Computing Betti Numbers via Combinatorial Laplacians

Joel Friedman
Department of Mathematics
1984 Mathematics Road
University of British Colubmia
Vancouver, BC V6T 2G3
CANADA
jf@math.ubc.ca

November 30, 1995

Abstract

We use the Laplacian and power method to compute Betti numbers of simplicial complexes. This has a number of advantages over other methods, both in theory and in practice. It requires small storage space in many cases. It seems to run quickly in practice, but its running time depends on a ratio, ν, of eigenvalues which we have yet to fully understand.

We numerically verify a conjecture of Björner, Lovász, Vrećica, and Živaljević on the chessboard complexes $C(4,6)$, $C(5,7)$, and $C(5,8)$. Our verification suffers a technical weakness, which can be overcome in various ways; we do so for $C(4,6)$ and $C(5,8)$, giving a completely rigourous (computer) proof of the conjecture in these two cases. This brings up an interesting question in recovering an integral basis from a real basis of vectors.

1 Introduction

A fundamental and important set of invariants of a topological space, X, is its collection of homology groups, $H_i(X) = H_i(X, \mathbf{Z})$. Computing these groups is of great importance, especially in topology and pure mathematics. However, there are numerous applied areas where computing homology is of interest. These include pattern recognition and classification in biology, chemistry, robotics, and scene analysis, involving low dimensional topological spaces, as well as time series analysis and dynamical systems, involving higher dimensional spaces (see [Cha95]).

A part of the homology groups are the Betti numbers, the i-th Betti number, $b_i = b_i(X)$, being the rank of $H_i(X)$. The Betti numbers often have intuitive meanings. For example, b_0 is simply the number of connected components of the space. As another example, the oriented two-dimensional manifolds are completely classified by $b_1 = 2g$, where g is the genus (i.e. number of "handles") of the two-manifold.

When one wants to know the homology groups, sometimes it suffices to know the Betti numbers. First, if one is trying to distinguish two objects via their homology, their Betti numbers may already distinguish them; similarly, to prove the nonvanishing of a homology group it suffices (but is not necessary) to have the corresponding Betti number vanish. Second, there are certain situations where one *a priori* knows that there is no torsion in the homology groups; then homology is completely described by the Betti numbers. Moreover, it is often much simpler to compute the Betti numbers, and so one does this as a first step in determining the homology groups.

A *simplicial complex* is a collection of subsets (called faces) of a fixed set, S, which is closed under taking subsets. It gives rise to a certain topological space, and the homology of any "reasonable" topological space can be computed from those of an "approximating" simplicial complex (see [Mun84]). In this paper we restrict ourselves to input data being a simplicial complex (actually it suffices to list the maximal faces). In applications one is often given a topological space directly in this form.

Computing homology at present seems hard; currently algorithms require computing the Smith normal form of the matrix. The latter can be done in polynomial time (see [KB79]), but the current polynomial required is still quite large in degree[1]. Computing Betti numbers, on the other hand, seems much easier. To find b_i, one only need compute the ranks of two matrices. This gives what might be called the "standard algorithm," requiring $S = O(Tn_i(n_{i-1} + n_{i+1}))$ storage and $O(S \max(n_i, n_{i-1} + n_{i+1}) \log T)$ operations, where n_i is the number of i-faces (i.e. faces of dimension i, i.e. subsets of size $i + 1$), and where T is the number of storage registers needed to store a matrix entry during the rank calculation; unless the model allows storage registers of arbitrary size, T would need to be $O(\min(n_i, n_{i-1} + n_{i+1}))$. This points to a practical problem: when the n_j are large an exact computation of the rank may be unreliable unless a multiple precision package is used; thus the operations become very expensive. It becomes important to have techniques for computing the Betti numbers which are faster and require less storage.

As a step in this direction, [DE93] give an algorithm to compute Betti numbers; it works in almost linear time in the n_i, but only works for subtriangulations of $\mathbf{R}^3$. Their method is based on a standard type of topological induction, and for more general simplicial complexes it is not clear how to make this method fast.

In this paper we exploit the combinatorial Laplacian to give a simple algorithm to compute the Betti numbers of a simplicial complex. It gives an algorithm which given

[1] In [DC91] it is claimed that this can be done faster for sparse matrices, but the latest version of this paper still has serious flaws.

$\epsilon, \delta > 0$ computes b_i with probability $\geq 1-\delta$, requiring $S = O(n_i(d_i + b_i))$ storage and $O(S(b_i+1)r) + O(n_i(n_0 - i)(i+1)\log N)$ operations; here $N = \max(n_{i-1}, n_i, n_{i+1})$, d_i is the average degree of the i-th Laplacian (always $\leq 1 + (i+1)(n_0 - i - 1)$), and $r = O(\log(n_i b_i \delta^{-1}\epsilon^{-1})/\nu)$, where $\nu = \lambda_1/\lambda_{\max}$ is a ratio of Laplacian eigenvalues discussed later; here we are assuming that ϵ is chosen so that $2\epsilon b_i < \nu$. We do not need to know ν or b_i before we choose ϵ, but we do need to choose ϵ so that we are confident that $2\epsilon b_i < \nu$ holds; in practice this doesn't seem to be a problem. It is an important question to determine what ν is, in various situations. In examples of interest here, $1/\nu$ is always ≤ 25 while n_i gets as high as around 10,000. We know that for $i = 0, 1$, ν can be as small as roughly $1/n_i^2$; we also know that for $i = 0$ we have $\nu \geq 1/2$ with high probability in certain random settings.

After running the main algorithm, one can try to verify rigourously (not merely with high probability and not requiring assumptions on ν) that b_i is at least the computed value for b_i by an "integralizing" process. This requires no more storage and less time than does the main algorithm. However, this does not always work in single or double precision, and brings up an interesting question about the ability to "integralize."

We use our algorithm to compute the Betti numbers of a number of complexes, especially the "chessboard complexes" (see [BLVŽ94]); these complexes arise in several contexts, including some combinatorial geometric problems where there homology groups are of interest (see [ŽV92, ABFK92]). We use our algorithm to verify a conjecture of [BLVŽ94] involving three of the chessboard complexes. In two cases we are able to "integralize" and give rigourous demonstrations of the conjecture in these cases. We also notice that all eigenvalues of the Laplacians are integers for all the complexes we checked; we conjecture this is always true[2].

