Chapter VII

Persistence

The central concept of this chapter is motivated by the practical need to cope with noise in data. This includes defining, recognizing, and possibly eliminating noise. These are lofty goals and the challenge can be overwhelming. Indeed, the distinction between noise and feature is not well-defined but lies instead in the eye of the beholder. In any particular case, the focus is on a range of scales and a desire to ignore everything that is smaller or larger. In other words, we make ourselves the measure of all things and by doing so derive a unit, a point of view, and an opinion. Motivated by this thought, we take an agnostic approach and offer a means to measure scale, a tool that can be used to make judgments based on quantitative information, if one so desires.

VII.1 Persistent Homology

Persistent homology can be used to measure the scale or resolution of a topological feature. There are two ingredients, one geometric, defining a function on a topological space, and the other algebraic, turning the function into measurements. The measurements make sense only if the function does.

The elder rule. We begin with a simplified scenario in which we develop our intuition. Let $\mathbb{X}$ be a connected topological space and $f : \mathbb{X} \to \mathbb{R}$ a continuous function. The sublevel sets of f form a 1-parameter family of nested subspaces, $\mathbb{X}_a \subseteq \mathbb{X}_b$ whenever $a \leq b$. It is convenient to write about this family as if it were one sublevel set that evolves as the threshold increases. We visualize this evolution by drawing each component of $\mathbb{X}_a$ as a point. The result is a 1-dimensional graph, $G(f)$, not unlike the Reeb graph discussed in the previous chapter. Thinking of f as a height function, we draw the graph from bottom to top. Since components never shrink, the arcs of the graph may merge, but they never split. In the end, for large enough threshold a, we have a single component. It follows that $G(f)$ is a tree, and we refer to it as the *merge tree* of the function; see Figure VII.1.

We decompose this tree into disjoint paths that increase monotonically with f. To obtain the paths, we draw them from bottom to top, simultaneously, while keeping their upper endpoints at the same height, a. Paths extend; however, when they merge, we end the one that started later. Thinking of the difference between two function values as age, we give precedence to the older path.

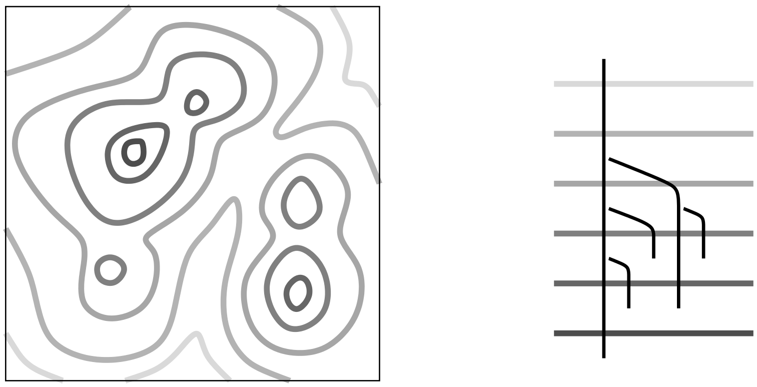

Figure VII.1: Left: a function on the unit square visualized by drawing six level sets with lighter shades of gray indicating larger values. Right: the path decomposition of the merge tree of the function.

Elder Rule. At a juncture, the older of the two merging paths continues and the younger path ends.

Letting $a \leq b$ be two thresholds, we let $\beta(a, b)$ be the number of components in $\mathbb{X}_b$ that have a non-empty intersection with $\mathbb{X}_a$. In terms of the merge tree, this is the number of subtrees with topmost points at value b that reach down to level a or below. Each such subtree has a unique path, its longest, that spans the entire interval between a and b. It follows that $\beta(a, b)$ is also the number of paths in the path decomposition of $G(f)$ that span $[a, b]$. We note that any path decomposition that is not generated using the Elder Rule does not have this property. In particular, if f is Morse, then the Elder Rule generates a unique path decomposition, which is the only one for which the number of paths spanning $[a, b]$ equals $\beta(a, b)$ for all values of $a \leq b$.

Filtrations. We obtain persistence by formulating the Elder Rule for the homology groups of all dimensions. Consider a simplicial complex, K, and a function $f : K \to \mathbb{R}$. We require that f be *monotonic*, by which we mean it is non-decreasing along increasing chains of faces, that is, $f(\sigma) \leq f(\tau)$ whenever σ is a face of τ. Monotonicity implies that the sublevel set, $K(a) = f^{-1}(-\infty, a]$, is a subcomplex of K for every $a \in \mathbb{R}$. Letting m be the number of simplices in K, we get $n+1 \leq m+1$ different subcomplexes, which we arrange as an increasing sequence:

$$\emptyset = K_0 \subseteq K_1 \subseteq \ldots \subseteq K_n = K.$$

In other words, if $a_1 < a_2 < \ldots < a_n$ are the function values of the simplices in K and $a_0 = -\infty$, then $K_i = K(a_i)$ for each i. We call this sequence of complexes the *filtration* of f and think of it as a construction by adding chunks of simplices at a time. We have seen examples before, namely the Čech and the alpha complexes in Chapter III and the lower star filtration of a piecewise linear function in Section VI.3. More than in the sequence of complexes, we are interested in the topological evolution, as expressed by the corresponding sequence of homology groups. For every $i \leq j$ we have an inclusion map from the underlying space of K_i to that of K_j and therefore an induced homomorphism, $f_p^{i,j} : \mathsf{H}_p(K_i) \to \mathsf{H}_p(K_j)$, for each dimension p. The filtration thus corresponds to a sequence of homology groups connected by homomorphisms,

$$0 = \mathsf{H}_p(K_0) \to \mathsf{H}_p(K_1) \to \ldots \to \mathsf{H}_p(K_n) = \mathsf{H}_p(K),$$

again one for each dimension p. As we go from K_{i-1} to K_i, we might gain new homology classes and we might lose some when they become trivial or merge with each other. We collect the classes that are born at or before a given threshold and die after another threshold in groups.

Definition. The *p-th persistent homology groups* are the images of the homomorphisms induced by inclusion, $\mathsf{H}_p^{i,j} = \operatorname{im} f_p^{i,j}$, for $0 \leq i \leq j \leq n$. The corresponding *p-th persistent Betti numbers* are the ranks of these groups, $\beta_p^{i,j} = \operatorname{rank} \mathsf{H}_p^{i,j}$.

Similarly, we define reduced persistent homology groups and reduced persistent Betti numbers. Note that $\mathsf{H}_p^{i,i} = \mathsf{H}_p(K_i)$. The persistent homology groups consist of the homology classes of K_i that are still alive at K_j or, more formally, $\mathsf{H}_p^{i,j} = \mathsf{Z}_p(K_i)/(\mathsf{B}_p(K_j) \cap \mathsf{Z}_p(K_i))$. We have such a group for each dimension p and each index pair $i \leq j$. We can be more concrete about the classes counted by the persistent homology groups. Letting γ be a class in $\mathsf{H}_p(K_i)$, we say it is *born at* K_i if $\gamma \notin \mathsf{H}_p^{i-1,i}$. Furthermore, if γ is born at K_i, then it *dies entering* K_j if it merges with an older class as we go from K_{j-1} to K_j, that is, $f_p^{i,j-1}(\gamma) \notin \mathsf{H}_p^{i-1,j-1}$ but $f_p^{i,j}(\gamma) \in \mathsf{H}_p^{i-1,j}$; see Figure VII.2. This is again the Elder Rule. If γ is born at K_i and dies entering K_j, then we call the difference in function value the *persistence*,

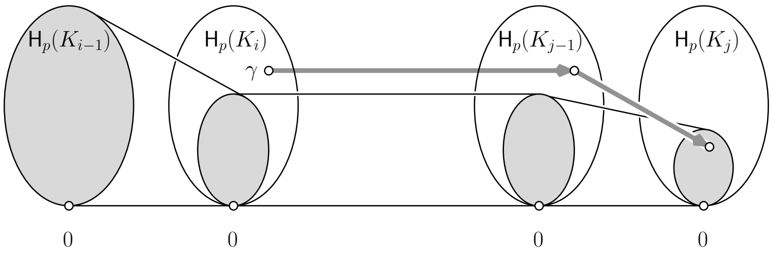

Figure VII.2: The class γ is born at K_i since it does not lie in the (shaded) image of $\mathsf{H}_p(K_{i-1})$. Furthermore, γ dies entering K_j since this is the first time its image merges into the image of $\mathsf{H}_p(K_{i-1})$.

$\mathrm{pers}(\gamma) = a_j - a_i$. Sometimes we prefer to ignore the actual function values and consider the difference in index, $j - i$, which we call the *index persistence* of the class. If γ is born at K_i but never dies, then we set its persistence as well as its index persistence to infinity. We note that births and deaths can also be defined for a sequence of vector spaces that are not necessarily homology groups. All we need is a finite sequence and homomorphisms from left to right, which, for vector spaces, are usually referred to as linear maps.

Persistence diagrams. We visualize the collection of persistent Betti numbers by drawing points in two dimensions. Some of these points may have coordinates equal to infinity, and some might be the same, so we really talk about a multiset of points in the extended real plane, $\bar{\mathbb{R}}^2 = (\mathbb{R} \cup \{\pm\infty\})^2$. Letting $\mu_p^{i,j}$ be the number of independent p-dimensional classes that are born at K_i and die entering K_j, we have

$$\mu_p^{i,j} \quad = \quad (\beta_p^{i,j-1} - \beta_p^{i,j}) - (\beta_p^{i-1,j-1} - \beta_p^{i-1,j}),$$

for all $i < j$ and all p. Indeed, the first difference on the right-hand side counts the classes that are born at or before K_i and die entering K_j, while the second difference counts the classes that are born at or before K_{i-1} and die entering K_j. Drawing each point (a_i, a_j) with multiplicity $\mu_p^{i,j}$, we get the *p-th persistence diagram* of the filtration, denoted as $\mathrm{Dgm}_p(f)$. It represents a class by a point whose vertical distance to the diagonal is the persistence. Since the multiplicities are defined only for $i < j$, all points lie above the diagonal. For technical reasons which will become clear in the next chapter, we add the points on the diagonal to the diagram, each with infinite multiplicity. Examples of persistence diagrams can be seen in Figure VII.5. It is easy to read off the persistent Betti numbers. Specifically, $\beta_p^{k,l}$ is the number of points in the upper left quadrant with corner point (a_k, a_l). A class that is born at K_i and dies entering K_j is counted iff $a_i \leq a_k$ and $a_j > a_l$. The quadrant is therefore closed along its vertical right side and open along its horizontal lower side.

Fundamental Lemma of Persistent Homology. Let $\emptyset = K_0 \subseteq K_1 \subseteq \ldots \subseteq K_n = K$ be a filtration. For every pair of indices $0 \leq k \leq l \leq n$ and every dimension p, the p-th persistent Betti number is $\beta_p^{k,l} = \sum_{i \leq k} \sum_{j > l} \mu_p^{i,j}$.

This is an important property. It says the diagram encodes all information about persistent homology groups.

Matrix reduction. Besides having a compact description in terms of diagrams, persistence can also be computed efficiently. The particular algorithm we use is a version of matrix reduction. Perhaps surprisingly, we can get all the information with a single reduction. To describe this, we use a *compatible ordering* of the simplices, that is, a sequence $\sigma_1, \sigma_2, \ldots, \sigma_m$ such that $i < j$ if $f(\sigma_i) < f(\sigma_j)$ or if σ_i is a face of σ_j. Such an ordering exists because f is monotonic. Note that every initial subsequence of simplices forms a subcomplex of K. We use this sequence when we set up the m-by-m boundary matrix, ∂, which stores the simplices of all dimensions in one place; that is,

$$\partial[i,j] \;=\; \begin{cases} 1 & \text{if } \sigma_i \text{ is a codimension-1 face of } \sigma_j; \\ 0 & \text{otherwise.} \end{cases}$$

In words, the rows and columns are ordered like the simplices in the total ordering and the boundary of a simplex is recorded in its column. The algorithm uses column operations to reduce ∂ to another 0-1 matrix R. Let $low(j)$ be the row index of the lowest 1 in column j. If the entire column is zero, then $low(j)$ is undefined. We call R *reduced* if $low(j) \neq low(j_0)$ whenever j and j_0, with $j \neq j_0$, specify two non-zero columns. The algorithm reduces ∂ by adding columns from left to right.

```
$R = \partial$;
for $j = 1$ to $m$ do
  while there exists $j_0 < j$ with $low(j_0) = low(j)$ do
    add column $j_0$ to column $j$
  endwhile
endfor.
```

The running time is at most cubic in the number of simplices. In matrix notation, the algorithm computes the reduced matrix as $R = \partial \cdot V$; see Figure VII.3. Since each simplex is preceded by its proper faces, ∂ is upper triangular. The j-th column of V encodes the columns in ∂ that add up to give the j-th column in R. Since we only add from left to right, V is also upper triangular and so is R.

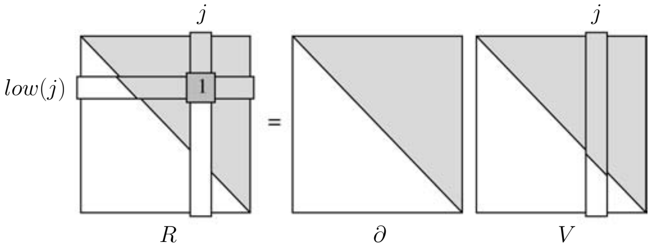

Figure VII.3: Reducing ∂ expressed as matrix multiplication. White areas are necessarily zero while entries in shaded areas can be either 0 or 1.

To get the ranks of the homology groups of K, we notice that the number of zero columns of R that correspond to p-simplices is the rank of Z_p. Similarly, the number of non-zero columns gives the rank of B_p. The difference is the p-th Betti number.

Pairing. However, there is significantly more information that we can harvest. To see this, we need to understand how the lowest 1s relate to the persistent homology groups. We begin by showing that they are unique, and this in spite of the fact that the reduced matrix, R, is not. Indeed, R is characterized by being reduced and is obtained by left-to-right column operations. But we may or may not continue the operations once we have reached a reduced matrix. To see that the lowest 1s are unique, we consider the lower left submatrix R_i^j of R whose corner element is $R[i,j]$. In other words, R_i^j is obtained from R by removing the first $i-1$ rows and the last $n-j$ columns. Since left-to-right column operations preserve the rank of every such submatrix, the rank of R_i^j is the same as that of the corresponding submatrix of ∂, the one similarly obtained by removing the first $i-1$ rows and the last $n-j$ columns. We consider the expression

$$r_R(i,j) \quad = \quad \operatorname{rank} R_i^j - \operatorname{rank} R_{i+1}^j + \operatorname{rank} R_{i+1}^{j-1} - \operatorname{rank} R_i^{j-1}$$

and note that $r_R(i,j) = r_\partial(i,j)$ for all i and j, where $r_\partial(i,j)$ has an analogous definition except when we take ranks of submatrices of ∂. To evaluate this expression, we observe that the linear combination of any collection of non-zero columns in R_i^j is again non-zero. It follows that the rank of R_i^j is equal to its number of non-zero columns. Now, if $R[i,j]$ is a lowest 1, then R_i^j has one more non-zero column than the other three submatrices, which implies $r_R(i,j) = 1$. If $R[i,j]$ is not a lowest 1, then we consider two subcases. If none of the columns from 1 to $j-1$ has its lowest 1 in row i, then R_i^j and R_{i+1}^j have the same number of non-zero columns and so do R_i^{j-1} and R_{i+1}^{j-1}. Second, if one of these columns has its lowest 1 in row i, then R_i^j has one more non-zero column than R_{i+1}^j and R_i^{j-1} has one more non-zero column than R_{i+1}^{j-1}. In either case, $r_R(i,j) = 0$. Since the ranks of the lower left submatrices of R are the same as those of ∂, we have a characterization of the lowest 1s that does not depend on the reduction process.

Pairing Lemma. We have $i = low(j)$ iff $r_\partial(i,j) = 1$. In particular, the pairing between rows and columns defined by the lowest 1s in the reduced matrix does not depend on R.

Now that we know for sure that the lowest 1s are not an artifact of the particular strategy used for reduction, we ask what exactly they mean. Note that column j reaches its final form at the end of the j-th iteration of the outer loop. At this moment, we have the reduced matrix for the complex consisting of the first j simplices in the total ordering. We distinguish the case in which column j ends up zero from the other in which it has a lowest 1.

Case 1: column j of R is zero. Consistent with the terminology introduced in Section V.4, we call σ_j positive since its addition creates a new cycle and thus gives birth to a new homology class.

Case 2: column j of R is non-zero. It stores the boundary of the chain accumulated in column j of matrix V and is thus a cycle. Again consistent with the terminology in Section V.4, we call σ_j negative because its addition gives death to a homology class.

The class that dies in Case 2 is represented by column j. We still need to verify that it is born at the time the simplex of its lowest 1, σ_i with $i = low(j)$, is added. But this is clear because the cycle in column j of R just died and all other cycles that die with it have 1s below row i; otherwise, we could further reduce the matrix and obtain $low(j) < i$, which contradicts the algorithm. It follows that the lowest 1s indeed correspond to the points in the persistence diagrams. More precisely, (a_i, a_j) is a finite point in $\mathrm{Dgm}_p(f)$ iff $i = low(j)$ and σ_i is a simplex of dimension p. In this case, σ_j is a simplex of dimension $p+1$. We have (a_i, ∞) in $\mathrm{Dgm}_p(f)$ iff column i is zero but row i does not contain a lowest 1. In other words, σ_i is positive, but it does not get paired with a negative simplex.

