [12] also noted this vector partitioning formulation, but called it a "geometric clustering" problem. The derivation is somewhat different, relying on the factorability of the adjacency matrix, i.e., if A can be expressed as $A = CC^T$ where C is an $n \times d$ matrix, then A only has rank d. This implies that the columns of C span the same space as the columns (or rows) of A, hence the columns of C form a basis for this space. The indicator vector for $\{v_i\}$ becomes the i^{th} row of C when expressed in this basis. Viewing the rows of C as points in d-space centered at the origin, the bipartitioning problem reduces to finding a subset of points whose center is furthest from the origin.

To find an appropriate matrix C, the authors of [12] invoke a theorem from principal components analysis [152], that the best rank-d approximation to A with respect to both the spectral and Frobenius norms is $U^d \Delta_d U_d^T$. (Following [12], we use the eigenvectors of A so the columns of U_d are $\vec{\mu_1}, \ldots, \vec{\mu_d}$ with eigenvalues $\lambda_1 \geq \ldots \geq \lambda_n$.) This result implies that $C = U_d \Delta^{\frac{1}{2}}$ is the closest approximation to $A = CC^T$, and thus C is "equivalent" to V_d, meaning the reductions of [68] and [12] are also equivalent. Hence, Arun and Rao choose $C = U_d \Delta^{\frac{1}{2}}$. Like Barnes [18], they solve the 1-dimensional geometric clustering problem by sorting the entries of $\vec{\mu_1}$, and the 2-dimensional problem by testing all possible hyperplanes which divide the set of points into two clusters. In [13], they extend their 2-dimensional hyperplane algorithm to d dimensions, finding the optimal vector partitioning in $O(n^{d(d+3)/2})$ time. Frankle [67] has shown how to exhaustively search a d-dimensional vector space with probes, finding the optimal solution in $O(n^{d-1})$ time.

The min-cut bipartitioning reductions of [68] and [12] can be extended to many multi-way formulations, including minimum Scaled Cost, minimum total net cut, or maximum cluster I/O ($|E(C_h)|$ in FPGA partitioning), via a *vector partitioning* formulation [6]. Let S be the set of n vectors

$\{\vec{y_1}, \vec{y_2}, \ldots, \vec{y_n}\}$ which form the n rows of V_d. The authors of [6] showed that min-cut k-way partitioning reduces to the following *vector partitioning* problem: Find k mutually disjoint subsets of vectors $\{S_1, S_2, \ldots, S_k\}$, with $S_1 \cup S_2 \cup \ldots \cup S_k = S$, so as to maximize

$$\sum_{h=1}^{k} ||\vec{Y_h}||^2 \quad \text{where} \quad \vec{Y_h} = \sum_{\vec{y_i} \in S_h} \vec{y_i}$$

If $d = n$ and $\{S_1, S_2, \ldots, S_k\}$ is the optimum vector partitioning solution, then $\{C_1, C_2, \ldots, C_k\}$ is the optimum vector partitioning solution where $C_h = \{v_i \mid \vec{y_i} \in S_h\}$.

Alpert and Yao propose the greedy MELO algorithm, which does not explicitly construct a vector partitioning, but instead generates a 1-dimensional ordering that is split into a clustering using dynamic programming (see Section 4.5). Given a d-dimensional vector partitioning instance, MELO starts with a set of vectors $S_1 = \emptyset$ and iteratively adds to S_1 the vector $\vec{y_i} \in S - S_1$ that maximizes

$$||\vec{y_i} + \sum_{\vec{y_j} \in S_1} \vec{y_j}||$$

If $\vec{y_i}$ is the j^{th} vector added to S_1 then v_i is the j^{th} module in the linear ordering. MELO can also be interpreted strictly in terms of equation (4.6). Recall that the indicator $\vec{x}$ should maximally project onto the d eigenvectors with smallest eigenvalues. Starting with $\vec{x} = \vec{0}$, MELO iteratively sets $x_i = 1$, for that i which maximizes $\sum_{j=1}^{d} \alpha_j(\vec{x})^2 (H - \lambda_j)$, until $\vec{x} = \vec{1}$.

4.5 From Orderings to Partitionings

We have seen that spectral methods can be used to construct various geometric representations of the netlist, including multi-dimensional embeddings and 1-dimensional orderings. The latter type of representation has led to several intuitive and efficient heuristics (e.g., we have discussed constructing a bipartitioning from a 1-dimensional placement by sorting coordinates to obtain a linear ordering of modules, and then splitting this ordering into two clusters), and has received increased attention in recent years. We conclude this section by surveying various methods which construct partitionings from 1-dimensional orderings.

4.5.1 The Net-Based Approach

If a vertex ordering is constructed for the intersection graph or dual graph, then we will actually obtain an ordering of signal nets, rather than of modules. An approach which then splits the net ordering will require an additional step to construct a module partitioning from the resulting net partitioning. In other words, if we are given, say, a bipartition of signal nets, some modules will belong only to nets on one side of the partition and can be unambiguously assigned to that side, but other modules will be *shared* by nets on both sides of the partition (cf. terms such as "boundary graph" [112] [83] or "module contention" [49]). We must therefore seek a *completion* of the net partition which assigns each shared module to a single cluster, such that the partition cost is minimized. This general net-based partitioning approach (i.e., first obtain a net partition, then complete the net partition into a module partition) was established in [83] and extended in, e.g., [48] and [49] (see Section 5.2).