We also mention that with our algorithm it is much easier to verify (with high probability) that a Betti number vanishes, rather than to show that it doesn't and to evaluate it. But indeed, for a large class of combinatorial complexes (e.g. those which are shellable) only one Betti number other than $b_0 = 1$ does not vanish; for these complexes we can compute all the Betti numbers (using the known Euler characteristic) as soon as we can detect whether or not a Betti number vanishes. This method was used in [Fri92] to help successfully calculate the Betti numbers of a large family of complexes.

We finish by summarizing the rest of this paper. In section 2 we describe the combinatorial Laplacians by way of Hodge theory. In section 3 we analyze some algorithms for determining the largest eigenvalue of a symmetric matrix and the eigenvalue's multiplicity. In section 4 we prove theorems about our Betti number computing algorithms. In section 5 we give two complexes whose Betti numbers we compute via our algorithm, and we give some computational data about our numerical experiments. In section 6 we describe our findings on the chessboard complexes. In section 7 we describe some provable and some conjectural lower bounds for λ_1 and $\lambda_1/\lambda_{\max}$ of the i-th Laplacian.

[2] This has lead to the recent work [FH], where this and part of the [BLVŽ94] conjecture is proven.

2 Hodge Theory and the Laplacian

Our main tool is to use the combinatorial Laplacians (see [Hod41, Eck45, Dod76, DP76]) to compute the Betti numbers. These Laplacians are most easily described via Hodge theory of Hodge [Hod41].

Recall that the Betti numbers, b_i, are the dimensions of the homology groups, $H_i = \ker(\partial_i)/\mathrm{im}(\partial_{i+1})$ of the chain complex,

$$\cdots \longrightarrow \mathcal{C}_{i+1} \xrightarrow{\partial_{i+1}} \mathcal{C}_i \xrightarrow{\partial_i} \mathcal{C}_{i-1} \longrightarrow \cdots \longrightarrow \mathcal{C}_{-1} = 0, \tag{2.1}$$

where $\mathcal{C}_i$ is the space of formal $\mathbf{R}$-linear sums of oriented i-dimensional faces, i.e. oriented subsets of the abstract simplicial complex of size $i+1$, and ∂_i is the boundary map (see [Mun84]).

Concretely, for each face, A, we fix an ordering of its vertices, $\mathcal{A}$. ∂_i is an $n_{i-1} \times n_i$ matrix indexed on the $(i-1)$- and i-faces as follows: if A, B are i- and $(i-1)$-faces respectively, with orderings $\mathcal{A}, \mathcal{B}$, then ∂_i at B, A is zero unless $B \subset A$; if $B \subset A$, then there is a unique permutation π on $\mathcal{A}$ such that $\mathcal{B}$ is $\pi(\mathcal{A})$ minus its first element (in order), and ∂_i at B, A is the sign of π.

So the "standard algorithm" to compute b_i referred to in the introduction is simply to determine the ranks of ∂_i and ∂_{i-1}.

Hodge theory works for an arbitrary chain complex over $\mathbf{R}$ (or any field of characteristic 0, such as $\mathbf{Q}$ or $\mathbf{C}$). Recall that a chain complex is a collection, $\mathcal{C}_i$, of vector spaces, with maps $\partial_i : \mathcal{C}_i \to \mathcal{C}_{i-1}$, as in equation 2.1, such that $\partial_{i-1} \circ \partial_i = 0$ for all i. Endowing each $\mathcal{C}_i$ with an inner product, we get maps $\partial_i^* : \mathcal{C}_{i-1} \to \mathcal{C}_i$ (i.e. the transpose of ∂_i), and thus a Laplacian (as in [Dod76]), $\Delta_i : \mathcal{C}_i \to \mathcal{C}_i$, for each i, defined by

$$\Delta_i = \partial_{i+1}\partial_{i+1}^* + \partial_i^*\partial_i.$$

For each i we define the set of *harmonic i-forms* to be

$$\mathcal{H}_i = \{c \in \mathcal{C}_i | \Delta_i c = 0\}.$$

For chain complexes where each $\mathcal{C}_i$ a finite dimensional $\mathbf{R}$-vector space, Hodge theory involves only elementary linear algebra, and says:

Proposition 2.1 (Hodge theory) *For each i we have $\mathcal{H}_i \cong H_i$, in that each member of $\mathcal{H}_i$ gives rise to a class in H_i, and each class in H_i contains a unique harmonic form in $\mathcal{H}_i$.*

In more detail, for each i we have that $\mathcal{C}_i$ decomposes into orthogonal subspaces

$$\mathcal{C}_i = \mathcal{H}_i \oplus im(\partial_{i+1}) \oplus im(\partial_i^*).$$

The Laplacian is positive definite on the latter two summands and is invariant on each. $\partial_{i+1}\partial_{i+1}^*$ is invariant on the middle summand and vanishes on the other two, and similarly for $\partial_i^*\partial_i$.

Proof Follows easily from the facts that (1) $A = \partial_i^* \partial_i$ and $B = \partial_{i+1} \partial_{i+1}^*$ are positive semi-definite and commute, satisfying $AB = BA = 0$, and (2) $\mathrm{im} S = \mathrm{im} S \circ S^*$ for any map of finite inner product spaces, $S: V \to W$.

□

We may therefore calculate the Betti numbers as the dimension of the $\mathcal{H}_i$. The algorithm we will give works better the larger the eigenvalues of Δ_i are on $\mathrm{im}(\partial_{i+1}) \oplus \mathrm{im}(\partial_i^*)$; this space is just $\mathcal{H}_i^\perp$, by Hodge theory. One easy observation which we will use is:

Proposition 2.2 *The set of (non-zero) eigenvalues of Δ_i on $\mathcal{H}_i^\perp$ is a subset of those on $\mathcal{H}_{i-1}^\perp$ and $\mathcal{H}_{i+1}^\perp$.*

Proof This follows immediately from the fact that for any matrix, A, we have that AA^* and A^*A have the same set of non-zero eigenvalues.

□

3 The Power Method

In this section we discuss the usual power method for finding the largest eigenvalue. We will give an analysis of certain aspects of it which we will need.

Let A be a symmetric matrix, with eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n \geq 0$ and an orthonormal set of eigenvectors, u_i, $i = 1, \ldots, n$, with $Au_i = \lambda_i u_i$. (At the end of this section we weaken the restriction $\lambda_n \geq 0$.) We wish to find λ_1, and perhaps its multiplicity and a basis for its eigenspace.