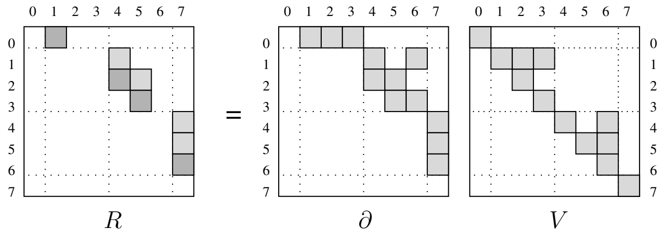

Figure VII.4: Reducing the boundary matrix of the complex consisting of a triangle and its faces. The shaded squares mark 1s in the matrices. The dark shaded squares mark lowest 1s in the reduced matrix.

An example. We illustrate the definitions with a small example. Let K consist of a triangle and its faces. To get a filtration, we first add the vertices, then the edges, and finally the triangle, numbering them in this order from 1 to 7. To make the exercise more interesting, we add the non-zero element of the (-1)-st reduced chain group as a dummy simplex of index 0 to compute reduced rather than ordinary homology. We recall that the augmentation map defines the boundary of each vertex as this dummy simplex. The resulting boundary matrix is shown as part of the matrix equation in Figure VII.4. We reduce it as described and get four non-zero columns in R. The first lowest 1 in R is in row 0 and column 1 and corresponds to the (-1)-dimensional reduced homology class that dies when we add vertex 1. The second lowest 1 is in row 2 and column 4. In words, the vertex 2 gives birth to the 0-cycle that the edge 4 kills. Similarly, the vertex 3 gives birth to the 0-cycle

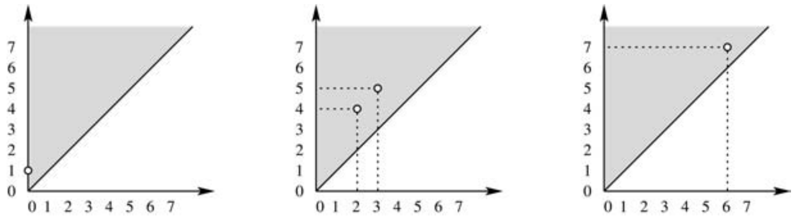

Figure VII.5: From left to right: the minus first, the zeroth, and the first persistence diagrams of the filtration that constructs a complex by first adding the three vertices, then the three edges, and finally the triangle.

that the edge 5 kills. Adding the edge 6 does not kill anything, which we see in the matrix since column 6 is zero. It corresponds to a 1-cycle obtained by adding the prior columns 4, 5, and 6, as indicated in V. The edge 6 thus gives birth to a 1-cycle that is then killed by the triangle 7. Figure VII.5 shows the corresponding three persistence diagrams which are drawn assuming the function value of a simplex is the same as its index. This particular function is injective, so all points in the diagrams have multiplicity one.

Bibliographic notes. The concept of persistent homology has been introduced for components by Frosini and Landi [73] and for general homology groups by Robins [127] and independently by Edelsbrunner, Letscher, and Zomorodian [60]. The latter paper gives the first fast algorithm for persistence, the same as described in this section but with the sparse matrix implementation discussed in the next section. A generalization of the notion of persistence to coefficient groups that are fields can be found in [161]; see also the monograph based on Zomorodian's thesis [160]. A recent survey on persistent homology is [57].

VII.2 Efficient Implementations

For practical applications, the number of simplices can be large so that storing the entire boundary matrix becomes prohibitive. As an alternative, we present a sparse matrix implementation of the Persistence Algorithm and give bounds on its running time that are better than cubic in the input size for many cases.

Sparse matrix representation. As in the previous section, we assume a monotonic function on a simplicial complex, $f : K \to \mathbb{R}$, and a compatible ordering of the simplices, $\sigma_1, \sigma_2, \ldots, \sigma_m$. We store the data using a linear array, $\partial[1..m]$, and a linked list of simplices per entry. The list in $\partial[j]$ corresponds to the j-th column of the boundary matrix, storing the codimension-1 faces of σ_j. By the end of the algorithm, the list in the j-th array entry corresponds to the column of the reduced matrix whose lowest 1 is in the j-th row. If there is no such column, then the list will be empty. To emphasize the transition, we change the name for the array from ∂ at the beginning to R at the end of the algorithm. All lists are sorted in the order of decreasing index so that the most recently added simplex is readily available at the top; see Figure VII.6. We see a general migration of the lists from right to left.

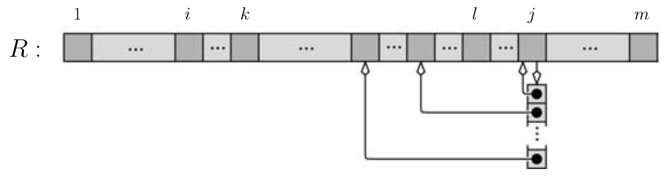

Figure VII.6: The sparse matrix representation of the reduced matrix with only one linked list shown.

To describe the algorithm that governs this migration, we write L for the linked list of the j-th array entry and $i = \text{TOP}(L)$ for the index of its top simplex. We call the i-th array entry *occupied* if it stores a non-empty list and *unoccupied* otherwise.

```
$R = \partial$;
for $j = 1$ to $m$ do
  $L = \partial[j].cycle$; $R[j].cycle = \text{NULL}$;
  while $L \neq \text{NULL}$ and $R[i]$ with $i = \text{TOP}(L)$ is occupied do
    $L = L + R[i].cycle$
  endwhile;
  if $L \neq \text{NULL}$ then $R[i].cycle = L$ endif
endfor.
```

Adding two lists means merging them while deleting both copies of every duplicate simplex. Since we store the lists in consistent sorted order, each addition can be done in parallel scans. It is instructive to compare this sparse matrix version of the Persistence Algorithm with its standard matrix implementation.

Analysis. The main structure of the sparse matrix implementation is that of two nested loops, the outer and the inner loop. The addition of two lists is another loop in disguise, so the running time is at most cubic in the input size. To improve on this first estimate, we define a *collision* as an attempt to deposit the list L that fails because the entry is occupied. Each collision requires the merging of two lists, which takes time proportional to the sum of their lengths. The loop ends when L runs empty or when the non-empty list L is successfully deposited. The first case identifies σ_j as giving birth to a homology class. The second case identifies σ_j as giving death and the simplex, σ_i, where the deposit happens as triggering the corresponding birth. Each list $R[k].cycle$ contains σ_k as its topmost simplex. Similarly, σ_k is the topmost simplex in L when it collides with the list in $R[k]$. Using modulo 2 arithmetic, σ_k gets deleted, which implies that the topmost simplex in the merged list has index less than k. The inner loop thus proceeds monotonically from right to left. It follows that collisions for a simplex σ_j happen only at entries between i and j, where $i = 1$ if σ_j gives birth and i is the index of the corresponding birth if σ_j gives death. Note that in the latter case, $j - i$ is what we call the index persistence of σ_j. Consider now the inner loop for σ_j. A collision at entry k can happen only if σ_k gave birth to a class that died at σ_l before σ_j is reached. We have $i < k < l < j$, as in Figure VII.6. Similarly, the collisions during the inner loop for σ_l correspond to birth-death pairs nested within $[k, l]$. Inductively, this implies that the lists added at collisions contain only faces of simplices with index in $[i, j]$. Letting p be the dimension of σ_j, the number of such faces is at most $p + 1$ times the number of indices in the interval. The time to merge two lists is therefore at most proportional to this number. In summary, the running time of the inner loop for a p-simplex σ_j is at most $(p + 1)(j - i)^2$.

There are situations in which we know ahead of time which simplices give birth and which give death. For example, if the complex is geometrically realized in $\mathbb{R}^3$, the Incremental Betti Number Algorithm described in Section V.4 gives such a classification. With this information, we can then save the effort for the simplices that give birth so that the total running time of the algorithm becomes output-sensitive, and in particular bounded by the dimension times the sum of squares of the index persistences. Assuming constant dimension, this is at most proportional to m^3, but for most practical data it is significantly smaller than that.

Zeroth diagram. The structure of the lists used to compute the 0-th persistence diagram is simpler than for dimensions beyond zero. This diagram depends solely on the vertices and edges of K and on their sequence in the compatible ordering. A vertex has no boundary and always gives birth to a component, so no choice there. An edge σ_j has two vertices as its boundary, $\partial\sigma_j = u + w$. Suppose u comes first, that is, $u = \sigma_i$, $w = \sigma_k$, and $i < k$. The first step of the algorithm is then its attempt to deposit the list L consisting of u and w in $R[k]$. If $L_k = R[k].\mathit{cycle}$ is empty, then the deposit is successful, σ_k, σ_j is a pair, and the inner loop ends. Otherwise, L_k is itself a list of two vertices, v and w in which v comes first. Adding the two lists gives $L + L_k$, which consists of u and v. Indeed, all non-empty lists have length two so that each addition takes only constant time. This implies that the total effort for dimension 0 is at most the sum of indices, for edges that give birth, and at most the sum of index persistences, for edges that give death. In any case, this is bounded from above by m^2.

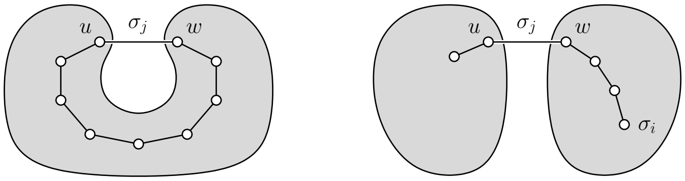

Figure VII.7: Adding the edge σ_j on the left gives birth to a 1-cycle while on the right it gives death to a component.

But we can do even better. Consider again the two cases for the edge with boundary $\partial\sigma_j = u + w$. It gives birth iff u and w belong to the same component of K_{j-1}, the complex right before we add σ_j; see Figure VII.7 on the left. Starting with σ_j, the algorithm adds an edge to the growing path at each collision, and L keeps track of its boundary, the two endpoints. Eventually, the two ends meet, L becomes empty, and the path becomes a 1-cycle. The edge σ_j gives death iff u and w belong to two different components of K_{j-1}; see Figure VII.7 on the right. The inner loop ends when one of the ends of the growing path reaches the first (oldest) vertex, σ_i, of one component. Since the inner loop works monotonically from right to left, this implies that the oldest vertex of the other component is even older. Following the Elder Rule, L gets deposited in $R[i]$ and σ_i, σ_j form a pair. Note that the outcome is predictable. All we need to know is whether or not u and w belong to different components in K_{j-1}, and if they do, which are the oldest vertices of these components. This is exactly the kind of information we can extract from the union-find data structure, as explained in Chapter I. Recall that this data structure stores each component as a tree of vertices. Given a vertex, we traverse the path up to the root to determine the name of the component. Using the index of the oldest vertex as the name gives the information we need at negligible cost. In summary, we compute the 0-th persistence diagram in time at most proportional to $m\alpha(m)$, where α is the inverse of the Ackermann function which, for all practical purposes, is bounded from above by a constant.

Surfaces. We now consider a simplicial complex, K, that triangulates a 2-manifold. This case is of some practical importance and it allows for a fast implementation of the Persistence Algorithm. Let $f : |K| \to \mathbb{R}$ be obtained by piecewise linear interpolation of its values at the vertices, as explained in Section III.1. There is possibly non-trivial information in the 0-th and the 1-st persistence diagrams of f but not in any of the others. To compute these two diagrams fast, we need to answer two questions.

1. How can we turn the 1-parameter family of sublevel sets into a filtration that we can feed to our algorithm?
2. How can we improve the slower running time for the 1-st persistence diagram to roughly the time needed for the 0-th diagram?

We deal with the first question now and defer the second question to later. Assume for simplicity that the restriction of f to the vertices of K is injective. As defined in Chapter VI, the lower star filtration is then the sequence $\emptyset = K_0 \subseteq K_1 \subseteq \ldots \subseteq K_n = K$, where K_i is the union of the lower stars of the first i vertices in the ordering by f. It is also the filtration generated by the monotonic function $g : K \to \mathbb{R}$ defined by mapping each simplex to $g(\sigma) = \max_{x \in \sigma} f(x)$. The diagrams of f are defined by the homology groups of the sublevel sets of f, $|K|_a = f^{-1}(-\infty, a]$, while those of g are defined by the homology groups of the sublevel sets of g, $K_a = g^{-1}(-\infty, a]$. By definition of lower star filtration, we have $|K_a| \subseteq |K|_a$, and the inclusion is a homotopy equivalence; see Figure VI.8 and the discussion around it. It follows that the vertical maps in the following diagram are isomorphisms:

$$\begin{tikzcd}
\mathsf{H}_p(|K|_a) \arrow[r] & \mathsf{H}_p(|K|_b) \\
\mathsf{H}_p(K_a) \arrow[r] \arrow[u] & \mathsf{H}_p(K_b), \arrow[u]
\end{tikzcd}$$

where p is any dimension and a, b, with $a \leq b$, are any two real numbers. The square commutes because all four maps are induced by inclusion. Indeed, these two conditions suffice for the diagrams defined by the two sequences to be the same.

Persistence Equivalence Theorem. Consider two sequences of vector spaces connected by homomorphisms $\phi_i : \mathsf{U}_i \to \mathsf{V}_i$:

$$\begin{tikzcd}
\mathsf{V}_0 \arrow[r] & \mathsf{V}_1 \arrow[r] & \ldots \arrow[r] & \mathsf{V}_{n-1} \arrow[r] & \mathsf{V}_n \\
\mathsf{U}_0 \arrow[r] \arrow[u] & \mathsf{U}_1 \arrow[r] \arrow[u] & \ldots \arrow[r] & \mathsf{U}_{n-1} \arrow[r] \arrow[u] & \mathsf{U}_n. \arrow[u]
\end{tikzcd}$$

If the ϕ_i are isomorphisms and all squares commute, then the persistence diagram defined by the U_i is the same as that defined by the V_i.

The proof is not difficult but it is tedious and is therefore omitted. As explained above, the 0-th persistence diagram of g can be computed in time at most proportional to $m\alpha(m)$. The equivalence with the 0-th persistence diagram of f thus implies that the latter can be computed in the same amount of time.

First diagram. Instead of computing the 1-st persistence diagram of f directly, we construct the 0-th persistence diagram of $-f$ and derive the diagram of f from it. We begin by describing the relation between $\mathrm{Dgm}_1(f)$ and $\mathrm{Dgm}_0(-f)$, omitting proofs since the relations are consequences of the more general theorems given in the next section.

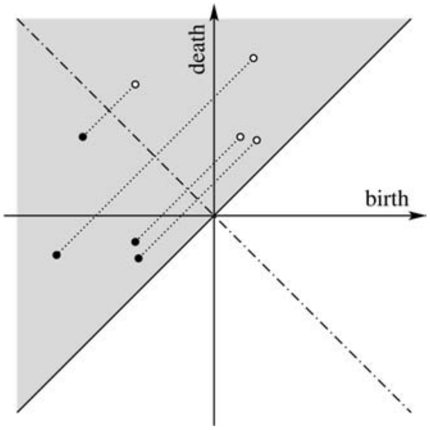

Figure VII.8: The white points of $\mathrm{Dgm}_1(f)$ are reflections of the black points of $\mathrm{Dgm}_0(-f)$ across the minor diagonal.

The 1-st persistence diagram of f consists of the diagonal, a finite portion of off-diagonal points (a, b), and an infinite portion of off-diagonal points (c, ∞). We construct the finite portion from the 0-th persistence diagram of $-f$. Specifically, the point (a, b) marks the birth of a 1-dimensional homology class at a and its death at b. Looking at $-f$ is like taking the complement and going backward. We thus have the birth of a 0-dimensional homology class at $-b$ and its death at $-a$. It follows that a point (a, b) belongs to $\mathrm{Dgm}_1(f)$ iff the point $(-b, -a)$ belongs to $\mathrm{Dgm}_0(-f)$. In other words, the finite portion of $\mathrm{Dgm}_1(f)$ can be obtained by reflecting the finite portion of $\mathrm{Dgm}_0(-f)$ across the minor diagonal, as illustrated in Figure VII.8. We get the points at infinity by partitioning the set of edges in the complex into three subsets: edges that give death in the lower star filtration of f, edges that give death in the lower star filtration of $-f$, and the rest. The first two contribute coordinates to the finite portions of the 0-th and the 1-st diagrams of f. For each edge in the third set, we have a point at infinity in the 1-st diagram, namely a class born when the edge is added and living on even when the complex K is complete. In summary, we have a three-pass algorithm for computing the persistence diagrams of a piecewise linear function f on a triangulated 2-manifold in time at most proportional to $m\alpha(m)$.

Bibliographic notes. The original paper on persistent homology by Edelsbrunner, Letscher, and Zomorodian [60] describes the sparse matrix version of the Persistence Algorithm explained in this section. Furthermore, the paper focuses on cases in which birth and death information is available using the Incremental Betti Number Algorithm by Delfinado and Edelsbrunner [45]. The standard matrix reduction version of the Persistence Algorithm came later historically and brought with it a more general appeal at the expense of increased computational resources. The Persistence Equivalence Theorem relating diagrams of different functions first appeared in [161].