If the nets are partitioned into two sets N_1 and N_2, a module bipartitioning $P^2 = \{C_1, C_2\}$ can be derived by having the nets "vote" to determine each module assignment. The IG-Vote algorithm of Hagen and Kahng [83] begins with all nets in N_1 and all modules in C_1. Iteratively, a net is moved from N_1 to N_2 according to the net ordering. After each net is moved, if any incident module in C_1 has stronger connections to nets in N_2 than to nets in N_1, then this module is moved to C_2. This process generates at most $n-1$ module bipartitionings; the one with smallest ratio cut is chosen. The variant approach of [48] also considers each split of the net ordering, and for each completes the module bipartitioning by using a matching-based approach (see Section 5.2).

4.5.2 Dynamic Programming for Restricted Partitioning

To split a linear ordering into more than two clusters, Alpert and Kahng [5] proposed the DP-RP ("dynamic programming for restricted partitioning") algorithm, which optimally solves the restricted formulation where each cluster of P^k must be contiguous in the linear ordering. Notice that there are at most n^2 contiguous subsets of the ordering, and these represent every possible cluster that can belong to P^k. DP-RP begins by computing the costs for each of these possible clusters. These correspond to the optimal 1-way partitioning solutions for all contiguous subsets of the ordering. Dynamic programming is then applied to find all the optimal 2-way solutions, then all the optimal 3-way solutions, etc. DP-RP is optimal for any partitioning objective that is a *monotone* function of some intercluster cost metric. Such objectives include standard k-Way Minimum Cut, Scaled Cost, Absorption, and the DS metric. Although the complexity of DP-RP depends on the objective function, $O(nU + kn(U - L))$ implementations exist for all of these objectives except DS, with L and U denoting lower and upper bounds on cluster size.

The DP-RP result highlights the problem of finding 1-dimensional netlist representations which will lead to good multi-way partitioning solutions. Alpert and Yao [6] have applied DP-RP to their (MELO) orderings (see also the discussions of [4] [100] [101] in Section 6). Note that Frankle and Karp's [68] probe technique can be used to yield linear orderings as well, simply by sorting the entries of $V_d\vec{y}$ for a given probe direction $\vec{y}$; we believe that applying DP-RP to these orderings is a promising direction for future research.

4.5.3 Placement-Based Approaches

Another method which constructs a bipartition from a linear ordering is due to Riess et al. [156]. The premise of their PARABOLI approach is that the *squared* wirelength objective is not as useful as a *linear* wirelength objective for 1-dimensional placement; this insight derives from experiments with the GORDIAN placement package [120]. The authors of [156] begin with the 1-dimensional placement induced by $\vec{\mu_2}$; the ten modules with largest coordinates are fixed at location 1.0, and the ten modules with smallest coordinates are fixed at location 0.0. The remaining "free" modules are constrained to have center of gravity (i.e., mean coordinate) at 0.5; quadratic programming techniques are used to reposition these modules to minimize linear wirelength (see Section 5.3.4). The 5% of modules with largest coordinates are assigned center of gravity 0.95, and the 5% with smallest coordinates are assigned center of gravity 0.05. The free modules are again replaced, and the next 5% at the extreme right (left) are assigned center of gravity 0.90 (0.10). Note that throughout this process, only the original 20 extreme modules are fixed, with all others free to move but restricted by their centers of gravity. This process is repeated ten times (e.g., the next iteration uses centers of gravity 0.85 and 0.15), inducing ten distinct linear orderings. Riess et al. choose the best ratio cut bipartitioning among all possible splits of all ten orderings. The intuition behind PARABOLI is that after the first iteration, the leftmost and rightmost modules clearly belong at opposite ends of the linear ordering, but the proper locations for the middle modules remains unclear. Iteratively constraining only small fractions of the modules to the extremes of the linear ordering makes placement of the inner modules easier, while retaining flexibility. Riess and Schoene [158] have extended this approach to a layout-driven formulation in MCM partitioning, optimizing a given multi-chip layout alternately in vertical and horizontal directions.

We believe the success of the PARABOLI approach may have just as much to do with its module assignment technique as its linear wirelength objective. Thus, an interesting experiment would evaluate the quality of linear orderings derived using the same methodology, but with a quadratic objective. The successive over-relaxation placement approach of Tsay and Kuh [183] can also be used for this type of quadratic optimization with fixed module locations (however, centers of gravity cannot be used); see Section 5.3.4.

5 Combinatorial Formulations

The engineering workstations in a typical modern design environment have processing capabilities exceeding those of mainframes from only a few years ago, thereby permitting previously infeasible approaches to be applied to CAD optimizations. For example, spectral computations proposed in