Our version of the power method finds λ_1 as follows. We choose a random vector, v, by taking its coordinates to be iid normal with mean 0 and variance 1. We set $v_r = A^r v$, and consider

$$\mathcal{T}(v_r) = (Av_r, Av_r)/(v_r, v_r) = (v_{r+1}, v_{r+1}),$$

where (.) denotes the inner product. $\mathcal{T}(v_r)$ should converge to λ_1^2, and accordingly v_r should approach (when suitably normalized) an eigenvector. We take r large enough (see below) so that $\mathcal{T}(v_r)$ is close to λ_1^2. If we only wish to find λ_1, we are done.

If we wish to determine the multiplicity of λ_1, we set $u_1 = v_r/\|v_r\|$, and apply the same power method to $\tilde{A} = PAP$ where P is the projection onto the orthogonal complement of v_1. The power method gives us a sequence v_i, each orthogonal to u_1, with $\mathcal{T}(v_r)$ approaching λ_2^2. Again, for large r we see if $\mathcal{T}(v_r)$ is close (see below) to our computed value of λ_1^2. If not we are done; if so we start a third sequence of iterations, on $\tilde{A} = PAP$ with P projecting out u_1, u_2. Continuing in this way we will find orthonormal $u_1, \ldots, u_m$ with each $\mathcal{T}(u_i)$ close to λ_1^2, and the next set of iterations giving v_i with $\mathcal{T}(v_r)$ not close to λ_1^2; we conclude λ_1 has multiplicity m.

We now analyze the algorithm.

Let $p_i = (v, u_i)$. The p_i are therefore iid $N(0,1)$. The quantity

$$L = (p_2^2 + \cdots + p_n^2)/p_1^2$$

will be crutial to analyzing the convergence of the power method; for example, if $\lambda_2 \neq \lambda_1$ then $L < \infty$ iff the power method converges. It is not hard to estimate:

Lemma 3.1 *For any $\alpha > 0$, the probability that $L \geq n\alpha$ is at most $\sqrt{2/(\pi\alpha)}$.*

Proof See [Fri95].

L's importance can be seen from the next lemma.

Lemma 3.2 *Let $\mathcal{T}(v_r) \leq \lambda_1^2(1-\eta)$ for some r and $\eta > 0$. Then $L \geq (1-\eta)^{-r}$. Also, if λ_1 has multiplicity m and $\lambda_{m+1}/\lambda_1 = \mu$, then $L \geq \eta\mu^{-2r}$.*

Proof We will easily reduce this to the following lemma:

Lemma 3.3 *If $q_2, \ldots, q_n$ are reals with $\sum q_i^2 \leq K$, then for any $\lambda_2, \ldots, \lambda_n \in \mathbf{R}$ we have:*

$$\frac{1 + \sum_{i>1} \lambda_i^{2r+2} q_i^2}{1 + \sum_{i>1} \lambda_i^{2r} q_i^2} \leq 1 - \eta$$

implies that $K(1-\eta)^r \geq 1$. Also, if $\lambda_i \leq \mu$ whenever $\lambda_i \neq 1$, then $\eta \leq K\mu^{2r}$.

Proof Call the left-hand-side of the above equation f, viewed as a function of the λ_i and the q_i.

First we claim that f is minimized when all the λ_i are equal. Viewing the q_i as fixed, it is clear that f achieves its minimum for the λ_i ranging over a compact set, such as $[-1,1]^n$. By differentiating $g(\lambda) = (A + q^2\lambda^{2r+2})/(B + q^2\lambda^{2r})$ one sees that g is minimized when $\lambda = (r/(r+1))g(\lambda)$. So at the minimum we have $\lambda_i = (r/(r+1))f_{\mathsf{min}}$, and so all λ_i are equal there.

So at its minimum value,

$$f = \frac{1 + Q\lambda^{2r+2}}{1 + Q\lambda^{2r}} = 1 - \frac{\lambda^{2r} - \lambda^{2r+2}}{(1/Q) + \lambda^{2r}}$$

where $Q = \sum q_i^2$, which is clearly minimized when Q is as large as possible, namely K. So it remains to consider the minimum of:

$$f(\lambda) = \frac{1 + K\lambda^{2r+2}}{1 + K\lambda^{2r}}.$$

Now if $f(\lambda) \leq 1 - \eta$ then we have $\lambda^2 \leq 1 - \eta$, since

$$f(\lambda) = \left(\frac{1}{1 + K\lambda^{2r}}\right) + \left(\frac{K\lambda^{2r}}{1 + K\lambda^{2r}}\right)\lambda^2 \geq \min(1, \lambda^2).$$

Since $\lambda^2 \leq 1 - \eta$, we have

$$f(\lambda) = 1 - (1 - \lambda^2)\left(\frac{K\lambda^{2r}}{1 + K\lambda^{2r}}\right) \geq 1 - (1 - \lambda^2)K\lambda^{2r} \geq 1 - \eta K(1 - \eta)^r,$$

so that $f(\lambda) \leq 1 - \eta$ implies $K(1 - \eta)^r \geq 1$.

Finally, in the case $\lambda_i \leq \mu$ whenever $\lambda_i \neq 1$, a similar argument shows that f's minimum is achieved when all $\lambda_i \neq 1$ are equal, say equal λ, and when Q is as large as possible. Thus $f(\{\lambda_i\}, \{q_i\})$ is lower bounded by

$$f(\lambda) = \frac{1 + K\lambda^{2r+2}}{1 + K\lambda^{2r}} = 1 - \frac{(1 - \lambda^2)K\lambda^{2r}}{1 + K\lambda^{2r}} \geq 1 - K\lambda^{2r} \geq 1 - K\mu^{2r}.$$

So $f(\lambda) \leq 1 - \eta$ implies $K\mu^{2r} \geq \eta$.

□

To finish the proof of the former lemma, observe that

$$\frac{\|Av_r\|^2}{\|v_r\|^2} = \frac{\lambda_1^{2r+2}p_1^2 + \sum_{i>1}\lambda_i^{2r+2}p_i^2}{\lambda_1^{2r+2}p_1^2 + \sum_{i>1}\lambda_i^{2r}p_i^2} = \frac{1 + \sum_{i>1}(\lambda_i/\lambda_1)^{2r+2}(p_i/p_1)^2}{1 + \sum_{i>1}(\lambda_i/\lambda_1)^{2r}(p_i/p_1)^2},$$

and observe that if $q_i = p_i/p_1$ that $\sum_{i>1} q_i^2 = L$.