VII.3 Extended Persistence

In this section, we discuss an extension of persistence that is motivated by an approach to fitting shapes to each other. The problem of fitting shapes arises when we solve a puzzle but also in the assembly of mechanical shapes, in the reconstruction of broken artifacts, and in protein docking.

Elevation on a surface. We give a brief sketch of the approach to fitting shapes and refer to Section IX.2 for a more detailed description. Let $\mathbb{M}$ be a smoothly embedded 2-manifold in $\mathbb{R}^3$. Given a direction $u \in \mathbb{S}^2$, the *height function* in this direction, $f : \mathbb{M} \to \mathbb{R}$, is defined by mapping each point x to $f(x) = \langle x, u \rangle$. We usually draw u vertically going up and think of the height as the signed distance from a horizontal base plane, as in Figure VII.9. Given a threshold $a \in \mathbb{R}$, we recall that the sublevel set consists of all points with height a or less, $\mathbb{M}_a = f^{-1}(-\infty, a]$. As mentioned in the previous sections, the sublevel sets are nested and define persistence through the corresponding sequence of homology groups. For a generic

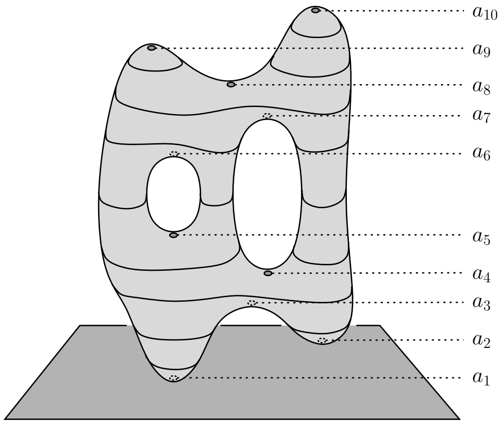

Figure VII.9: A smoothly embedded 2-manifold with level sets shown and critical points of the vertical height function marked.

smooth surface, the homological critical values of a height function are the height values of isolated critical points. If, furthermore, the direction is generic, then there are only three different types: minima which start components, saddles which merge components or complete loops, and maxima which fill holes. Assuming the critical points have distinct heights, the points in the persistence diagrams of f correspond to pairs of critical points. The elevation at the points x and y of such a pair is set to $|f(x) - f(y)|$. Since x is critical for two opposite directions, we need to make sure that the pairing is the same in both directions, else we get contradictory assignments of elevation. We also need all critical points to be paired; otherwise, we get white areas in which elevation remains undefined. The latter is the reason for why we extend persistence and the former is a constraint we need to observe in this extension.

Extended filtration. Let $a_1 < a_2 < \ldots < a_n$ be the homological critical values of the height function $f : \mathbb{M} \to \mathbb{R}$. At interleaved values

$$b_0 < a_1 < b_1 < a_2 < \ldots < a_n < b_n$$

we get sublevel sets $\mathbb{M}_{b_i} = f^{-1}(-\infty, b_i]$ which are 2-manifolds with boundary. Symmetrically, we define *superlevel sets* $\mathbb{M}^{b_i} = [b_i, \infty)$, which are complementary 2-manifolds with the same boundary. Finally, we use both to construct a sequence of homology groups going up and a sequence of relative homology groups coming back down:

$$\begin{array}{rcllcl} 0 & = & \mathsf{H}_p(\mathbb{M}_{b_0}) & \to \ldots \to & \mathsf{H}_p(\mathbb{M}_{b_n}) & \\ & = & \mathsf{H}_p(\mathbb{M}, \mathbb{M}^{b_n}) & \to \ldots \to & \mathsf{H}_p(\mathbb{M}, \mathbb{M}^{b_0}) & = \quad 0 \end{array}$$

for each dimension p. The homomorphisms are induced by inclusion. We recall that for modulo 2 arithmetic, the homology groups are isomorphic to the cohomology groups. Furthermore, Lefschetz duality implies $\mathsf{H}^p(\mathbb{M}_b) \simeq \mathsf{H}_{d-p}(\mathbb{M}, \mathbb{M}^b)$. This shows that the construction is intrinsically symmetric although not necessarily within the same dimension. Since we go from the trivial group to the trivial group, everything that gets born eventually dies. As a consequence, all births will be paired with corresponding deaths, as desired.

Tracing what gets born and dies in the relative homology groups is a bit less intuitive than for the absolute homology groups going up. However, we can translate the events between the absolute homology of $\mathbb{M}^b$ and the relative homology of the pair $(\mathbb{M}, \mathbb{M}^b)$. Coming down, the threshold decreases, so the superlevel set grows. We call a homology class in the superlevel set *essential* if it lives all the way down to b_0 and *inessential* otherwise.

Rule 1: a dimension p homology class of $\mathbb{M}^b$ dies at the same time that a dimension $p+1$ relative homology class of $(\mathbb{M}, \mathbb{M}^b)$ dies.

Rule 2: an inessential dimension p homology class of $\mathbb{M}^b$ gets born at the same time that a dimension $p+1$ relative homology class of $(\mathbb{M}, \mathbb{M}^b)$ gets born.

Rule 3: an essential dimension p homology class of $\mathbb{M}^b$ gets born at the same time that a dimension p relative homology class of $(\mathbb{M}, \mathbb{M}^b)$ dies.

We can prove these relationships by studying the kernels and cokernels of the maps from the homology groups of $\mathbb{M}^b$ into those of $\mathbb{M}$. Leaving this to the interested reader, we develop our intuition by considering an example.

Example. Consider the height function of the genus-2 torus in Figure VII.9. Going up, a_1 and a_2 give birth to classes in H_0, a_4, a_5, a_6, a_7, a_8 give birth to classes in H_1, and a_{10} gives birth to a class in H_2. All classes live until the end of the ascending pass, except for the dimension 0 class born at a_2, which dies at a_3, and the dimension 1 class born at a_8, which dies at a_9. These are the only two finite off-diagonal points in the ordinary persistence diagrams. Coming down, a_{10} kills the class in H_0 and a_9 gives birth to a class in H_1 that dies at a_8. Furthermore, a_7, a_6, a_5, a_4 kill the classes in H_1, a_3 gives birth to a class in H_2 that dies at a_2, and finally a_1 kills the class in H_2 that was born going up at a_{10}. To summarize, the pairs of critical values defining the points in the diagrams are $(a_1, a_{10}), (a_2, a_3)$ in dimension 0, $(a_4, a_7), (a_5, a_6), (a_6, a_5), (a_7, a_4), (a_8, a_9), (a_9, a_8)$ in dimension 1, and $(a_{10}, a_1), (a_3, a_2)$ in dimension 2. We show the diagrams in Figure VII.10 using different symbols for classes born and dying going up, born going up and dying coming down, and born and dying coming down. They make up the *ordinary*, the *extended*, and the *relative subdiagrams*, which we denote as Ord, Ext, and Rel, with the dimension in the index and the function in parentheses, as before. Note that the points of the ordinary subdiagrams lie above and those of the relative subdiagrams lie below the diagonal. The points of the extended subdiagrams can lie on either side.

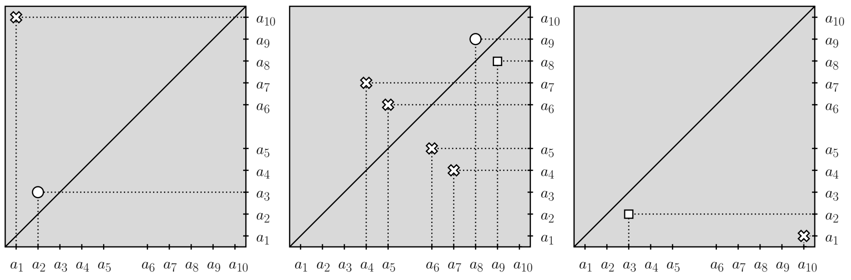

Figure VII.10: From left to right: the 0-th, 1-st, 2-nd persistence diagrams of the height function in Figure VII.9.

Duality and symmetry. The symmetries we observe in Figure VII.10 are not coincidental. They arise as consequences of Lefschetz duality between absolute and relative homology groups of complementary dimensions, $\mathsf{H}_p(\mathbb{M}_b) \simeq \mathsf{H}_{d-p}(\mathbb{M}, \mathbb{M}^b)$. This translates into a duality result for persistence diagrams, which we state without proof. We use a superscript 'T' to indicate reflection across the main diagonal, mapping the point (a, b) to (b, a).

Persistence Duality Theorem. Let f be a function on a d-manifold without boundary. Then the persistence diagrams are reflections of each other as follows:

$$\begin{array}{rcl} \mathrm{Ord}_p(f) & = & \mathrm{Rel}^T_{d-p}(f), \\ \mathrm{Ext}_p(f) & = & \mathrm{Ext}^T_{d-p}(f), \\ \mathrm{Rel}_p(f) & = & \mathrm{Ord}^T_{d-p}(f). \end{array}$$

Equivalently, the full p-th persistence diagram is the reflection of the full $(d-p)$-th persistence diagram, $\mathrm{Dgm}_p(f) = \mathrm{Dgm}^T_{d-p}(f)$. We have $d = 2$ for the example illustrated in Figures VII.9 and VII.10 and we indeed have diagrams that are reflections of each other as described. For $2p = d$, the extended subdiagram is the reflection of itself and is therefore symmetric across the main diagonal.

Recall that the definition of elevation requires that the pairing of critical points be the same for antipodal height functions. We can use duality to prove that they are indeed the same. More specifically, we have the following structural result, again expressed in terms of subdiagrams of the persistence diagrams. We use the superscript 'R' to indicate reflection across the minor diagonal, mapping the point (a, b) to $(-b, -a)$. Similarly, we use the superscript '0' to indicate central reflection or rotation by 180 degrees, mapping the point (a, b) to $(-a, -b)$.

Persistence Symmetry Theorem. Let f be a function on a d-manifold without boundary and let $-f$ be its negative. Then the persistence diagrams of the two functions are reflections of each other:

$$\begin{array}{rcl} \mathrm{Ord}_p(f) & = & \mathrm{Ord}^R_{d-p-1}(-f), \\ \mathrm{Ext}_p(f) & = & \mathrm{Ext}^0_{d-p}(-f), \\ \mathrm{Rel}_p(f) & = & \mathrm{Rel}^R_{d-p+1}(-f). \end{array}$$

In lieu of a proof, we just mention that each of the three equations can be obtained using the Persistence Duality Theorem together with the above three rules relating events in the parallel sequences of absolute and relative homology groups.

Lower and upper stars. To describe how we compute extended persistence, let K be a triangulation of a d-manifold $\mathbb{M}$. We assume the height function is defined at the vertices. We also assume that the height values are distinct, so we can index the vertices such that $f(v_1) < f(v_2) < \ldots < f(v_n)$. Let $f : |K| \to \mathbb{R}$ be obtained by piecewise linear extension. Writing $a_i = f(v_i)$ and introducing interleaved values $b_0 < a_1 < b_1 < \ldots < a_n < b_n$, we can define sublevel sets and superlevel sets as before. The set of points $x \in |K|$ with $f(x) \leq b_i$ is homeomorphic to $\mathbb{M}_{b_i}$ and thus is a manifold with boundary. Similarly, the set of points with $f(x) \geq b_i$ is homeomorphic to $\mathbb{M}^{b_i}$ and is a manifold with boundary. We can retract the partially used simplices and get homotopy equivalent subcomplexes of K. Specifically, let K_i be the full subcomplex defined by the first i vertices in the ordering and K^i the full subcomplex defined by the last $n - i$ vertices. The two subcomplexes of K

are disjoint although together they cover all n vertices. The only simplices not in either subcomplex are the ones that connect the first i with the last $n - i$ vertices. Recall that the lower star of a vertex v_i consists of all simplices that have v_i as their highest vertex. Symmetrically, we define the *upper star* to consist of all simplices that have v_i as their lowest vertex. More formally,

$$\begin{aligned} \mathrm{St}_{-}v_i &= \{\sigma \in \mathrm{St}\, v_i \mid x \in \sigma \Rightarrow f(x) \leq f(v_i)\}, \\ \mathrm{St}^{+}v_i &= \{\sigma \in \mathrm{St}\, v_i \mid x \in \sigma \Rightarrow f(x) \geq f(v_i)\}. \end{aligned}$$

Since every simplex has a unique highest and a unique lowest vertex, the lower stars partition K and so do the upper stars. With this notation, $K_0 = \emptyset$ and $K_i = K_{i-1} \cup \mathrm{St}_{-}v_i$ for $1 \leq i \leq n$. Equivalently, K_i is the union of the first i lower stars. Symmetrically, $K^n = \emptyset$, $K^i = K^{i+1} \cup \mathrm{St}^{+}v_{i+1}$, and K^i is the union of the last $n - i$ upper stars.

Computation. By the Persistence Equivalence Theorem in the previous section, the K_i have the same homotopy type as the sublevel sets, and the K^i have the same homotopy types as the superlevel sets of $\mathbb{M}$. We can therefore use them to compute persistence. Let A be the boundary matrix for the ascending pass, storing the simplices in blocks that correspond to the lower stars of v_1 to v_n, in this order. Within each block, we store the simplices in order of non-decreasing dimension and break remaining ties arbitrarily. All simplices in the same block are assigned the same value, namely the height of the vertex defining the lower star. If two simplices in the same block are paired, they define a point on the diagonal of the appropriate persistence diagram. In other words, the homology class dies as soon as it is born and therefore has zero persistence. Only pairs between blocks carry any significance.

Let B be the boundary matrix for the descending pass, storing the simplices in blocks that correspond to the upper stars of v_n to v_1, in this order. Using A and B, we form a bigger matrix by adding the zero matrix at the lower left and the permutation matrix P that translates between A and B at the upper right, as in Figure VII.11. We can think of the result as the boundary matrix of a new complex, namely the cone over K. We pick a new, dummy vertex, v_0, and for each i-simplex σ in K add the $(i+1)$-simplex $\sigma \cup \{v_0\}$. Adding the cone removes any non-trivial homology. This explains why reducing the big matrix works. As we move from left to right, we first construct K, forming pairs by reducing A. At the halfway point, the only unpaired simplices are the ones that gave birth to the essential homology classes. As we continue, we cone off K step by step, eventually removing all non-trivial homology. In the end, the ordinary, extended, and relative subdiagrams are given by the lowest 1s in the upper left, upper right, and lower right quadrants of the reduced matrix.

Indeed, we draw the diagram that corresponds to one of the three quadrants by marking each lowest one as a point, replacing indices by function values. For A, the birth values increase downward and the death values from left to right, so we need to turn the quadrant by 90° to get the ordinary subdiagram. Symmetrically, we turn the quadrant of B by $-90°$ to get the relative subdiagram and we reflect the quadrant of P across the main diagonal to get the extended subdiagram. Since

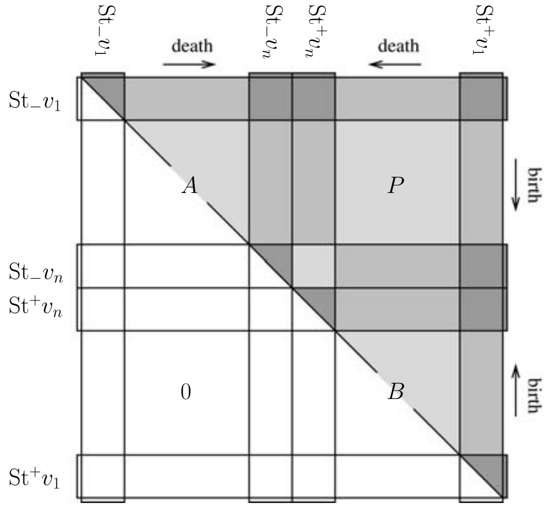

Figure VII.11: The block structure of the boundary matrix representing the construction of K going up and the subsequence destruction coming down.

the reduced versions of A and B are upper triangular, we indeed get the ordinary subdiagram above and the relative subdiagram below the diagonal.

Bibliographic notes. The extension of persistence described in this section is due to Cohen-Steiner, Edelsbrunner, and Harer [35]. It makes essential use of Poincaré and Lefschetz duality to obtain the desired symmetry properties for manifolds. The construction applies equally well to general topological spaces but without guarantee of duality and symmetry. The main motivation for the extension is the definition of the elevation function of a smoothly embedded surface in $\mathbb{R}^3$; see Section IX.2. This definition requires that all critical points be paired, which is not the case for ordinary persistence. The original paper on elevation contains an elementary description of extended persistence just for the case of surfaces [3].

VII.4 Spectral Sequences

Topologists will immediately recognize a connection between persistence and spectral sequences. We shed light on this relation by reviewing spectral sequences, first in terms of the matrix reduction algorithm and second in terms of groups and maps between them.