□

Combining the first two lemmas of this section easily yields:

Theorem 3.4 *The probability that $\mathcal{T}(v_r) \leq (1 - \eta)\lambda_1^2$ is at most*

$$(1 - \eta)^{r/2}\sqrt{2n/\pi}.$$

This probability is less than δ provided that $r \geq \log(n/\delta^2)/\eta$.

The above theorem tells us how soon the Rayleigh quotient $\mathcal{T}(v_r)$ approaches λ_1^2. To find the multiplicity of λ_1, we need the following result.

Let $u_1, \ldots, u_k$ be k be orthonormal vectors such that $\mathcal{T}(u_i) \geq (\lambda_1 - \epsilon)^2$ for some $\epsilon > 0$. Let P be the projection onto the subspace orthogonal to the u_i, and let $\tilde{A} = PAP$.

Theorem 3.5 *If A has eigenvalue λ_1 with multiplicity m, then $\tilde{A}$ has eigenvalue λ_1 with multiplicity $m-k$. The second eigenvalue, μ_2, of $\tilde{A}$ saisfies $\lambda_{m+1} \leq \mu_2 \leq \lambda_{m+1}+\epsilon k$.*

Proof This is standard and easy. If $w_1, \dots, w_{m-k}$ are orthonormal, orthogonal to the u_i, and in the eigenspace of A with respect to λ_1, then each w_i is clearly an eigenvector of $\tilde{A}$ with eigenvalue λ_1. This $\tilde{A}$ has eigenvalue λ_1 with multiplicity at least $m - k$. If v is a unit vector maximizing $\mathcal{R}(v) = (\tilde{A}v, v)/(v, v)$ subject to being orthogonal to the w_i, then clearly $Pv = v$ (or else $\mathcal{R}(Pv) < \mathcal{R}(v)$); hence $\mathcal{R}(v) = (Av, v)$, which is the next largest eigenvalue of $\tilde{A}$. An easy calculation shows that for any symmetric matrix, B, and orthonormal $s_1, \dots, s_r$ we have

$$\sum_{i=1}^{r}(Bs_i, s_i) \leq \sum_{i=1}^{r} \rho_i,$$

where ρ_i is the i-th largest eigenvalue of B. In particular we have

$$m\lambda_1 + \lambda_{m+1} \geq \sum(Aw_i, w_i) + \sum(Au_i, u_i) \ + \ (Av, v) \geq k(\lambda_1 - \epsilon) + (m-k)\lambda_1 + (\tilde{A}v, v),$$

which proves $\mu_2 \leq \lambda_{m+1} + \epsilon k$. But clearly we can find a v in the span of the λ_1 and λ_{m+1} eigenvectors of A orthogonal to the w_i and u_i, and for such a v we have $(Av, v) \geq \lambda_{m+1}$; also for such a v we have $Pv = v$, which shows that $\mu_2 \geq \lambda_{m+1}$.

□

Theorem 3.6 *Let A have eigenvalue λ_1 with multiplicity m. Fix an $\epsilon > 0$ and set $\mu = m\epsilon + (\lambda_{m+1}/\lambda_1)$. Then in m applications of the power method of r iterations each, we will find orthonormal vectors $w_1, \dots, w_m$ vectors with $\mathcal{T}(w_i) \geq \lambda_1^2(1-\epsilon)^2$ with probability at least $1 - m\mu^r\sqrt{2n/(\pi\eta)}$, where $\eta = 2\epsilon - \epsilon^2$. This probability is $\geq 1 - \delta$ if*

$$r \geq (1/2)\log\left(nm^2/(\eta\delta^2)\right)/(1-\mu).$$

Remark 3.7 Clearly all the theorems and lemmas in this section hold for arbitrary symmetric matrices A, provided that (1) we label the eigenvectors in increasing absolute value, $|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_n|$, and (2) all statements made about a λ_i are replaced with $|\lambda_i|$.

4 Algorithms and Theorems

In this section we use the power method to compute the multiplicity of the eigenvalue 0 of Δ_i, thereby computing b_i. The algorithm actually finds the number of eigenvalues of Δ_i less than a prescribed parameter; to verify that all these eigenvalues are 0, we can try an "integralization algorithm" for a rigourous verification (we can also use a lower bound on λ_1 the first non-zero eigenvalue of Δ_i, if such a bound is available). We make some further remarks about computing other eigenvalues of Δ_i as well.

First we must construct Δ_i; let d_i denote its average degree (i.e. average number of nonzero entries per row). Let X be a simplicial complex, and let X_i be the i-faces. We assume that each X_i is specified as a binary word of length n_0 in the obvious way (from some ordering of X_0).

Proposition 4.1 *The matrix Δ_i can be obtained from a list of X_i and X_{i+1} in space $O(n_i d_i)$ and time $T = O(n_i(n_0 - i)(i+1)\log N)$ where $N = \max(n_{i-1}, n_i, n_{i+1})$.*

Proof We begin by sorting the X_i and X_{i+1}. Since clearly $n_{i+1} \leq (n_0 - i)n_i$ (considering the map from X_{i+1} to X_i of deleting the smallest vertex), the sorting takes time $O(n_i(n_0 - i)\log N)$.

For $A, B \in X_i$, consider the Δ_i entry corresponding to A, B. If $A = B$, this entry is equal to the number of $C \in X_{i-1}$ such that $C \subset A$ plus the number of $C \in X_{i+1}$ such that $A \subset C$; one can count this number in $n_0(\log(\max(n_{i-1}, n_{i+1})))$. If $A \neq B$, then this entry is zero if B does not differ from A by one element (i.e. if $|A \cup B| \neq i + 1$); if B differs from A by one element (of which there are $(i+1)(n_0 - i - 1)$ such B's for a fixed A), then the entry is zero if $A \cup B \in X_i$, and is ± 1 otherwise (depending on the position of the elements where the ordered A, B differ). So this entry can be found in time $O(\log(n_{i+1}))$ for each A, B (determining the position of the elements where A, B differ comes at $O(1)$ cost if for fixed A we loop through the possible B appropriately).

□

Let Δ_i's first non-zero eigenvalue be λ_1, and its largest eigenvalue be $\lambda_{\max}$. Fix a $\delta > 0$; our algorithm is to work with probability $\geq 1 - \delta$. To compute b_i, we first approximate $\lambda_{\max}$. We apply the power method to Δ_i; we set $\tilde{B} = \sqrt{\mathcal{T}(v_r)}$ (notation as in the previous chapter), with

$$r \geq \log_{5/9}\Bigl(\delta^2\pi/(2n_i)\Bigr).$$

Then, by theorem 3.4, $\lambda_{\max} \leq 3\tilde{B}/2$ with probability $\geq 1 - (\delta/2)$. Of course, $\tilde{B} \leq \lambda_{\max}$.

If X_1 is empty and X is just a collection of vertices, it is trivial to compute the Betti numbers ($b_0 = n_0$, $b_i = 0$ for $i \geq 1$). If not, then Δ_i contains a positive integer somewhere along its diagonal. Hence $\lambda_{\max} \geq 1$. It follows that if we round $\tilde{B}$ up to the next integer, B, we have $B \leq 2\lambda_{\max}$, and $\lambda_{\max} \leq 3B/2$.

Next set $A = B - \Delta_i$. b_i is the multiplicity of B as an eigenvalue of A. A's smallest eigenvalue, $\mu_{\min}$, is at least $-B/2$; let μ_2 denote A's largest eigenvalue $< B$. Now we apply the power method to A. If $\mathcal{T}(v_r)$ does not seem to converge to B^2, we can stop according to theorem 3.4; i.e. when it tells us that with probability $\geq 1 - (\delta/2)$ A does not have an eigenvalue as large as B.

Theorem 4.2 *Assume $b_i = 0$, and let $\lambda_1 \neq 0$ and $\lambda_{\max}$ be the smallest and largest eigenvalues of Δ_i. Set $\nu = \min(\lambda_1/\lambda_{\max}, 3/4)$. Then in r iterations of the power method applied to A, where $r \geq \log(4n/\delta^2)/\nu$, we have that with probability $\geq 1 - \delta$ we have $b_i = 0$.*

Proof We have

$$\mathcal{T}(v_r) \leq \max(\mu_2^2, \mu_{\min}^2),$$

so that $\mathcal{T}(v_r) \leq (1-\eta)B^2$ where

$$\eta = \min\Big(1 - (\mu_{\min}^2/B^2), 1 - (\mu_2^2/B^2)\Big) \geq \min\Big(3/4, 1 - (B - \lambda_1)^2/B^2\Big)$$

$$= \min\Big(3/4, 2(\lambda_1/B) - (\lambda_1/B)^2\Big) \geq \min\Big(3/4, (3/2)(\lambda_1/B)\Big),$$

since for $x \in [0,1]$ either $x \geq 1/2$, in which case $2x - x^2 \geq 3/4$, or $x < 1/2$, in which case $2x - x^2 = x(2-x) \geq (3/2)x$. Since $\lambda_{\max} \leq 3B/2$ can take $\eta = \nu$ in theorem 3.4, and the theorem follows.

□

Our first main theorem easily follows:

Theorem 4.3 *Assume $b_i = 0$, and let $\lambda_1 > 0$ and $\lambda_{\max}$ be the smallest and largest eigenvalues of Δ_i. Set $\nu = \min(\lambda_1/\lambda_{\max}, 3/4)$. For any prescribed $\delta > 0$ the above algorithm will determine that $b_i = 0$ with probability $\leq 1 - \delta$ in $S = O(n_i d_i)$ space and $O(Sr)$ operations, where $r = O(\log(\delta/n_i)/\nu)$.*

Proof Each iteration requires a multiplication by A, requiring $O(S)$ operations, and an evaluation of $\mathcal{T}$ requiring $O(n_i) = O(S)$ operations.

□

When $b_i \neq 0$, then in performing the power method on A we will (with high probability) see $\mathcal{T}(v_i)$ approaching B. We will pick a point to stop iterating; namely we will fix a small $\epsilon > 0$ and stop iterating when $\mathcal{T}(v_r) \geq B^2(1-\epsilon)^2$. After m applications of the power method we find $u_1, \ldots, u_m$ orthonormal with $\mathcal{T}(u_i) \geq B^2(1-\epsilon)^2$ for each i. We then perform one more application of the power method (to $\tilde{A} = PAP$ with P as in section 3), and have $\mathcal{T}(v_r) \leq (\nu_2 + m\epsilon)^2$ no matter how large r is. Assuming that $\nu_2 + m\epsilon$ is less than $B(1-\epsilon)$ (perhaps much less than, relative to ϵ) we will conclude that $b_i = m$.

Theorem 4.4 *Let $\lambda_1 > 0$ and $\lambda_{\max}$ be the smallest non-zero and largest eigenvalues of Δ_i. Set $\nu = \min(\lambda_1/\lambda_{\max}, 3/4)$. Let $\epsilon, \delta > 0$ be given. Then in $O(n_i(d_i + b_i))$ space and $O(Sb_i r)$ operations, where $r = O(\log(n_i\delta^{-1}\epsilon^{-1})/\alpha)$ with $\alpha = \nu - 3b_i\epsilon/2$, the above algorithm will compute b_i correctly with probability $\geq 1 - \delta$ under the assumption that $\alpha \geq 3\epsilon/2$.*

We remark that we can simplify the theorem by setting $\nu = \lambda_1/\lambda_{\max}$ and insisting that $\nu \geq 2b_i\epsilon$; then r becomes $O(\log(n_i\delta^{-1}\epsilon^{-1})/\nu)$.

Proof Say that $u_1, \ldots, u_m$ with $\mathcal{T}(u_i) \geq B^2(1-\epsilon)^2$ have been found, and we are iterating $\tilde{A} = PAP$. First of all, $n_i m$ storage is required to store the u_i. Then applying P requires $O(mn_i)$ operations, while applying A requires $O(n_i d_i)$ operations.