The matrix reduction view. As usual, we start with a filtration of a simplicial complex,

$$\emptyset = K_0 \subseteq K_1 \subseteq \ldots \subseteq K_n = K,$$

letting $k_i = \operatorname{card} K_i$ be the number of simplices in the i-th complex. Using a compatible total ordering of the simplices, we let ∂ be the boundary matrix which we write in block form. Specifically, ∂_i consists of the rows numbered $k_{i-1}+1$ to k_i corresponding to the simplices in $K_i - K_{i-1}$, and ∂^j consists of the columns numbered $k_{j-1}+1$ to k_j corresponding to the simplices in $K_j - K_{j-1}$. We write ∂_i^j for the intersection of the i-th block of rows and the j-th block of columns; that is, ∂_i^j records the codimension-1 faces of the simplices in $K_j - K_{j-1}$ that lie in $K_i - K_{i-1}$. Since the boundary matrix is upper triangular, we have $\partial_i^j = 0$ whenever $i > j$. We reduce the boundary matrix with left-to-right column additions, as before, but instead of sweeping the matrix from left to right, we sweep it diagonally. More precisely, we work in phases, and in Phase r, we reduce columns in ∂^j by adding columns in the blocks from ∂^{j-r+1} all the way to ∂^j itself. The Spectral Sequence Algorithm thus reduces the columns from the diagonal outward, as illustrated in Figure VII.12.

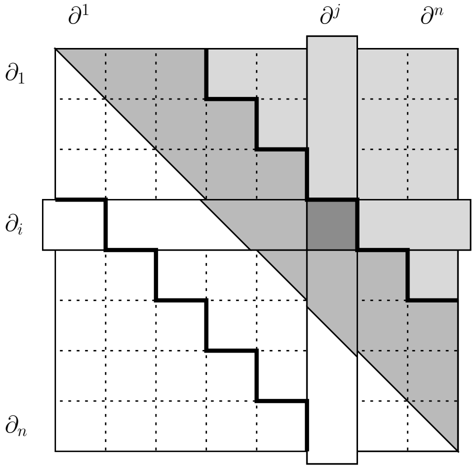

Figure VII.12: After three phases, the triple blocks along the diagonal are reduced. The highlighted blocks of rows and columns intersect in the block matrix ∂_i^j.

```
for $r = 1$ to $n$ do
  for $j = r$ to $n$ do
    for $\iota = k_{j-1} + 1$ to $k_j$ do
      while $\exists k_{j-r} < \iota' < \iota$ with $k_{j-r} < low(\iota') = low(\iota) \leq k_{j-r+1}$ do
        add column $\iota'$ to column $\iota$
      endwhile
    endfor
  endfor
endfor.
```

The result is the same as that of the Persistence Algorithm in the first section of this chapter; only the order in which the columns are added is different. An easy connection to persistence arises by considering the monotonic function $f : K \to \mathbb{R}$ mapping a simplex $\sigma \in K_i - K_{i-1}$ to $f(\sigma) = i$. A leftmost lowest one in ∂_i^j then belongs to a simplex pair of persistence $j - i$. The Spectral Sequence Algorithm thus computes the pairs in the order of non-decreasing index persistence.

Groups and maps. We now interpret the algorithm in terms of groups that make up the spectral sequence of the filtration. Recall the chain groups and boundary maps, $\partial : \mathsf{C}_p \to \mathsf{C}_{p-1}$, which form the chain complex defined by K. For each j, we let C_p^j be the group of p-chains of $K_j - K_{j-1}$, and for each chain $c \in \mathsf{C}_p^j$, we let $\partial_i^j c$ be the sum of terms of ∂c that lie in $K_i - K_{i-1}$. Suppressing the dimension in the notation for the boundary map, we have $\partial_i^j : \mathsf{C}_p^j \to \mathsf{C}_{p-1}^i$ and

$$\partial c \quad = \quad \partial_j^j c + \partial_{j-1}^j c + \ldots + \partial_1^j c.$$

The block ∂_i^j in the boundary matrix represents the maps ∂_i^j simultaneously for all dimensions. In spectral sequences, we approximate ∂ by the sum of maps ∂_j^j to ∂_i^j and then decrease i. The spectral sequence itself consists of a collection of groups $\mathsf{E}_{p,q}^r$ and maps $\mathsf{d}_{p,q}^r$ between them. To describe them, we break with the convention of using p for the dimension. Instead, we follow the convention entrenched in the spectral sequence literature in which the first subscript, p, identifies the block of columns, the sum of subscripts, $p+q$, gives the dimension, and the superscript, r, counts the phases in the iteration.

As usual, we think of the columns of the boundary matrix as generators of the chain groups. Limiting our attention to the p-th block of columns, ∂^p, we get the groups of $(p+q)$-chains of $K_p - K_{p-1}$, for all q. If we further limit ∂^p to the blocks of rows ∂_i to ∂_p, we effectively ignore any boundary in K_{i-1}. For $i = p$, this is equivalent to taking the relative chain groups, $\mathsf{C}_{p+q}(K_p, K_{p-1})$. For $i < p$, we have a subgroup of the relative chain group $\mathsf{C}_{p+q}(K_p, K_{i-1})$, namely the one generated by the $(p+q)$-simplices in $K_p - K_{p-1}$; see Figure VII.13. For what follows, it is

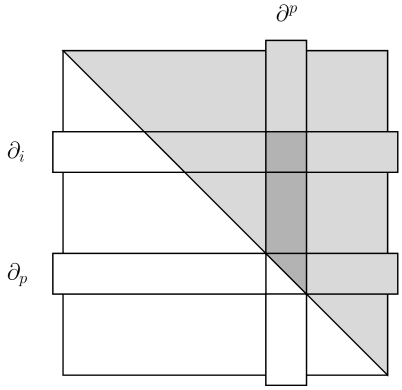

Figure VII.13: The darker shaded portion of the p-th block of columns represents the chains of $K_p - K_{p-1}$ and their boundaries in $K_p - K_{i-1}$.

important to remember that the boundary matrix, ∂, represents simplices of all dimensions at once. Hence, each block will correspond to a sequence of groups, one for each dimension.

The E^0-term of the spectral sequence. To prepare for the first phase of the algorithm, we focus on the diagonal blocks of the boundary matrix. Fixing $r = 0$, we write $\mathsf{E}^0_{p,q} = \mathsf{C}^p_{p+q}$ for the group of $(p+q)$-chains of $K_p - K_{p-1}$. Fixing p and varying q, these groups are generated by the p-th block of columns. Furthermore, we let

$$\mathsf{d}^0_{p,q} : \mathsf{E}^0_{p,q} \to \mathsf{E}^0_{p,q-1}$$

be defined by the $(p+q)$-dimensional boundary map restricted to the block ∂^p_p. In other words, $\mathsf{d}^0_{p,q}$ is ∂^p_p applied to $(p+q)$-chains. We note that $\mathsf{E}^0_{p,q}$ is isomorphic to the relative chain group $\mathsf{C}_{p+q}(K_p, K_{p-1})$ and $\mathsf{d}^0_{p,q}$ agrees with the corresponding relative boundary map. It follows that the maps satisfy the Fundamental Lemma of Homology, that is, $\mathsf{d}^0_{p,q-1} \circ \mathsf{d}^0_{p,q} = 0$. Indeed, a codimension-2 face of a $(p+q)$-simplex in $K_p - K_{p-1}$ either does not belong to $K_p - K_{p-1}$ or it does, but then both codimension-1 faces that contain it also belong to $K_p - K_{p-1}$. Hence, we get a chain complex,

$$\ldots \to \mathsf{E}^0_{p,q+1} \to \mathsf{E}^0_{p,q} \to \mathsf{E}^0_{p,q-1} \to \ldots,$$

in which the maps are implied. It is customary to draw this chain complex vertically, and adding the chain complexes for the other diagonal blocks, we get a 2-dimensional grid of groups, as shown in Figure VII.14. To reduce the clutter, we omit the arrows that connect the groups in each vertical line from top to bottom. We call this the E^0-*term* of the spectral sequence, noting that a vertical line in the grid contains all groups represented by a diagonal block of the boundary matrix.

$$\begin{array}{ccccccc}
 & \vdots & \vdots & \vdots & \vdots & \vdots & \\
\cdots & \mathsf{E}^0_{1,1} & \mathsf{E}^0_{2,1} & \mathsf{E}^0_{3,1} & \mathsf{E}^0_{4,1} & \mathsf{E}^0_{5,1} & \cdots \\
\cdots & \mathsf{E}^0_{1,0} & \mathsf{E}^0_{2,0} & \mathsf{E}^0_{3,0} & \mathsf{E}^0_{4,0} & \mathsf{E}^0_{5,0} & \cdots \\
\cdots & \mathsf{E}^0_{1,-1} & \mathsf{E}^0_{2,-1} & \mathsf{E}^0_{3,-1} & \mathsf{E}^0_{4,-1} & \mathsf{E}^0_{5,-1} & \cdots \\
\cdots & 0 & \mathsf{E}^0_{2,-2} & \mathsf{E}^0_{3,-2} & \mathsf{E}^0_{4,-2} & \mathsf{E}^0_{5,-2} & \cdots \\
\cdots & 0 & 0 & \mathsf{E}^0_{3,-3} & \mathsf{E}^0_{4,-3} & \mathsf{E}^0_{5,-3} & \cdots \\
 & \vdots & \vdots & \vdots & \vdots & \vdots &
\end{array}$$

Figure VII.14: The E^0-term of the spectral sequence. We have maps going vertically downward, from $\mathsf{E}^0_{p,q}$ to $\mathsf{E}^0_{p,q-1}$ for every choice of p and q.

The E^1-term. After interpreting the diagonal blocks of the original boundary matrix in terms of relative chain groups, we now push this interpretation through the phases of the algorithm. For the first phase, we take the homology of the above vertical complexes and define $\mathsf{E}^1_{p,q} = \ker \mathsf{d}^0_{p,q} / \mathrm{im}\, \mathsf{d}^0_{p,q+1}$. An element of $\mathsf{E}^1_{p,q}$ is thus the equivalence class of a chain $c \in \mathsf{C}^p_{p+q}$ with $\partial^p_p c = 0$, where two chains are equivalent if their difference lies in the image of ∂^p_p, taking of course the boundary map that applies to chains of one higher dimension. In other words, the element is a relative homology class and more generally $\mathsf{E}^1_{p,q} \simeq \mathsf{H}_{p+q}(K_p, K_{p-1})$. Representatives of $\mathsf{E}^1_{p,q}$ are computed by reducing the matrix ∂^p_p, which is what the algorithm does in Phase $r = 1$. The zero columns in ∂^p_p correspond to simplices that give birth and represent cycles. Some are paired and have zero persistence since their classes come and go within $K_p - K_{p-1}$. Others are not paired, and their cycles are the generators of $\mathsf{E}^1_{p,q}$. Next, we let

$$\mathsf{d}^1_{p,q} : \mathsf{E}^1_{p,q} \to \mathsf{E}^1_{p-1,q}$$

be defined by the $(p+q)$-th boundary map restricted to ∂^{p-1}_p. Recall that an element in $\mathsf{E}^1_{p,q}$ is represented by a relative $(p+q)$-cycle, c. Hence, $\partial^p_p c = 0$, but $\partial^{p-1}_p c$ is possibly non-zero and represents a class in $\mathsf{E}^1_{p-1,q}$. All this sounds complicated, but it is rather straightforward if interpreted in terms of the boundary matrix after one phase of the algorithm. As before, the boundary maps satisfy the Fundamental Lemma of Homology, $\mathsf{d}^1_{p-1,q} \circ \mathsf{d}^1_{p,q} = 0$, so we again get a chain complex:

$$\ldots \to \mathsf{E}^1_{p+1,q} \to \mathsf{E}^1_{p,q} \to \mathsf{E}^1_{p-1,q} \to \ldots .$$

Going back to the grid in Figure VII.14, we can see these complexes as horizontal lines going from right to left. Of course, we are now in the next phase, so we need to substitute $r = 1$ for the superscript 0 everywhere. This is the E^1-*term* of the spectral sequence.

The E^2-term. We take one more step before appealing to induction, taking the homology of the horizontal complexes, $\mathsf{E}^2_{p,q} = \ker \mathsf{d}^1_{p,q} / \mathrm{im}\, \mathsf{d}^1_{p+1,q}$. An element of $\mathsf{E}^2_{p,q}$ is the equivalence class of the sum of a chain $c \in \mathsf{C}^p_{p+q}$ and another chain $c' \in \mathsf{C}^{p-1}_{p+q}$. The chains satisfy $\partial^p_p c = 0$ and $\partial^p_{p-1} c + \partial^{p-1}_{p-1} c' = 0$, and being equivalent means that the difference lies in $\mathrm{im}\, \partial^p_p + \mathrm{im}\, \partial^p_{p-1} + \mathrm{im}\, \partial^{p-1}_{p-1}$. The group $\mathsf{E}^2_{p,q}$ is not a relative homology group by itself but a subgroup of one, namely $\mathsf{E}^2_{p,q} \oplus \mathsf{E}^1_{p-1,q+1} \simeq \mathsf{H}_{p+q}(K_p, K_{p-2})$. Representatives of $\mathsf{E}^2_{p,q}$ are computed by reducing the double block of matrices ∂^p_p, ∂^{p-1}_{p-1}, ∂^{p-1}_p, ∂^p_{p-1}. The first two have already been reduced, and the third is zero. Phase $r = 2$ completes the reduction of the double block for the remaining fourth matrix. Next, we let

$$\mathsf{d}^2_{p,q} : \mathsf{E}^2_{p,q} \to \mathsf{E}^2_{p-2,q+1}$$

be defined by the $(p+q)$-th boundary map restricted to ∂^p_{p-2}. By construction, an element of $\mathsf{E}^2_{p,q}$ is represented by a $(p+q)$-chain, c, whose boundary in $K_p - K_{p-2}$ is empty. Its boundary in $K_{p-2} - K_{p-3}$ is possibly non-empty and represents a class in $\mathsf{E}^2_{p-2,q+1}$, the image of the class of c in $\mathsf{E}^2_{p,q}$. Taking the thus defined boundary map twice gives zero again, so we get a chain complex,

$$\ldots \to \mathsf{E}^2_{p+2,q-1} \to \mathsf{E}^2_{p,q} \to \mathsf{E}^2_{p-2,q+1} \to \ldots ,$$

similar to before. Going back to the grid in Figure VII.14, we see this complex along a line of slope one half going from right to left. In other words, the groups are connected by knight moves in chess, two to the left and one up. Of course, we are now in the next phase, so we need to substitute $r = 2$ for the superscript 0 everywhere. This is the E^2-*term* of the spectral sequence.

Iteration. The process continues, and for general phase numbers r, the maps take the topologist's chess move, that is, r steps to the left and $r-1$ steps up:

$$\mathsf{d}^r_{p,q} : \mathsf{E}^r_{p,q} \to \mathsf{E}^r_{p-r,q+r-1}.$$

This gives a set of chain complexes, and we take homology to enter the next phase. Since K is finite, the maps are eventually zero and the sequence converges to a limit term, $\mathsf{E}^r = \mathsf{E}^\infty$ for r large enough. The homology groups of K are obtained by taking direct sums along the diagonal lines in the limit term for which the dimension is constant.

Before reaching the limit term, we may consider each class in $\mathsf{E}^r_{p,q}$ as generated by an "almost" cycle of dimension $p+q$. This is a chain whose boundary in $K_p - K_{p-r}$ is empty but may have non-empty boundary in K_{p-r}. It is either an essential cycle of K, or a cycle of persistence at least r, assuming the monotonic function $f : K \to \mathbb{R}$ that maps $\sigma \in K_p - K_{p-1}$ to $f(\sigma) = p$, as before. This leads to the following summary connection between persistence and spectral sequences.

Spectral Sequence Theorem. The total rank of the groups of dimension $p+q$ after $r \geq 1$ phases of the Spectral Sequence Algorithm equals the number of points in the $(p+q)$-th persistence diagram of f whose persistence is r or larger; that is,

$$\sum_{p=1}^{n} \operatorname{rank} \mathsf{E}^r_{p,q} \quad = \quad \operatorname{card}\{a \in \operatorname{Dgm}_{p+q}(f) \mid \operatorname{pers}(a) \geq r\},$$

where q decreases as p increases so that the dimension remains constant.

In the limit, for r large enough, we have $\sum_{p=1}^{n} \operatorname{rank} \mathsf{E}^r_{p,q} = \operatorname{rank} \mathsf{H}_{p+q}(K)$ equal to the number of points in the $(p+q)$-th persistence diagram whose persistence is infinite.

Bibliographic notes. A comprehensive account of spectral sequences can be found in [109]. The treatment in this section follows the more concise presentation in the survey of persistent homology [57]. Similar to persistent homology, working over a field is crucial for the construction of spectral sequences. Over $\mathbb{Z}$, there are extension problems to solve because of torsion; see [24].

Exercises

The credit assignment reflects a subjective assessment of difficulty. A typical question can be answered using knowledge of the material combined with some thought and analysis.

1. **Tetrahedron complex** (one credit). Let K consist of a tetrahedron and its faces.

   (i) Apply the matrix reduction algorithm to the filtration of K obtained by adding the simplices in the order of dimension.

   (ii) Do any of the three diagrams depend on the way you order the simplices of the same dimension?

2. **Matrix reduction revisited** (two credits). Change the standard matrix reduction implementation of the persistence algorithm described in Section VII.1 by adding each j-th column to columns on its right rather than adding columns on its left to it. Specifically, consider the following implementation.