If we apply the same number, r, iterations to find each u_j, $j = 1, \ldots, b_i$, we require a number of operations proportional to

$$\sum_{j=1}^{b_i} n_i(d_i + j) \leq n_i(d_i + b_i)b_i,$$

and $O(n_i b_i)$ storage (in addition to the $O(n_i d_i)$ required to store Δ_i). Next, by theorem 3.6 with probability $\geq 1 - (\delta/2)$ we need $r = O(\log(nb_i\epsilon^{-1}\delta^{-1})/(1-\mu))$ iterations for each u_j, where

$$\mu = b_i\epsilon + \max(\mu_2, |\mu_{\mathbf{min}}|)/B \leq b_i\epsilon + \max(1 - (\lambda_1/B), 1/2).$$

Hence

$$1 - \mu \geq \min\Big((2/3)(\lambda_1/\lambda_{\mathbf{max}}), 1/2\Big) - b_i\epsilon \geq (2/3)\nu - b_i\epsilon.$$

The final phase of the algorithm, i.e. a applying the power method after having found b_i approximate eigenvectors, requires (by the same analysis as in theorem 4.3) $r = O(\log(n_i\delta^{-1})/\alpha$ iterations to be able to assert that with probability $\geq 1 - \delta$ we have B has multiplicity $\leq b_i$ in A.

□

Next we describe the "integralizing" procedure to rigourously verify that $b_i \geq m$ for some m. Say that our power method algorithm computes that $b_i \geq m$, meaning that it has found orthonormal vectors $u_1, \ldots, u_m$ with $\mathcal{T}(u_j) \geq B^2(1-\epsilon)^2$. To check that the kernel of Δ_i is at least m-dimensional, we proceed as follows. By Gauss-Jordan elimination, and by renumbering the coordinates, we can transform the u_j into a basis v_j where v_j's k-th coordinate is 1 if $j = k$ and 0 if $j \neq k$ and $k \leq n$. If the u_j were a basis for the kernel of Δ_i, then by Kramer's rule all coordinates of the v_j's would be rational numbers. If each v_j has an approximate common denominator, meaning an integer $D_j > 0$ such that $D_j v_j$ is nearly integral, then we can let w_j be the "integralized" (i.e. rounded) $D_j v_j$. The w_j will still be linearly independent; we can check $\Delta_i w_j = 0$ by integer arithmetic, getting a definite answer. If we are able to verify this, then we have a rigourous proof that $b_i \geq m$; of course, our main algorithm tells us that $b_i \leq m$ with probability $\geq 1 - \delta$.

We we shall see in section 6, this integralizing procedure does not always work. The problem is that the approximate common denominators may be large, requiring extra precision in the computation. This leads us to the following question.

Question 4.5 Let $w_1, \ldots, w_m$ be integer-valued vectors in $\mathbf{R}^n$ from the integers $\{-D, \ldots, D\}$, and let S denote their span intersected with the unit sphere. Let $u_1, \ldots, u_m$ be chosen inductively with u_i chosen from S_i uniformly, with S_i the intersection of S with the orthogonal complement of $u_1, \ldots, u_{i-1}$. How much precision (and how much time and space) is required to construct an integral basis for S?

In this question we allow the integral basis to have coefficients larger than D in absolute value if need be.

Finally we describe an algorithm to compute other eigenvalues of Δ_i; we use it in section 6. Namely, to compute the multiplicity of λ in Δ_i, we compute B as before and set $C = R - (\Delta_i - \lambda)^2$, where $R = \max(\lambda^2, ((3/2)B - \lambda)^2)$. (If we perform more iterations to find B, so that B is with high probability very close to $\lambda_m x$, we can replace the 3/2 by 1.) Then the multiplicity of R in C is that of λ in Δ_i; the next largest eigenvalue of C, which controls the convergence rate and therefore speed of the algorithm, is just $R - D^2$, where D is the distance of λ to the nearest other eigenvalue. We do not analyze this algorithm here, but its analysis is similar to that of the above algorithms.

Also, if we expect that Δ_i has, in addition to λ, eigenvalues $\lambda_1, \ldots, \lambda_r$, then we can multiply the initial vector, v_1, in the power method by $(\Delta_i - \lambda_1)\cdots(\Delta_i - \lambda_r)$ (to eliminate the λ_j components of v_1). This greatly sped up this algorithm for the chessboard complexes of section 6.

5 Experiments on $\mathcal{A}_k^{4,2}$

$i =$	$n_i =$	$\lambda_{\max}$	λ_1	$\lambda_1/\lambda_{\max}$	$\delta = 10^{-2}$	$\delta = 10^{-10}$
0	16	16.0000	0.	0.		
1	120	16.0000	16.0000	1.0000	0	0
2	560	16.0000	12.0000	.7500	40	130
3	1796	16.0000	8.0000	.5000	56	172
4	4080	16.0000	4.0362	.2522	95	285
5	6520	16.0000	4.0362	.2522	98	287
6	7104	16.0000	1.9632	.1227	178	522
7	4962	16.0000	0.	0.		
8	1984	16.0000	1.9175	.1198	170	521
9	376	14.0000	3.3316	.2379	87	285
10	16	11.0000	11.0000	1.0000	0	0

Table 1: Data for $\mathcal{A}_0^{4,2}$.

Sometimes we are interested in just verifying that certain Betti numbers vanish. As we mentioned in the introduction, there is a large collection of simplicial complexes for which $b_0 = 1$ and only one other Betti number does not vanish. In this case it suffice to check the non-vanishing of all but one of the b_i's with $i \geq 1$ to determine all the

$i =$	$n_i =$	$\lambda_{\max}$	λ_1	$\lambda_1/\lambda_{\max}$	$\delta = 10^{-2}$	$\delta = 10^{-10}$
0	16	16.0000	0.	0.		
1	120	16.0000	16.0000	1.0000	0	0
2	560	16.0000	16.0000	1.0000	0	0
3	1820	16.0000	16.0000	1.0000	0	0
4	4368	16.0000	8.0000	.5000	59	175
5	7912	16.0000	3.1547	.1971	120	351
6	10560	16.0000	2.9539	.1846	129	372
7	9762	16.0000	1.5637	.09773	221	642
8	5632	16.0000	0.	0.		
9	1672	16.0000	2.3843	.1490	139	430
10	208	15.0000	4.0000	.2666	77	258

Table 2: Data for $\mathcal{A}_1^{4,2}$.

b_i's. The data we present shows that for the complexes studied here, $\nu = \lambda_1/\lambda_{\max}$ is large enough to give a good performance for the algorithm.

We are studying the simplicial complex $\mathcal{A}_k^{4,2}$ of [Fri92] which is defined as follows. We consider all subsets of the four-dimensional Boolean cube, $\mathbf{B}^4$, which do *not* contain more than k 2-dimensional subcubes. So these are simplicial complexes on $n_0 = 16$ vertices.

The data presented is the following. For each i we list n_i, then $\lambda_{\max}(\Delta_i)$, $\lambda_1(\Delta_i)$, and then the ratio of $\lambda_1/\lambda_{\max}$. After that we calculate how many iterations are needed before we can be sure with probability $\geq 1-\delta$ that $b_i = 0$; this number is the ceiling of

$$\log_{9/5} C + \log_{1-\nu} C \qquad \text{where } C = 8n_i\delta^{-2}/\pi.$$

For both $\mathcal{A}_0^{4,2}$ and $\mathcal{A}_1^{4,2}$ there is only one Betti number other than $b_0 = 1$ which does not vanish, and so we can calculate it via the Euler characteristic.

The data for $\mathcal{A}_i^{4,2}$, $i = 0, 1$ shows certain curious features. For example, $\lambda_{\max} = 16$ for a very large number of the Δ_i, and $\lambda_{\max}$ never exceeds 16. The author has no idea why this happens, and is very curious to know if this is a property of the specific complex, or if it reflects some general truth about combinatorial Laplacians.

We also remark that in calculating the λ_1 and $\lambda_{\max}$, we kept iterating until these values converged within double precision; this typically took a few hundred iterations, and never took more than a few thousand.

For practical interest, we also include a short table of the time taken to compute 1000 iterations of the power method applied to $B - \Delta_i$. There we list i, n_i, the sum of the degrees of all vertices of the Laplacian, and the time per 1000 iterations. As expected, for comparable sum of degrees, the Laplacian with n_i smaller iterates faster.

$i =$	$n_i =$	deg sum	time
2	120	120	.13
3	560	560	.65
4	1820	1820	2.25
5	4368	7248	7.41
6	7912	33736	25.68
7	10560	90304	57.83
8	9762	128354	74.69
9	5632	91808	50.96
10	1672	23464	11.45
11	208	2704	1.28

Table 3: CPU Seconds per 1000 iterations.

The times is cpu seconds measured on a DEC 3000/400 workstation.

6 Experiments on Chessboard Complexes

For m, n positive integers, consider an $m \times n$ chessboard, and consider those subsets of the squares such that no two squares lie on the same column or row (i.e. if rooks were placed on the squares, no two rooks could take each other); this collection of subsets, $C(m,n)$, is closed under taking subsets. $C(m,n)$ is the chessboard complex of [Gar79],[BLVŽ94]. however it appears in some combinatorial geometric problems, such as the Colored Tverberg's Problem, as in [ŽV92] (see also [ABFK92] for applications). The k-connectivity of $C(m,n)$ was a key fact in [ŽV92] for certain values of m, n, k; in general, k-connectivity is equivalent to the vanishing of H_i for all $i \leq k$. In [BLVŽ94] the connectivity of $C(m,n)$ was studied; there it was proven that $C(m,n)$ is $v-2$-connected for $v = \min(m, n, \lfloor (m+n+1)/3 \rfloor)$. It was conjectured that $C(m,n)$ is not $v-1$ connected, i.e. that $H_{v-1}(C(m,n)) \neq 0$. This was proven for $n \geq 2m-1$ and all m, n with $m \leq n$ and $m \leq 5$, with the exception of $(m,n) = (4,6), (5,7), (5,8)$. The conjecture holds if $b_{v-1}(C(m,n)) > 0$.

We computed $b_2(C(4,6))$. It has $n_2 = 480$, $d_2 = 13$. We found $\lambda_{\mathsf{max}}(\Delta_2) = 16$. We took $\epsilon = 10^{-10}$ and computed $b_2 = 5$ (and $\lambda_1 = 1$). Upon integralizing we verified that $b_2 \geq 5$, finding a basis of $0, \pm 1$ valued vectors in $\ker(\Delta_2)$; of course, we know that $b_2 < 6$ with high probability from the algorithm, so that $b_2 = 5$ with high probability.

We similarly computed $b_3(C(5,7)) = 98$ ($n_3 = 4200$, $d_3 = 17$) and $b_3(C(5,8)) = 14$ ($n_3 = 8400$, $d_3 = 21$). Integralizing we found a $0, \pm 1$ basis for $\ker(\Delta_3)$ for $C(5,8)$, but we failed to find an integral basis for $C(5,7)$. Thus we rigorously know $b_3 \geq 14$, and know $b_3 = 14$ with high probability for $C(5,8)$; we know $b_3 < 99$ with high probability, and $b_3 = 98$ under the assumption that $\lambda_1 > 10^{-8}$, for $C(5,7)$.

We began to suspect that the eigenvalues of any Δ_i on $C(n,m)$ are all integers, and we verified this for $C(m,n)$ with $m = 3$ and $n = 3,\ldots,7$, with $m = 4$ and $n = 4,5,6$, and with $m = n = 5$. We conjectured this suspicion until its recent proof, in [FH]; there the other calculations made in this section can be verified.

7 Worst Case Eigenvalue Separation

In this section we give a number of arguments which indicate that λ_1 and/or $\nu = \lambda_1/\lambda_{\max}$ of Δ_i, will be bounded away from zero. Most of the results are worst case estimates, and are not very optimistic; we believe that in many applications the true λ_1 and/or ν will be much better than given here (see, for example, section 5 and 6).

A major problem which is left open is to bound λ_1 and ν of Δ_i for all i in terms of the n_i and d_i. This could be done in a worst case, restricted, or probabilistic setting.

7.1 Bounds Based on Random Graphs

Consider Δ_0, which just depends on the 0- and 1-faces of the complex, which is just a graph. It is well known that if X is a random graph on n vertices of degree d, then $\lambda_1(\Delta_0) = d - \sqrt{d}(2 + o(1))$ with high probability in many situations (e.g. $d \geq O(\log^2 n)$ and n large or d fixed and even and n large) (see [Fri91]). It follows that $\nu \geq 1 - d^{-1/2}(2 + o(1))$ for most graphs.