   ```
   $R = \partial$;
   for $j = 1$ to $m$ do
     while there exists $j_0 > j$ with $low(j_0) = low(j)$ do
       add column $j$ to column $j_0$
     endwhile
   endfor.
   ```

   (i) Show that this implementation of the persistence algorithm generates the same lowest 1s as the standard matrix reduction implementation.

   (ii) Give an example for which this and the standard implementation of the persistence algorithm compute different reduced matrices.

3. **Sublevel sets** (two credits). Let $f : |K| \to \mathbb{R}$ be a piecewise linear function defined by its values at the vertices, $f(u_1) < f(u_2) < \ldots < f(u_n)$. Let b be strictly between $f(u_i)$ and $f(u_{i+1})$, for some $1 \leq i \leq n-1$, and recall that the sublevel set defined by b is $f^{-1}(-\infty, b]$.

   (i) Prove that the sublevel sets defined by b and by $f(u_i)$ have the same homotopy type.

   (ii) Draw an example for the case in which the sublevel sets defined by b and by $f(u_{i+1})$ have the same homotopy types, and another example for the case in which they have different homotopy types.

4. **Graphs without branching** (three credits). Let K be a 1-dimensional simplicial complex in which each vertex belongs to one or two edges. In other words, K is a simple graph whose components are paths and closed curves. Show that the sparse matrix implementation of the persistence algorithm described in Section VII.2 takes time proportional to the number of simplices in K.

5. **Persistence diagram** (one credit). Draw a genus-3 torus, consider its height function, and draw the non-trivial persistence diagrams of the function. Distinguish between points in the ordinary, extended, and relative subdiagrams.

6. **Breaking symmetry** (two credits). Design a topological space $\mathbb{X}$ and a continuous function $f : \mathbb{X} \to \mathbb{R}$ such that

(i) the persistence diagrams violate the Persistence Duality Theorem in Section VII.3;

(ii) the persistence diagrams violate the Persistence Symmetry Theorem in the same section.

7. **Matrix reduction once again** (one credit). Prove that the reduced matrix computed by the spectral sequence algorithm in Section VII.4 is the same as that generated by the persistence algorithm in Section VII.1.

8. **Parallel matrix reduction** (three credits). First, rewrite the Spectral Sequence Algorithm of Section VII.4 for the case in which each block, $K_j - K_{j-1}$, consists of a single simplex. Second, show that the thus simplified algorithm can be run on a parallel computer architecture using n processors taking time at most proportional to n^2.

Chapter IX

Applications

The primary application of the mathematical and computational tools introduced in the previous chapters is in data analysis, an activity that reaches into every discipline in science and engineering. The data may comprise the readings of an array of sensors, the pixels of an image, the accumulation of observations, or what have you. Invariably, there is noise in the data, which may be systematic or random. It may also reflect genuine properties of the measured phenomenon but at a scale that is outside the window of interest. The traditional approach to noise is to 'smooth' or 'regularize' the data, which invariably means we change the data. This is in contrast to the approach we advocate here, namely to measure the noise and not change the data. What is new is the measurement and the additional level of rationality and consistency it affords us. The four case studies selected to illustrate the possibilities that this paradigm affords us all start with biological data.

IX.1 Measures for Gene Expression Data

Our first application deals with 1-dimensional real-valued functions, the simplest kind of objects about which persistent homology can make meaningful statements. Such functions arise in the development of somites in vertebrates.

Background. Vertebrates are characterized by a spinal column consisting of a sequence of vertebrae that provide a periodic segmentation of their body along the axis. Mice are one example, with a spinal column of about sixty-five vertebrae. The numbers are larger for snakes, whose columns might be segmented into a few hundred vertebrae. This structure arises in the development of the embryo, when the vertebral precursors, the *somites*, are formed rhythmically from the presomitic mesoderm. This process is associated with a molecular oscillator that drives gene expression with a period corresponding to that of somite formation. We refer to this oscillator as the *segmentation clock*. The desire to fully understand this clock is the motivation for the work described in this section.

An early indication of the molecular underpinnings of somite development was the visual exposure of a cyclically expressed gene called lunatic fringe. Adding a fluorescent marker, its expression could be observed as a wave initiated in the posterior presomitic mesoderm. Migrating up, the wave narrows as it moves to the anterior, where the somites form.

Technology. The segmentation clock is one of the most reliable organic structures, and it has a built-in counter that terminates its rhythm after some number of periods. Its operation suggests an elaborate mechanism involving more than a few genes. Microarray technology offers a way to pursue the broad question of which genes are involved by testing the entire genome of an organism at once. An array is a 2-dimensional organization of array elements, each measuring the expression of a particular gene. This is done by depositing pieces of DNA that are specific to the RNA product of that particular gene. These pieces bind to copies of particular RNA strands, if they are present in the tissue probe. The binding event is made observable by fluorescence whose intensity quantifies the abundance of the particular strand in the tissue.

The organism of choice for the study of the segmentation clock is the mouse. We start with a microarray primed with the entire mouse genome. Copies of this array are used to measure the expression of all genes several times during a single period. In the mouse, a somite is developed roughly every two hours, and measurements are taken at seventeen time points in that interval. It is important to mention that this description is a simplification of the actual experiment. Tissue probes are taken from seventeen embryos and during five periods. Rather than timing the probes with a stopwatch, the time within a period is estimated from the state of the observed wave of lunatic fringe expression. Instead of quantified time, we thus have ranked time, events subjectively ordered by visual inspection. In the end, we have a series of seventeen measurements for each of about seven and a half thousand genes in the mouse genome. Each measurement is a real number representing the observed intensity at the particular array location, which quantifies the abundance of the corresponding strand of RNA.

Before discussing the mathematical analysis of this data, we draw attention to an inherent limitation that results from folding data from several periods into one. Suppose we have a gene that is rhythmically expressed but with a different period, say, three instead of two hours. The sorting process will shuffle the data collected for this gene, destroying any clear signal if there was one. It is thus reasonable to use this data to decide whether a gene is rhythmically expressed with a period consistent with somite development, but not whether a gene is rhythmically expressed at all, or what the most likely length of the period would be.

Lipschitz functions on the circle. We model the results of the time series of microarray experiments as a set of functions from the circle to the real numbers, $f : \mathbb{S}^1 \to \mathbb{R}$, one for each gene. The circle represents the two hours of one period, and the function tracks the abundance of the gene product within the period. Change requires energy, namely for the production and degradation of RNA. We use this as a justification to assume that f is Lipschitz, that is, there is a smallest positive constant, called the *Lipschitz constant* of f and denoted as $\mathrm{Lip}(f)$, such that $|f(s) - f(t)| \leq \mathrm{Lip}(f)\|s - t\|$, where the distance between s and t is measured along the circle. For a differentiable function, this is equivalent to constraining the derivative between $\pm\mathrm{Lip}(f)$. Defining the *total variation* as the integral of the norm of the derivative, we thus get

$$\mathrm{Var}(f) \quad = \quad \int_{s=0}^{2\pi} |f'(s)| \,\mathrm{d}s \quad \leq \quad 2\pi\mathrm{Lip}(f).$$

This inequality will be relevant shortly, when we study the stability of different ways to measure functions on the circle. To prepare this study, we consider the persistence diagram of f. It expresses the history of births and deaths in the sequence of sublevel sets, $f^{-1}(-\infty, a]$. Assuming f is Morse, we have the birth of a component at every minimum and the death of a component at every maximum, except at the last, global maximum at which we have the birth of a 1-dimensional class. This class never dies. Similarly, the 0-dimensional class born at the global minimum never dies. All other classes are born and die at finite values. Using the notation from Chapter VI, we write c_0 for the number of minima and c_1 for the number of maxima of f. Clearly, $c_0 = c_1$. By what we said above, the only persistence diagram that contains interesting information is the zeroth, $\mathrm{Dgm}_0(f)$, containing one point at infinity and $n = c_0 - 1 = c_1 - 1$ points in its finite portion, $\mathbb{R}^2$. Each finite point corresponds to a minimum paired with a maximum, and we write $x_i = (b_i, d_i)$, where b_i and d_i are the values of f at the minimum and the maximum. Let b_0 and b_{n+1} be the values of f at the global minimum and the global maximum, remembering that (b_0, ∞) is the point at infinity in $\mathrm{Dgm}_0(f)$ and (b_{n+1}, ∞) is the only point in $\mathrm{Dgm}_1(f)$. Consistent with the notation in the preceding chapter, we write

$$\Phi(f) \quad = \quad \sum_{i=1}^{n} (d_i - b_i)$$

for the total persistence of f. Note that this is the same as half the total variation minus the amplitude; that is, $\Phi(f) = \frac{1}{2}\mathrm{Var}(f) - (b_{n+1} - b_0)$. Indeed, $\Phi(f) + (b_{n+1} - b_0)$ is the sum of the values of f at the $n+1$ maxima minus the sum of the values at the $n+1$ minima. Decomposing f into increasing and decreasing portions, we can write $\mathrm{Var}(f)$ as the sum of two integrals, each equal to the same difference of sums. Hence, $\mathrm{Var}(f) = 2\Phi(f) + 2(b_{n+1} - b_0)$, as claimed.

Simplification. Before we introduce a measure for how close a function $f : \mathbb{S}^1 \to \mathbb{R}$ is to being periodic, in our assessment, we need to understand how we can simplify. Call another continuous function $f_\varepsilon : \mathbb{S}^1 \to \mathbb{R}$ an *ε-simplification* of f if

(i) $|f(s) - f_\varepsilon(s)| \leq \varepsilon$ for all $s \in \mathbb{S}^1$;

(ii) an off-diagonal point belongs to $\mathrm{Dgm}_p(f_\varepsilon)$ iff it belongs to $\mathrm{Dgm}_p(f)$ and its vertical distance from the diagonal exceeds ε.

Condition (ii) says the persistence diagrams of the two functions are the same except that the diagrams of f_ε have no points of persistence ε or less. We prove the existence of ε-simplifications by explicit construction of a function f_ε. It is convenient to assume that f is PL Morse. If f has only one minimum and one maximum, then its persistence diagrams have no finite points and f is its own ε-simplification for every real number $\varepsilon \geq 0$. So assume f has at least two minima and two maxima. Let u and v be a pair with minimum persistence and let $x = (f(u), f(v))$ be the corresponding point in the zeroth diagram. By assumption of minimality, the function value increases monotonically to $c = f(v)$ on both sides of u. Let $t \neq v$ be the point with $f(t) = c$ reached from u going in the direction away from v. If $f(v) - f(u) < \varepsilon$, we change f by setting $f(s) = c$ for all points s on the arc from t to v that contains u. The values on the complementary arc are preserved. We can make the new function PL Morse by giving a subtle slope to the flat interval between v and t, slightly extending it beyond t to pick up a small amount of height. The persistence diagrams of the new f are the same as before, except that the point x has disappeared. We get f_ε by repeating this step for all minimum-maximum pairs with persistence ε or less. This construction is illustrated in Figure IX.1, which shows the function for the gene Axin2 along with three simplifications.

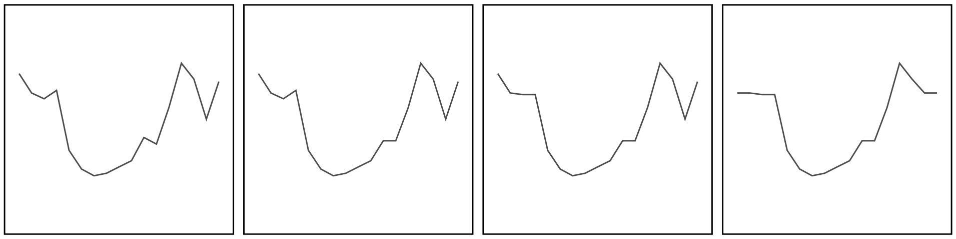

Figure IX.1: Simplifications of the expression profile of the gene Axin2. From left to right: the original function and the simplified functions obtained by canceling one, two, and all three minimum-maximum pairs. Notice that the last cancellation affects the curve at both ends, because the domain of the function is the circle.

Measures. The sine function, which maps points of $\mathbb{S}^1$ to their second Cartesian coordinates in $\mathbb{R}^2$, is the prototypical periodic function. It has a single minimum and a single maximum and varies smoothly between the two. Allowing for more general patterns to increase and decrease, we retain the property of having only two critical points as the characteristic ideal of a periodic function. To quantify periodicity more generally, we assign zero to a function with $c_0 + c_1 = 2$ and a positive number to every other function. Again, we find it convenient to restrict the discussion to PL Morse functions. Specifically, we set $\mu_0(f) = \frac{1}{2}(c_0+c_1)-1$, and for every positive integer q, we define the *degree-q periodicity measure* by integrating the degree-$(q-1)$ measure over the ε-simplifications of f:

$$\mu_q(f) \quad = \quad \int_{\varepsilon \geq 0} \mu_{q-1}(f_\varepsilon)\, \mathrm{d}\varepsilon.$$

Note that $\mu_1(f)$ is proportional to the average number of critical points of the ε-simplifications. To see that $\mu_q(f)$ is well defined, we show that the measures do not depend on which ε-simplifications we use. We do this by proving that μ_q is equal to the *degree-q total persistence* of f defined as $\Phi^q(f) = \sum_{i=1}^n (d_i - b_i)^q$.

Periodicity Measure Lemma. Let $f : \mathbb{S}^1 \to \mathbb{R}$ be a PL Morse function. Then $\mu_q(f) = \Phi^q(f)$ for all non-negative integers q.

Proof. We use induction over q to prove that the contribution of the point $x_i = (b_i, d_i)$ in $\mathrm{Dgm}_0(f)$ to the degree-q periodicity measure is $(d_i - b_i)^q$. For $q = 0$, this point contributes one to $\mu_0(f)$ as well as to $\Phi^0(f)$. This establishes the base case. Let $q \geq 1$. By definition of ε-simplification, the point x_i belongs to the zeroth diagram of f_ε for all $0 \leq \varepsilon < d_i - b_i$ but not for any larger values of ε. The contribution of x_i to the degree-q periodicity measure is therefore $d_i - b_i$ times its contribution to the degree-$(q-1)$ measure, which, by inductive assumption, is $(d_i - b_i)^{q-1}$. Summing over all points x_i, for $1 \leq i \leq n$, gives $\Phi^q(f)$. There are no other finite points in the diagrams of the f_ε, which implies the claim. ⊔⊓

The Periodicity Measure Lemma implies an algorithm for computing $\mu_q(f)$, namely constructing the zeroth persistence diagram and summing the q-th powers of the vertical distances of its finite points from the diagonal. It also provides a definition of μ_q for real values q that are not integers.

Instability for small degree. Whether or not the periodicity measure is stable depends on the choice of q. Clearly, μ_0 is not stable because arbitrarily small perturbations can change the measure by an arbitrary amount. Perhaps less obviously, μ_1 is also not stable. Perhaps less obviously, μ_1 is also not stable. To see this, we construct a series of Lipschitz functions, $g_k : \mathbb{S}^1 \to \mathbb{R}$, that approach the zero function while their total persistence approaches π. Replacing each point of $\mathbb{S}^1$ by its angle, $\varphi \in [0, 2\pi)$, we set $g_0(\varphi) = \min\{\varphi, 2\pi - \varphi\}$ and define $g_k(\varphi) = \frac{1}{2} g_{k-1}(2\varphi)$ for all positive integers k; see Figure IX.2. The maximum difference between g_k and

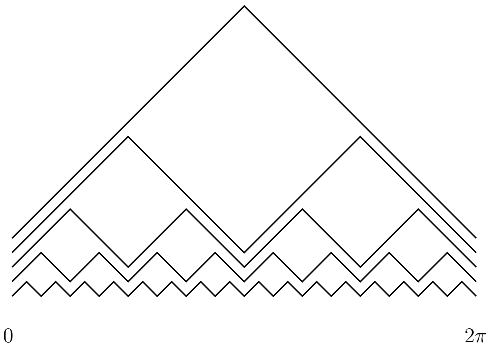

Figure IX.2: The graphs of the functions g_k, for $k = 0, 1, 2, 3, 4$, with vertical off-set for clarity.

the zero function is $\|g_k\|_\infty = \max_{0 \le \varphi < 2\pi} g(\varphi) = \pi/2^k$, which goes to zero as k goes to infinity. On the other hand, every function g_k has slope ± 1 almost everywhere. The total variation is therefore $\mathrm{Var}(g_k) = 2\pi$. We divide by two and subtract the amplitude to get the total persistence as $\Phi^1(f) = \pi - \pi/2^k$, which goes to π as k goes to infinity.

Stability for degree at least two. There is a qualitative difference between the periodicity measures when q passes from one to two. In particular, μ_q is stable for every constant $q \ge 2$.