7.2 Bounds Based on Worst Case Graphs

We can get some bounds on Δ_0 just from graph theory bounds. Indeed, from [Fri94], it follows that for a connected graph on n nodes we have

$$\lambda_1(\Delta) \geq 2 - 2\cos(\pi/n) = \pi^2/n^2 + O(n^{-4})$$

where Δ is the graph Laplacian. Since Δ_0 is just a graph Laplacian, it follows that the same bound holds for Δ_0 (and $n = n_0$). This also gives ν proportional to $1/n^2$; we don't know if this is the worst possible ν for a graph on n vertices.

Sometimes a similar bound will hold of Δ_d for a d-dimensional complex (meaning that d is the top dimension of a face). Recall that Δ_d depends on an ordering of each d-face. We say that two faces, A, B, of size d *intersect* if $|A \cap B| = d$ (i.e. Δ_d is non-zero at the A, B entry), and that they intersect *negatively* if A and B have differing signs in the $A \cap B$ row of ∂_d (i.e. Δ_d is -1 at the A, B entry).

Definition 7.1 *We say that a d-dimensional simplicial complex is* orientable *if there is an ordering of the d-faces such that any two intersecting d-faces intersect negatively.*

In particular, a complex based on an *orientable manifold* is orientable. Also, in an orientable complex, each $(d-1)$-face is incident on at most two d-faces.

For a d-dimensional orientable complex we clearly have that Δ_d is a graph Laplacian, where the vertices are the d-faces and the vertices the $(d-1)$-faces. To summarize we have:

Theorem 7.2 *We have that $\lambda_1(\Delta_0) \geq 2 - 2\cos(\pi/n_0)$; for an orientable complex we have $\lambda_1(\Delta_d) \geq 2 - 2\cos(\pi/n_d)$.*

Notice that for $d = 2$, $\lambda_1(\Delta_1)$ is bounded below by the min of that of $\lambda_1(\Delta_0)$ and $\lambda_1(\Delta_2)$. It follows that:

Corollary 7.3 *For an orientable complex of dimension $d \leq 2$ we have that for all i we have $\lambda_1(\Delta_i) \geq 2 - 2\cos(\pi/\tilde{n})$, where $\tilde{n} = \max(n_0, n_d)$.*

7.3 A General Conjecture

Although we can't in general bound $\lambda(\Delta_i)$ for all i when $d \geq 3$, we can give some arguments which suggest how $\lambda_1(\Delta_i)$ changes if we take a fixed simplicial complex and subdivide it more and more finely.

For one thing, it is known (see [Cha84]) that for continuous Laplacians acting on the i-forms of d-dimensional manifolds, we have that the λ_j are infinite and grow like $\lambda_n \approx cn^{2/d}$. It is known that certain types of refinements of combinatorial Laplacians have limits whose eigenvalues converge to those of the continuous Laplacian in a certain sense (see [DP76]). So it makes sense to conjecture that for regular types of refinements (i.e. those where the aspect ratio of sides of a simplex remains bounded with respect to some fixed, smooth metric) we have $\lambda_n \approx cn^{2/d}$.

Conjecture 7.4 *For regular refinements of a given simplicial complex of dimension d we have $\nu = \lambda_1/\lambda_{\max} = \Omega(n_i^{-2/d})$ as $n_i \to \infty$, i.e. as the refinements get finer and finer.*

References

[ABFK92] N. Alon, I. Bárány, Z. Füredi, and D. Kleitman. Point selection and weak ϵ-nets for convex hulls. *Combinatorics, Probability, and Computing*, 1:189–200, 1992.

[BLVŽ94] A. Björner, L. Lovász, S.T. Vrećica, and R.T. Živaljević. Chessboard complexes and matching complexes. *J. London Math. Soc.*, 49:25–39, 1994.

[Cha84] Isaac Chavel. *Eigenvalues in Riemannian Geometry.* Academic Press, 1984.

[Cha95] Bernard Chazelle. Computational geometry: A retrospective. In D.-Z. Du and F. Hwang, editors, *Computing in Euclidean Geometry (2nd Edition)*, pages 22–46. World Scientific Press, 1995.

[DC91] Bruce R. Donald and David R. Chang. On computing the homology type of a triangulation. In *32nd Annual Symposium on Foundations of Computer Science*, pages 650–661, 1991. also preprint (of the full version).

[DE93] C.J.A. Delfinado and H. Edelsbrunner. An incremental algorithm for Betti numbers of simplicial complexes. In *9th Annual ACM Conference on Computational Geometry*, pages 232–239, 1993.

[Dod76] Jozef Dodziuk. Finite-difference approach to the Hodge theory of harmonic forms. *American Journal of Mathematics*, 98:79–104, 1976.

[DP76] J. Dodziuk and V.K. Patodi. Riemannian structures and triangulations of manifolds. *J. of the Indian Math. Soc.*, 40:1–52, 1976.

[Eck45] B. Eckmann. Harmonische Funktionen und Randvertanfgaben in einem Komplex. *Commentarii Math. Helvetici*, 17:240–245, 1944–45.

[FH] J. Friedman and P. Hanlon. On the Betti numbers of chessboard complexes. To appear.

[Fri91] J. Friedman. On the second eigenvalue and random walks in random d-regular graphs. *Combinatorica*, 11:331–362, 1991.

[Fri92] J. Friedman. Probabilistic spaces of Boolean functions of a given complexity: Generalities and random k-sat coefficients. July 1992. Preprint.

[Fri94] J. Friedman. Minimum higher eigenvalues of Laplacians on graphs. October 1994. To appear in *Duke J. Math.*

[Fri95] J. Friedman. Error bounds on the power method for determining the largest eigenvalue. October 1995. Preprint.

[Gar79] P.F. Garst. Some Cohen-Macaulay complexes and group actions. *Ph.D. Thesis*, 1979. University of Madison, Wisconsin.

[Hod41] W.V.D. Hodge. *The Theory and Applications of Harmonic Integrals.* Cambridge University Press, 1941.

[KB79] R. Kannan and A. Bachem. Polynomial algorithms for computing the Smith and Hermite normal forms of an integer matrix. *SIAM J. Computing*, 8:499–507, 1979.

[Mun84] James R. Munkrees. *Elements of Algebraic Topology.* Benjamin/Cummings, 1984.

[ŽV92] R.T. Živaljević and S.T. Vrećica. The colored Tverberg's problem and complexes of injective functions. *J. Combin. Theory Ser. A*, 61:309–318, 1992.