Stability Theorem for Total Persistence. Let $f, g : \mathbb{S}^1 \to \mathbb{R}$ be Lipschitz functions with Lipschitz constant one and let $q \ge 2$. Then

$$|\Phi^q(f) - \Phi^q(g)| \quad \le \quad 4q\pi^{q-1} \cdot \|f - g\|_\infty .$$

Proof. We begin by noting that $y^q - x^q = \int_x^y qt^{q-1}\,\mathrm{d}t \le q|y-x| \max\{x,y\}^{q-1}$ for all $x, y \ge 0$ and $q \ge 1$. We use the Stability Theorem for Tame Functions in Chapter VIII to index the persistences of the finite points in the zeroth diagrams of f and g such that

$$\begin{aligned} \Phi^1(f) &= \phi_1 + \phi_2 + \ldots + \phi_m, \\ \Phi^1(g) &= \gamma_1 + \gamma_2 + \ldots + \gamma_m, \end{aligned}$$

and $|\phi_i - \gamma_i| \le 2\varepsilon$ for all i, where $\varepsilon = \|f - g\|_\infty$, possibly after adding zeros. Both sums are bounded from above by half the total variation, which implies $\Phi^1(f) + \Phi^1(g) \le 2\pi$. We also note that $\phi_i \le \pi$ and $\gamma_i \le \pi$ for $1 \le i \le n$. Writing $\Delta = \Phi^q(f) - \Phi^q(g)$, we therefore get

$$\begin{aligned} |\Delta| &\le \sum_{i=1}^m |\phi_i^q - \gamma_i^q| \\ &\le \sum_{i=1}^m q|\phi_i - \gamma_i| \max\{\phi_i, \gamma_i\}^{q-1} \\ &\le q(2\varepsilon)\pi^{q-2} \sum_{i=1}^m \max\{\phi_i, \gamma_i\}. \end{aligned}$$

The sum in the last expression is bounded from above by $\sum_{i=1}^m (\phi_i + \gamma_i) \le 2\pi$. The claimed inequality follows. □

For constant $q \ge 2$, the right-hand side of the inequality in the theorem is at most some constant times the L_∞-difference between the two functions. It follows that the difference between the degree-q total persistences goes to zero as the difference between the functions goes to zero. The above theorem is thus a statement of stability for total persistence and therefore for the periodicity measure.

Notes. The background for the material in this section is provided by the biological work on somite development in Pourquié's group; see e.g. [118, 123]. The microarray time series data of the mouse genome forms the motivation for our work. It had originally been analyzed using a variant of Fourier analysis [47]. Because of limitations in the discerned patterns, the same data was later re-analyzed using four custom-made mathematical methods, all designed to recognize rhythmic gene expression of the kind exhibited by a small number (fewer than 30) of genes verified to participate in somite development. One of these methods was the periodicity measure described in this section. Assessing all seven and a half thousand expression profiles, each method generated a list of the genes, ordered from most to least compatible with the rhythm of somite development. These lists were then compared on the basis of their ranking of the verified genes. The results of the comparison can be found in [46], including the discussion of a small number of newly identified genes.

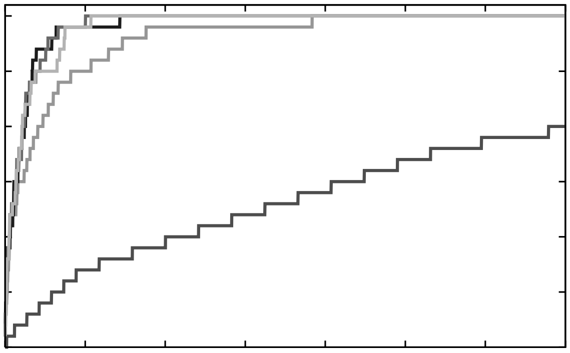

Figure IX.3: The step functions characterizing the distribution of the verified genes in the ordered lists generated with the periodicity measures μ_q, for $q = 0, 1, 2, 3, 4$. For better visibility, we truncate the lists after the first three and a half thousand genes. Moving northwest, toward the upper left corner of the rectangle, we first cross the graph of f_0, then that of f_1, and finally the graphs of f_2, f_3, f_4 in an order that depends on the exact route we choose.

The particular periodicity measure used in the re-analysis was μ_2, which we preferred over the other choices because $q = 2$ is the smallest power for which we know that the measure is stable. There is indeed evidence that stability is an important property for the task at hand. This is illustrated in the following comparison of the measures for $q = 0, 1, 2, 3, 4$. For each q, we generate an ordered list of the genes, as before, and we construct a step function, $f_q : [0, 1] \to [0, 1]$, that counts the verified genes in every initial segment of the list. In other words, $f_q(x)$ equals the percentage of the total number of verified genes that lie within the initial x percent of the list. If the verified genes are distributed evenly among the others, then we get a step function whose graph is close to the diagonal. On the other hand, if the verified genes are all listed first, then the function shoots up to one and stays there until the end. In general, one measure performs better than another if the first function majorizes the second. As shown in Figure IX.3, there is indeed a marked difference between the unstable measures, μ_0 and μ_1, and the stable measures, μ_2, μ_3, and μ_4. In summary, the graph of f_0 is slightly above the diagonal, indicating that μ_0 performs only marginally better than giving a random ordering. The visibly most striking improvement is from μ_0 to μ_1. However, as we get closer to the ideal step function, improvements are more difficult to come by, so the improvement from μ_1 to μ_2 is also significant. Thereafter, the graphs for $q = 2, 3, 4$ are almost indistinguishable.

Recall that the periodicity measures are defined in terms of ε-simplifications of the expression profiles. The concept of an ε-simplification was introduced in [61], where the main result is a construction for functions on 2-manifolds. As described in this section, existence is obvious for functions on a 1-manifold. The situation is much less understood for functions on a 3-manifold. The question of the stability of the total persistence for Lipschitz functions was considered in [36]. Similar to the degree-q Wasserstein distance between diagrams studied in Section VIII.2, the difference between degree-q total persistences goes to zero as the functions approach each other for some values of q and not for others. For both concepts, the qualitative change happens at a value of q that depends on the dimension of the manifold on which the Lipschitz functions are defined.

IX.2 Elevation for Protein Docking

In this section, we express the protrusions and cavities of a surface using a real-valued function whose design is motivated by the 3-dimensional shape matching problem central to the molecular basis of life.

Background. According to the central dogma of biology, strands of DNA are transcribed to pieces of RNA, which are then translated into proteins. Transcription works by complementarity, while translation is more involved, going from an alphabet of four nucleotides to one of twenty amino acids. Proteins are made of strings of amino acids. These strings are highly variable in order and length. In principle, this suggests an astronomical number of different possible proteins, but nature apparently uses only a tiny fraction of perhaps a few hundred thousand types. Once a protein has been formed, it folds into a characteristic shape. This shape determines its function, that is, how the protein acts within its environment and, in particular, how it interacts with and binds to other proteins.

The interaction between proteins is one of the most fundamental processes in biology and holds the key to how biological systems work. Cells send signals to one another and build machines that perform the many tasks that make life possible. To understand these and other processes, it would be wonderful if we could predict which proteins interact with which other proteins simply by knowing their shapes and the forces exerted by their atoms. This is the *protein docking problem*, the computational prediction of protein interaction. However, this prediction has proven notoriously difficult. There is significant debate in the biochemistry community about the relative importance of the geometry (shape) and the physics (forces), but it is clear that both are involved. It stands to reason that the relative importance of the geometry increases with the size of the involved molecules. But proteins flex, so geometry alone cannot predict the matching of undocked proteins. Nevertheless, geometric analysis is the first step.

Technology. The starting point for most docking efforts is geometric structures of proteins and other molecules collected by the biochemical community. The Protein Data Bank is an archive that contains information about experimentally determined geometric structures of proteins and nucleic acids. Data comes in the form of 3-dimensional atomic coordinates labeled by atom type and other descriptors. Data is determined primarily using two technologies, x-ray crystallography and nuclear magnetic resonance. For the former, biochemists crystallize the molecule and image the crystallized arrangement with a beam of x-rays that scatter in a variety of directions. From the angles and intensities of the scattered rays, a 3-dimensional picture of the density of electrons in produced. This density then allows the estimation of the positions of atoms in the crystal, as well as their chemical bonds. In contrast, nuclear magnetic resonance aligns nuclei with a magnetic field and perturbs this alignment with an orthogonal field. The response to the perturbation is then used to estimate the location of individual atoms.

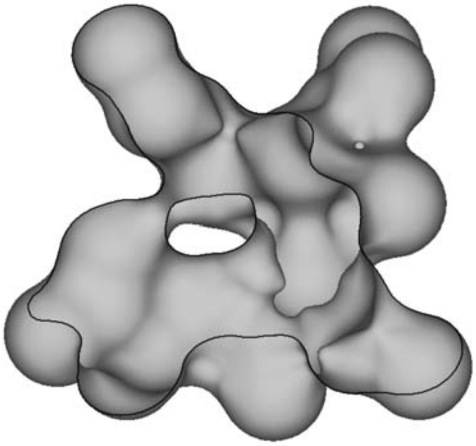

Figure IX.4: Cut-away view of a molecular skin surface.

Protein surfaces. Given atomic coordinates, we are interested in features on the surface of a protein that suggest binding configurations. The first problem is to define what we mean by the protein surface. Individual atoms attract and repel one another in several ways. The strong forces that hold the molecule together are chemical bonds and electrostatic interaction of ions. A weaker set of forces, known collectively as the *van der Waals force*, is strongly repulsive at short distance, attractive at medium distance, and negligible at large distance. It favors a fixed distance along a large patch of contact, which is the reason why geometry plays a role in the interaction. To model this contact, we place a small sphere, called the *van der Waals sphere*, around the center of each atom. To define a surface, we keep the spheres fixed while rolling a ball about the configuration, always touching but never overlapping any of the spheres. The radius of the ball is chosen to approximate that of a water molecule. As the ball rolls around, it traces out the *molecular surface*, which is made up of sphere and torus patches. This is roughly how the surrounding water experiences the protein. Except for the occasional sharp edge formed by intersecting blending surfaces, the patches meet to form a continuous bundle of normal vectors. If continuity is important everywhere, we may alternatively use the *molecular skin*, which consists of sphere and hyperboloid patches; see Figure IX.4. However a protein surface is defined, we look for protrusions and cavities that might line up when two proteins interact. The mathematical tool we use to do this is called elevation, and it can be defined for curves in the plane or surfaces in 3-dimensional space. Although our primary application is to surfaces, we simplify the discussion by restricting ourselves to curves.

Curves in the plane. Suppose $\mathbb{M} \subseteq \mathbb{R}^2$ is the image of a smooth embedding of the circle. Define $F : \mathbb{M} \times \mathbb{S}^1 \to \mathbb{R}$ by mapping each point $x \in \mathbb{M}$ and each $u \in \mathbb{S}^1$ to the height of x in the direction u, that is, $F(x, u) = \langle x, u \rangle$. Fixing a direction, we get $f_u : \mathbb{M} \to \mathbb{R}$ defined by $f_u(x) = F(x, u)$, the height function in the direction u. We are interested in conditions for which this height function is Morse. Recalling the definition in Section VI.1, we note that f_u may fail to be Morse for two reasons, namely because it contains a degenerate critical point or it has two critical points sharing the same height value.

A simple degenerate critical point is modeled by the family of functions $g_s(t) = t^3 + st$. For $s < 0$, we have two critical points, one a local maximum and the other a local minimum. For $s = 0$, we have a degenerate critical point at $t = 0$, and for $s > 0$, we have no critical points. As s goes from negative to positive, the pair of critical points cancel each other. We call this event a *cancellation*, or an *anti-cancellation* if we go in the other direction. Similarly, we can use a parametrized fourth degree polynomial to model an *interchange* at which two critical points momentarily share the function value. In our case, varying the parameter, s, corresponds to moving the direction such that the critical points slide on the curve. A cancellation occurs when two critical points collide, which happens at an inflection point. This motivates us to assume that the curve has only a finite number of inflection points and only a finite number of lines that are tangent at two or more points. It follows that there are only a finite number of directions for which f_u is not Morse. Equivalently, the 1-parameter family of height functions passes through only a finite number of cancellations, anti-cancellations, and interchanges.

Elevation function. When f_u is a Morse function, we can use extended persistence to pair up its critical points. These are the points for which u is normal to the curve, and we associate to each the persistence of the pair to which it belongs, calling this real number the *elevation* of the point. By the Persistence Symmetry Theorem of Section VII.3, the pairing is the same if we substitute $-u$ for u. It follows that f_u and $f_{-u} = -f_u$ define the same elevation values for the same points. In other words, elevation depends only on the normal line to the curve. Since every point of $\mathbb{M}$ has a unique normal line, this defines the *elevation function*, $E : \mathbb{M} \to \mathbb{R}$, except at points at which the height functions are not Morse. We take limits to define E also at these exceptional points.

Recall that a cancellation happens at an inflection point, x, for which we set $E(x) = 0$. Indeed, it is easy to see that the two critical points are paired right before they meet and cancel each other at x. The limit is the same from both sides, namely zero, which implies that E is continuous at x. At the points involved in an interchange, however, we may have different left and right limits and thus an ambiguity of how to define E. This is illustrated in Figure IX.5. Here, the point x would be paired with either y or z. In fact, if we rotate the vertical, upward directed vector u slightly to the right, the critical point near x is paired with the critical point near y. In contrast, if we rotate u slightly to the left, the critical point near x is paired with the critical point near z. Thus, there is a jump from y to z in moving the normal from right to left. The elevation near x varies continuously, so we can simply set $E(x)$ equal to the absolute height difference between x and y, which is the same as the absolute height difference between x and z. But the left and right limits at y are different, jumping from the absolute height difference between y and x to that between y and w. We get the same two different limits and the opposite jump at z. Continuity can therefore be obtained through surgery. Specifically, we cut $\mathbb{M}$ at y and at z and we glue the four ends in pairs to get a new curve on which E is continuous at the cut points. Of course, the new curve is no longer embedded in the plane. In this particular case, the new curve consists of two components, a long loop that contains w and a short loop that contains x. If we perform surgery at all such discontinuities, we get a curve, $\mathbb{N}$, on which E is everywhere continuous.

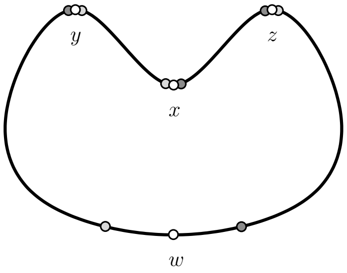

Figure IX.5: The four white points share the same normal direction, as do the four light shaded and the four dark shaded points.

Elevation maxima. Our interest in surgery is merely a means to an end, namely the determination of the interesting features of the curve. Call a point $x \in \mathbb{N}$ a *local maximum* of E if it has an open neighborhood such that $E(y) \leq E(x)$ for all y

in this neighborhood. For convenience, we assume the smooth curve is *generic* by which we mean

(i) it has only finitely many height functions that are not Morse;

(ii) its elevation function has only a finite number of local maxima.

Condition (i) has been discussed earlier, where it was used to define the elevation function. Condition (ii) prohibits curves of (locally) constant width, such as for example the circle. Consider the curve in Figure IX.5 as an example. Its elevation function has six local maxima, x and w, the two cut points formed by gluing the four ends obtained by cutting the curve at y and at z, as well as the leftmost point, p, and the rightmost point, q, of the curve (both not shown).

The local maxima come in pairs, by construction. For example, p and q form a pair, and $E(p) = E(q)$ is the Euclidean distance between p and q. Since neither point is a cut point, we call this pair a *one-legged elevation maximum*. We note that having p and q paired by extended persistence is necessary to form an elevation maximum but it is not sufficient. In the one-legged case, the line connecting p and q must be in the direction of the normal vector, and the curvature at p and at q must be such that a small rotation does not increase the local width. A second pair is formed by x and the cut point produced by gluing the right end at y to the left end at z. We call this a *two-legged elevation maximum* because we have two legs connecting x to y and to z on the original curve. Again, having x paired to y and to z by extended persistence is necessary but not sufficient to form a two-legged elevation maximum. We also need the property that the orthogonal projection of x onto the line of y and z falls between the points y and z. Finally, we have a third pair formed by w and the cut point produced by gluing the left end at y to the right end at z. This is another two-legged elevation maximum.

Piecewise linear curves. To design an algorithm for computing the elevation maxima of a curve, we face the usual dilemma that input is never smooth. Instead, we assume a simple, closed polygon with vertices $x_0, x_1, \ldots, x_{n-1}$ and edges e_i connecting x_i to x_{i+1}, for $0 \leq i < n$ where we take indices modulo n. We assume the polygon is *generic*, by which we mean that no two of the $\binom{n}{2}$ lines connecting the n vertices are parallel or orthogonal to each other. We may think of this polygon as approximating a smoothly embedded curve and this way get an idea of what the elevation maxima ought to be. Alternatively, we may approximate the polygon by a generic, smooth curve and obtain the definitions by limit considerations. This is what we do next.

Let P be the subset of $\mathbb{R}^2$ whose boundary is the polygon. We write P^ε for the set of points at distance at most ε from P and $(P^\varepsilon)^{-\delta}$ for the set of points of P^ε at distance more than δ from the complement. For $\delta = \frac{\varepsilon}{2}$ and sufficiently small $\varepsilon > 0$, the boundary of $(P^\varepsilon)^{-\delta}$ alternates between an arc on a circle with center x_i and radius δ and a straight line segment parallel to e_i. Finally, we replace the straight line segment by circular arcs of ever so small, positive curvature, κ, calling them the *chords* connecting the circular arcs around the vertices. While the resulting curve

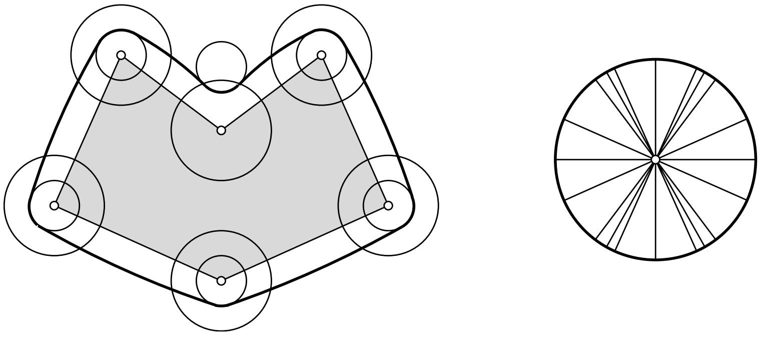

Figure IX.6: Turning a polygon into a generic curve. To simplify the drawing, we have chosen to ignore the requirements for a generic polygon. The circles at the vertices have radius ε and $\delta = \frac{\varepsilon}{2}$; they illustrate the construction of $(P^\varepsilon)^{-\delta}$. On the right, we see the circle of directions decomposed into arcs of constant height ordering.

is not smooth, its normal bundle is continuous and it satisfies the requirements of a generic curve, which suffices to define the elevation function; see Figure IX.6. For sufficiently small $\varepsilon > \kappa > 0$, the points on a chord are all paired with points on the same circular arc. In contrast, points on a circular arc may be paired with points of more than one chord or arc.

Algorithm. We begin by recalling the Extended Persistence Algorithm applied to a height function, f_u, defined on the polygon. Assume the vertices have distinct height values and they are relabeled such that $f_u(y_0) < f_u(y_1) < \ldots < f_u(y_{n-1})$. Then each vertex can be unambiguously classified as a minimum, a regular vertex, or a maximum. Since minima and maxima alternate along the polygon, we have the same number of each, and the algorithm outputs a pairing (a perfect matching) between the two collections. The global minimum, y_0, is necessarily paired with the global maximum, y_{n-1}. The other pairs depend on the sequence of sublevel sets or, equivalently, the lower star filtration of f_u, as explained in Chapter VII. We note that the pairing is the same for other piecewise linear functions for which the ordering of the vertices is the same. It thus suffices to run the Extended Persistence Algorithm for $\binom{n}{2}$ height functions, one per antipodal pair of arcs defined by the vertex pairs; see Figure IX.6. Skipping a few details, we note that this can be done in time at most some constant times n^3.

We now discuss how the $\binom{n}{2}$ pairings are used to extract all elevation maxima. As mentioned earlier, there are two types. We first discuss the one-legged elevation maxima. Let x_i, x_j be two vertices and $u = (x_j - x_i)/\|x_j - x_i\|$ the direction they define. Then x_i and x_j form a one-legged elevation maximum iff x_i and x_j are paired by the algorithm applied to f_u. In the piecewise linear case, the condition on the curvature at the two points is void. We second discuss two-legged elevation maxima. Let x_i, x_j, x_k be three vertices and let u be normal to $x_k - x_j$. Suppose furthermore that x_j and x_k lie on opposite sides of the line with direction u that passes through x_i. Let u_- and u_+ be directions sufficiently close to and on opposite sides of u. Then x_i and x_j, x_k form a two-legged elevation maximum iff x_i and x_j are paired for f_{u_-} and x_i and x_k are paired for f_{u_+}. In summary, the $\binom{n}{2}$ runs of the Extended Persistence Algorithm provide all the information we need to identify the elevation maxima of the polygon.

Notes. The difficulty of predicting the binding between proteins of known geometric structure combined with the importance of this question has lead to a community organized competition [87]. Using yet unpublished geometric structures determined by x-ray crystallography [52] or by nuclear magnetic resonance [159], the participants are asked to submit their best predictions, which are then compared to the observed configuration. The idea of using protrusions and cavities of protein surfaces to predict binding configurations goes back to Connolly [40]. He represents the protein by its molecular surface, which decomposes $\mathbb{R}^3$ into the *inside* and the *outside*, two 3-manifolds with disjoint interiors and common boundary, namely the molecular surface. Fixing a radius, $r > 0$, he places the center of a sphere with radius r at every point x of the surface and assigns to x the fraction of the sphere contained in the inside. As shown in [29], the limit of this function, as r approaches zero, is related to the mean curvature function of the surface. This should be contrasted to the relationship between the total mean curvature and the elevation that follows from integral geometric considerations described in [33]; see also Section VIII.3. As demonstrated in [152], the elevation maxima are useful in the coarse alignment of protein structures. This suggests we use elevation as a first pass toward predicting a binding configuration and refine the resulting alignments with methods that incorporate detailed knowledge of the physical behavior of molecular systems [130].

While this section focuses on the simpler setting of a curve embedded in $\mathbb{R}^2$, the important setting is of course that of a surface embedded in $\mathbb{R}^3$. This is described in the original paper on the subject by Agarwal, Edelsbrunner, Harer, and Wang [3]. In going from curves to surfaces, the ideas remain the same but get technically more complicated. For smoothly embedded surfaces, the construction is based on Cerf theory [30], which is part of differential topology. Instead of two, we get four types of elevation maxima; see Figure IX.7. Except for the one-legged case, each

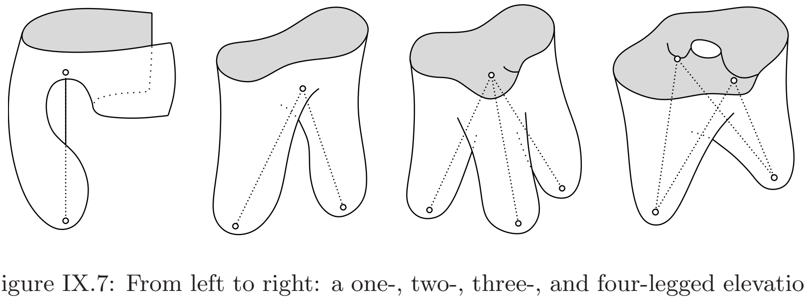

Figure IX.7: From left to right: a one-, two-, three-, and four-legged elevation maximum of a surface embedded in 3-dimensional space.

type arises at an ambiguity of the pairing of critical points produced by extended persistence. The algorithms are again for piecewise linear functions. Similar to the polygonal case, we construct all elevation maxima by running the Extended Persistence Algorithm for a finite collection of height functions. We refer to Section VII.2 for a fast implementation of the Persistence Algorithm for piecewise linear functions on a 2-manifold. Computing the extended persistence pairs is, however, more difficult and a similarly efficient algorithm requires sophisticated data structures [77].

IX.3 Persistence for Image Segmentation

In this section, we discuss image data and, in particular, the problem of segmenting the data into meaningful pieces. A popular approach is the watershed method, but it is sensitive to noise in the data, tending to overdo the segmentation. We show how to use persistent homology to cope with this difficulty.

Background. When we collect data about a physical phenomenon, we do so to varying degrees of resolution. Images are high-resolution data sets, representing shapes and scenes in great detail. A large part of biological and medical research, as well as medical practice, depends on technology that produces 2- and 3-dimensional images. But we can go beyond three dimensions, eg. with video sequences that unwind in time. There are also reasons for generating images synthetically, using methods such as Fourier transforms, for finding symmetries and for other purposes. The high resolution of image data suggests we think of it as a continuous object and apply methods from continuous rather than discrete mathematics for its analysis. By its nature, an image contains more than the desired information. Therefore the first task is often the extraction of interesting features. Capturing and describing these features is the province of image analysis. It includes tasks such as denoising, segmentation, registration, comparison, and more.

Technology. The last decades have witnessed a revolutionary change in how science is practiced, and this change is fueled by ever improving ways of acquiring data. Using new technology, we are able to collect high-resolution data on physical events that have traditionally been beyond our reach. Examples are sensor networks monitoring environments and microarrays measuring the expression of the entire genomes. These are relatively recent technologies for which we can expect rapid improvements in the accuracy and volume of collected data. More traditional imaging technologies generate 2- and 3-dimensional arrays of measurements.

Microscopy. This is an umbrella under which we distinguish different technologies depending on the medium used to generate the image. Optical microscopy involves light diffracted from an object passing through a lens to allow for a magnified view. Similarly, electron microscopy measures the diffraction of electromagnetic radiation by an object. In contrast, scanning probe microscopy, as the name suggests, measures the interaction of a probe with an object. In each case, the output is a

2-dimensional array of tiny squares, referred to as *pixels*, short for picture elements. Being 2-dimensional, the number of pixels is usually not more than perhaps a few million, which is easy to manage with current computer storage technology.

Magnetic resonance imaging. The principles are the same as for nuclear magnetic resonance, but the N-word has been dropped for medical applications. Here we use a magnetic field to align the nuclei of hydrogen atoms in water. Radio frequency fields are then used to systematically alter the alignment, which creates the signal detected by the scanner. This technology is widely employed in radiology to study the internal structure of the human body. The output is a 3-dimensional array of tiny cubes, referred to as *voxels*, short for volume elements. Being 3-dimensional, the voxel array provides a lot more information than a pixel array generated with microscopy. This wealth comes with a cost, namely the added difficulty of managing and analyzing such a large amount of data.

Segmentation. Once an image is acquired, it becomes a mathematical object that we can study. In particular, a 2-dimensional image is a function that is piecewise constant on a rectangular configuration of pixels; see Figure IX.8. Inside a pixel, the function is constant and measures intensity or gray value, and similarly for color pictures, except that we get three separate images, one for red, one for green, and one for blue. Given an image, the *segmentation problem* looks to identify regions of interest from these values. In Figure IX.8, these regions would be the cells imaged by microscopy. The general problem is hopelessly ill-defined but of major importance. As a result, the state of the art in the field is at best imperfect. Every type of image provides a different set of challenges, motivating a variety of approaches to the problem. Specifying global or adaptive thresholds to define the regions is a good first step. Methods of mathematical morphology can then be used to refine the result. Deformable models or level set methods solve differential equations to shrink-wrap a region with a curve or a surface. Region growing

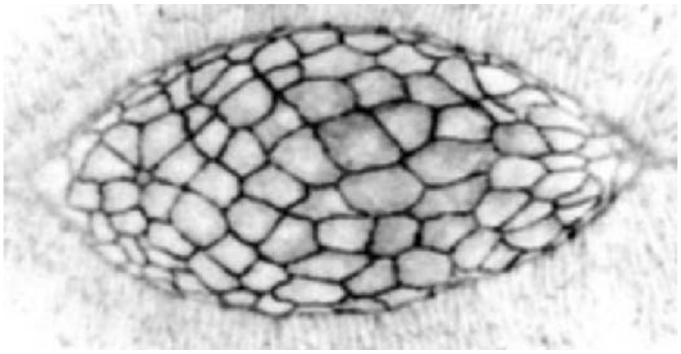

Figure IX.8: Confocal microscopy image of a cross-section of the cells in a Drosophila embryo during the developmental process known as dorsal closure [image courtesy of Daniel P. Kiehart and Adrienne Wells, Biology, Duke University].

and region splitting methods seek to improve segmentations by locally improving a quality assessment. In this section, we consider yet another approach, one that fits nicely into the framework of computational topology.

Watershed intuition. We begin with an intuitive description of the method. Let us treat an image as a continuous function defined on a region of the plane, usually a rectangle, although for this description we assume it is defined on all of $\mathbb{R}^2$. Think of the graph of this function, a surface in $\mathbb{R}^3$, which we imagine permeable, with soil below and air above. Now suppose it rains and the water level rises everywhere on the plane. As is common on planet Earth, we call *land* only the part of the surface above the water level. As the level rises, we see the land shrinking and its topology changing. When the water reaches a local minimum, a lake forms and grows as the level rises. When the water reaches a saddle point, two lakes merge into a single lake or an island separates from the mainland. When the water reaches a local maximum, the corresponding island has completely submerged under water.

We can keep the water from overflowing by building watershed lines as we pass saddles. These are the curves that separate lakes where the water meets as it rises. Mathematically, they form the unstable or ascending 1-manifolds corresponding to the saddles. They prevent the lakes from merging and form roads (anti-canals) between the islands and the mainland, thus maintaining the lakes as topological disks throughout the process.

The Watershed Algorithm. To formalize this process, we construct a piecewise linear function that represents a given image. For each pixel, we have a vertex at its center and we connect the vertices to form a triangulation. It is convenient to compactify by adding a dummy vertex to get a triangulation of $\mathbb{S}^2$. Specifying a value at each vertex, we get a function by piecewise linear interpolation; see Section III.1. More generally, we can begin with a triangulated 2-manifold and a piecewise linear function, $f : \mathbb{M} \to \mathbb{R}$. Recall that in Section VI.2, we constructed a complex whose vertices were the maxima, edges were the unstable 1-manifolds, and regions were the unstable 2-manifolds of the function. We now give an algorithm that constructs an approximation of the unstable manifolds. It is convenient to assume that the vertices have distinct function values and they are indexed such that $f(x_1) < f(x_2) < \ldots < f(x_n)$. We recall from Section VI.3 that each vertex is classified as regular or critical by looking at the values of f in its link. Call the part of the link spanned by vertices whose function value exceeds $f(x_i)$ the *upper link* of x_i. Then x_i is a minimum if the upper link is the entire link, a maximum if the upper link is empty, and a k-fold saddle if the upper link consists of $k+1$ components, each a path or an isolated vertex. The vertex is regular if it is a 0-fold saddle, that is, if it has a non-empty, connected, upper link that does not exhaust the entire link.

We process the vertices from lowest to highest. Specifically, we run a loop from $i = 1$ to $i = n$ and distinguish between the different types of vertices. Initially, all simplices in the triangulation are unmarked.

Case 1: x_i is a saddle. We mark x_i together with the edges that connect the saddle to the highest vertex in each component of the upper link.

Case 2: x_i is regular. If x_i has an incoming marked edge, then we mark x_i together with the edge to the highest vertex in the upper link.

Case 3: x_i is a maximum. We mark x_i.

In the end, we have $k + 1$ paths running upward from a k-fold saddle. Sometimes these paths merge, but then they continue together until they reach a maximum. The number of paths ending at a maximum varies depending on the surrounding configuration of minima and saddles. It is even possible that a maximum is isolated, without a path ending at the vertex. But this can only happen if the manifold is a sphere and f has one minimum, one maximum, and no saddles. It is not difficult to see that the paths consisting of the marked edges and vertices cut the 2-manifold into open disks, one for each minimum. This is also true if we have no saddles and therefore no marked edges. The open disk is then the sphere minus the maximum, which is marked by the algorithm.

Cleaning up. The Watershed Algorithm is widely used, but it tends to overdo the segmentation, creating too many regions and identifying small noise in the image rather than just the desired features. For this reason, there is always a clean-up step, sometimes done systematically and sometimes in an ad hoc way or even manually. This is illustrated in Figure IX.9, where the goal of the segmentation is to identify the location and the shape of cells of a fly embryo. To appreciate the importance of a reliable and consistent segmentation, we note that the image shows a cross-section of the embryo and similar images are taken at other cross-sections. Furthermore, the images are taken in a time series. After segmenting each cross-section, the task is to connect the results to reconstruct the 3-dimensional cells and then the cells to reconstruct the motion. Finally, the details of the motion are used as cues to hypothesize the forces that drive the motion. We can clearly see that the segmentation in Figure IX.9 at the top is too fine to capture individual cells. We need to simplify the segmentation by distinguishing more from less important separations. Persistence gives us just the tool we need for this.

Simplification. We begin by computing the persistence of the minima, saddles, and maxima, which can be done during the same bottom-up sweep that constructs the segmentation. We get infinite persistence for the critical vertices, giving birth to essential homology classes, and finite, positive persistence for all other critical vertices. For simplicity, assume that all saddles are 1-fold and thus get assigned a unique persistence value, the absolute height difference to the paired minimum or maximum.

We simplify the segmentation in the order of increasing persistence. Assuming the absolute height differences of the vertex pairs computed by the Persistence Algorithm are distinct, the pair with minimum persistence is unique. Consider first the case in which this pair consists of a minimum, x, and a saddle, y. Passing y during the sweep merges the component started at x with another component

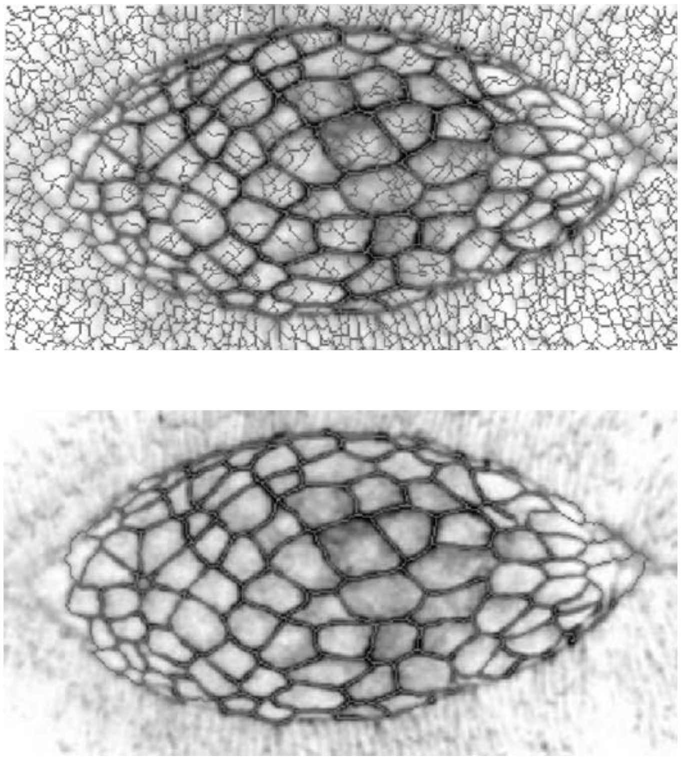

Figure IX.9: Top: the initial watershed segmentation before clean-up. Bottom: the result of simplifying the segmentation using persistence.

started earlier. By minimality of persistence, the watershed line started at y is part of the boundary of the region of x. Therefore, we can simplify by removing this line, which we do in two passes, both beginning at y. Unmarking edges and vertices in sequence, we stop each pass when we reach a maximum or the line merges with another. The result is that the two regions separated by y are now joined, and since x was the minimum of one of these, the other minimum represents the merged region. Consider second the case in which the minimum persistence pair consists of a saddle, y, and a maximum, z. By minimality of persistence, the watershed line started at y ends at z on one side and at a higher maximum on the other side. If this is the only watershed line ending at z, we remove it by unmarking its vertices and edges in sequence, beginning at z. Otherwise, we let the watershed line be, except that we think of it as an extension of the other lines ending at z.

We note that the change effectively treats the critical vertices of the minimum persistence pair as if they were regular. The rest of the persistence pairing remains unchanged. We can therefore proceed to the next lowest persistence pair and continue until we exceed a pre-chosen threshold. Applying this strategy to the segmentation in Figure IX.9, top, we get the segmentation shown at the bottom.

Notes. Images are generated by a plethora of technology, including microscopy [42] and magnetic resonance [81]. The 2-dimensional images in this section are from work on the dorsal closure in fly embryos [91]. Algorithms for processing images are described in the image analysis literature [135]. Many problems in this area are of a topological nature [93], including the segmentation of images into regions of interest. The Watershed Algorithm for segmentation goes back to the early eighties of last century [18]; see [128] for a survey of the general literature on the topic. Because of the importance and the large amount of medical data, the 3-dimensional version of the algorithm is of particular interest. We refer to [131] for the description of the method for magnetic resonance images using a diffusion filter to cope with the endemic over-segmentation. The algorithm for 3-dimensional images is similar to but more complicated than for 2-dimensional images. From Morse theory, we know that we have four types of simple critical vertices: minima, index-1 saddles, index-2 saddles, and maxima. We get a 3-dimensional cell for each minimum, a surface for each index-1 saddle, a curve for each index-2 saddle, and a vertex for each maximum. Together, they form a complex akin to the unstable manifolds discussed in Chapter VI. Persistence pairs minima with index-1 saddles, index-1 with index-2 saddles, and index-2 saddles with maxima. The simplification can again be done in the order of increasing persistence, but this is now more complicated than for 2-manifolds.

IX.4 Homology for Root Architectures

In this section, we look at the problem of recovering the structure of a plant root from photographic images. We combine standard image processing techniques with homology computations to capture interesting traits, such as the branching pattern and the distribution in space.

Background. Plant biologists understand much more about how plants grow and develop above the ground than underground. Yet, the root is every bit as important in how a plant responds to environmental variation and how it adapts to soil and moisture conditions. Learning about root architecture beyond what we know today is necessary before we can begin to understand the connection between phenotype and genotype in root development. The genotype is studied in a variety of biological experiments, many involving microarrays used to measure the expression of an entire collection of genes. To characterize the phenotype, we need an accurate set of measurements, preferably obtained without moving or damaging the plant. This way we can repeat the measurements during development, while the root makes decisions about where and when to grow.

We focus here on topological features of a plant root, in particular on a decomposition into tips, forks, and branches and on a characterization of space utilization. At a fork, a growing root either divides into two or a lateral root emerges. If we remove the fork, the rest of the root is branches, and we call the end of a branch that is not a fork a tip. Plant roots grow from the tip, so the number and location of these is of importance to biologists. Note, however, that the location of the forks and tips along the root says little about the way the root distributes itself in the soil. To study this distribution, we consider the complement of the root embedded in space and measure its connectivity using persistent homology.

Technology. For simple and rather obvious reasons, studying the architecture of a plant root is a difficult undertaking. We can dig up the plant to measure traits like length, branching, and more, but there are limitations to this approach. The first is that removing a plant from the ground seriously disrupts its growth pattern and may even kill it. A second is a lack of information about space utilization. We would like to find out how the root distributes itself in the soil and how its growth varies in response to a variety of stimuli, such as soil nutrient abundance and distribution, other forms of environmental stress, and the availability of water.

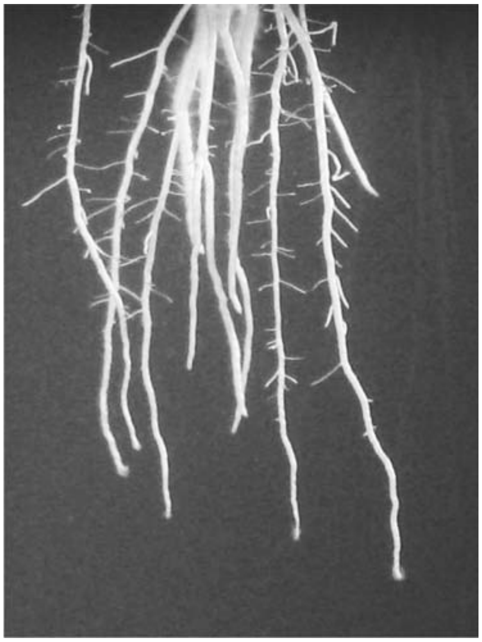

Figure IX.10: Rice root system growing in gel [image courtesy of Philip N. Benfey and Anjali Iyer-Pascuzzi, Biology, Duke University].

To cope with these difficulties, we need a new medium to grow the plant and a way to image the root. We know of two solutions: growing the plants in styrofoam containers and taking x-ray images, and growing them in transparent gel and taking photographs. The styrofoam provides a fairly realistic medium for the plant and both nutrition and water conditions are easy to vary without disturbing the plant. The disadvantage is the need to vacuum the container to remove as much water as possible before taking images. Water is an issue because x-rays refract when they pass through water, so any moisture left in the container shows up as noise in the image. The problem is severe since vacuuming disturbs the plant while moisture compromises the images. The transparent gel provides a nutrient mix that is less realistic than that of the styrofoam but much more than that of a hydroponic system, for example. The main advantage is the ease with which we can take photographic images; see Figure IX.10. Placing the gel together with the root inside a glass container, we can take photographs 360 degrees around. The task thus reduces to extracting the desired information from a 2-parameter sequence of images, going around the root and taking the photographs over a period of a few weeks. Each image is a 2-dimensional array of pixels. We discuss the extraction of features directly from these images as well as attempts to reconstruct a 3-dimensional image from the sequence of 2-dimensional images.

Tips, forks, and branches. Suppose first that we are working with a single photographic image, that is, a projection of our root to the plane represented by an array of pixels, p, with intensities, $f(p)$. Specifying a threshold, θ, we decompose the image into *foreground*, the union $\mathbb{Y}$ of all pixels with intensity $f(p) \geq \theta$, and *background*, the union $\mathbb{X}$ of pixels with intensity $f(p) < \theta$. We recall that each pixel is a closed square so that foreground and background are both closed and intersect in their common boundary, which is 1-dimensional.

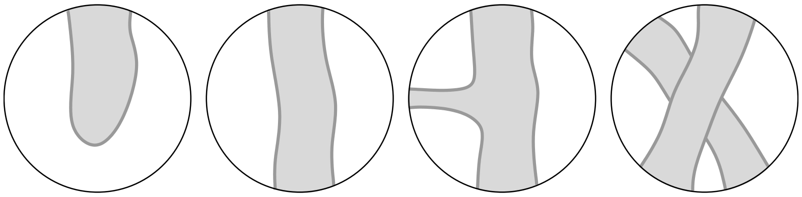

Figure IX.11: From left to right: schematic local pictures of a tip, a branch, a fork, and a crossing.

Assuming the foreground represents the root, it consists of streams of pixels forming roads that fork and cross and eventually end. While the roads vary in their thickness, we think of them as forming a 1-dimensional graph, with nodes connected by arcs. We call a degree-1 node a *tip*, a degree-3 node a *fork*, and an arc between two such nodes a *branch*. We illustrate the definitions by showing typical local pictures in Figure IX.11. There are also degree-4 intersections, but since roots rarely sprout two lateral branches at the same location, we assume these are artifacts of occlusion and represent them as crossings between branches rather than nodes in the graph. Of course, there can be more complicated situations caused by accumulated occlusion. These are better dealt with in three dimensions, and we ignore them for the time being.

Persistent local homology. As suggested by Figure IX.11, we look at the homology of the foreground and the background within small circular windows to classify a pixel to belong to a tip, a fork, or a branch. Fixing a point $x \in \mathbb{R}^2$ and a real number $r \geq 0$, we write $B(r) = B_x(r)$ for the closed disk with center x and radius r. The foreground and background within this window are $\mathbb{Y}(r) = \mathbb{Y} \cap B(r)$ and $\mathbb{X}(r) = \mathbb{X} \cap B(r)$. We are interested in the first homology of the foreground within the window relative to its boundary on the circle, $\mathsf{H}_1(\mathbb{Y}(r), \operatorname{bd} B(r))$. Assuming the generic case in which the circle intersects the boundary of $\mathbb{Y}$ transversally, we replace the boundary on the circle by the boundary $\mathbb{Y}(r)$ shares with $\mathbb{X}(r)$. Since $\mathbb{Y}(r)$ is in the plane, it is easy to see that this gives an isomorphic first homology group; compare with Exercise 7 at the end of this chapter. We thus get

$$\begin{aligned} \mathsf{H}_1(\mathbb{Y}(r), \operatorname{bd} B(r)) &\simeq \mathsf{H}_1(\mathbb{Y}(r), \mathbb{Y}(r) \cap \mathbb{X}(r)) \\ &\simeq \mathsf{H}_1(B(r), \mathbb{X}(r)) \\ &\simeq \tilde{\mathsf{H}}_0(\mathbb{X}(r)), \end{aligned}$$

where we get the second line by excision and the third line using the exact sequence of the pair $(B(r), \mathbb{X}(r))$. Instead of looking at the first relative homology group, we can therefore use the zeroth absolute homology group of the local complement, $\mathbb{X}(r)$, to distinguish between the different types of neighborhoods; see Table IX.1. Using persistence, we eliminate the dependence on the choice of r. Specifically, we increase r from zero to infinity and consider the zeroth persistence diagram of the local background pictures, $\mathbb{X}(r) \subseteq \mathbb{X}(s)$ for $0 \leq r \leq s < \infty$. If x is part of a tip, we see the following typical behavior as we grow the window.

	tip	branch	fork	crossing
rank $\mathsf{H}_0(\mathbb{X}(r))$	1	2	3	4

Table IX.1: The rank of the zeroth homology groups of the neighborhoods of a pixel inside a tip, a branch, a fork, and a crossing.

Tip. For very small r, $\mathbb{X}(r)$ will be empty. Its first component will be born when r reaches the distance from x to $\mathbb{X}$. This might be the only event for a while, but more likely we will see births and deaths of components in quick succession. However, these extra components correspond to points in the diagram whose persistence is negligible. Of course, once r gets large, all kinds of things may happen. In summary, we see only one birth that happens for small r and whose persistence is not negligible.

Similarly, for a branch we see two births for small r with larger than negligible persistence, for a fork we see three such births, and for a crossing we see four. As one can imagine, using persistence instead of a fixed radius greatly increases the number of pixels that can be correctly classified, but it is still a long shot from classifying all pixels. Ambiguities arise for a variety of reasons, including spurious foreground and background components, thicker than expected branches, and other root portions reaching into the local window. We can conceive of heuristics coping with these difficulties, but ultimately we need to face the fact that the problem as described does not admit a perfect solution.

3-dimensional reconstruction. Important information about the root, including estimates for the number of tips and forks, can be computed directly from the 2-dimensional images. However, to learn how the root distributes itself to explore the soil requires a reconstruction as a subset of 3-dimensional space. We describe here an image processing approach to this problem. It starts with a cubic block of voxels from which the algorithm sculpts the root. We assume a small, but not too small, number of photographic images taken of the same root from different directions and at about the same time. For each image, we know the position of the camera and the direction of the projection. As before, we use a threshold to decompose an image into foreground and background. If the projection of a voxel into the plane of the image lands inside the background, then the voxel cannot be part of the root. As illustrated in Figure IX.12, the Space Carving Algorithm combines the information gleaned from all 2-dimensional images and this way arrives at a first approximation of the 3-dimensional structure.

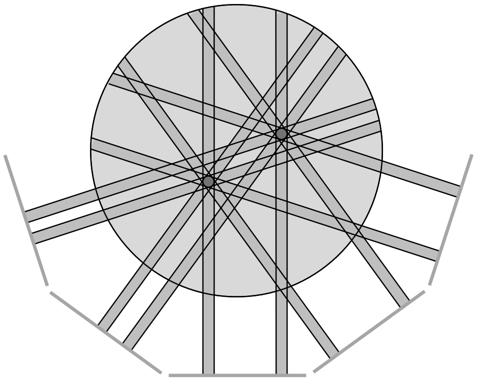

Figure IX.12: Schematic cross-section of a root growing inside a cylindrical container. We reconstruct the two shaded spots from their images in the five projections.

The quality of the reconstruction depends on the number and resolution of the images, the calibration of the cameras, and other factors. As before, we can make amendments to the algorithm to improve the quality, such as estimating probabilities for a voxel to belong to the root or using prior knowledge about the structure of the root. While we can perhaps reach acceptable results, we should keep in mind that perfection is at best reachable in the limit of our improvement efforts.

Utilization of space. Suppose now that we have reconstructed the 3-dimensional structure of the root. Reusing the 2-dimensional notation, we write $\mathbb{Y} \subseteq \mathbb{R}^3$ for the space we use to represent the root. We may revisit the decomposition of the root into tips, forks, and branches using local homology within spherical balls. Since crossings no longer confuse the picture, we can expect a higher success rate in the classification. We can also address the global question of how the root distributes itself in space. For this purpose, we introduce the Euclidean distance function,

$f : \mathbb{R}^3 \to \mathbb{R}$, defined by mapping every point $x \in \mathbb{R}^3$ to its distance from $\mathbb{Y}$, that is, $f(x) = \inf_{y \in \mathbb{Y}} \|x - y\|$. Note that $\mathbb{Y} = f^{-1}(0)$. The sublevel sets of f form a nested sequence of spaces, $\mathbb{Y}_r = f^{-1}[0, r]$. The corresponding sequence of reduced homology groups,

$$0 \to \tilde{\mathsf{H}}_p(\mathbb{Y}_0) \to \ldots \to \tilde{\mathsf{H}}_p(\mathbb{Y}_r) \to \ldots \to 0,$$

characterizes how thickening up the root fills space. As described in Chapter VII, we use the persistence diagram to characterize the main events in the filtration. The root is connected, so the zeroth diagram, $\mathrm{Dgm}_0(f)$, should be empty. Any deviation from this ideal will have to be explained by failures to accurately reconstruct the root. There is more interesting information in the first and second diagrams. Consider for example a diffuse root system, that is, a root that distributes itself reasonably densely and more or less uniformly in the available space. Then $\mathbb{Y}_r$ will have trivial first and second homology groups already for small values of r. Correspondingly, $\mathrm{Dgm}_1(f)$ and $\mathrm{Dgm}_2(f)$ will have no points of larger than negligible persistence. On the other hand, if the root has a tendency to grow around pieces of space, then this will express itself in voids and tunnels of $\mathbb{Y}_r$. Correspondingly, one of the two or both diagrams will have points with larger than negligible persistence. We note that this discussion neglects the possibility of a root that grows radially in a non-uniform manner but avoids the creation of tunnels and voids while exploring space. But we can detect such behavior by considering spherical cross-sections, for example.

There is more than one way we can compute the persistence diagrams of f. For example, we can grow $\mathbb{Y}_r$ by successively adding voxels to the initial space, $\mathbb{Y} = \mathbb{Y}_0$. Alternatively, we may let $S \subseteq \mathbb{R}^3$ be the set of centers of the voxels constituting $\mathbb{Y}$. We then compute the Delaunay complex of S and the family of alpha complexes, as explained in Chapter III. The Stability Theorems of Chapter VIII imply that the diagrams we get from these and reasonable other methods are only a small bottleneck distance away from each other.

Notes. The study of plant roots has a long tradition in biology [26]. The project that provides the background for the discussions in this section targets agricultural plants, such as rice and maize. Local homology is a natural choice in the study of structural features such as forks and tips in roots. In mathematics, the concept makes its first appearance in Poincaré duality. Letting $\mathbb{M}$ be a d-dimensional manifold without boundary and x a point of $\mathbb{M}$, we find that the relative homology group $\mathsf{H}_p(\mathbb{M}, \mathbb{M} - \{x\})$ is trivial for all dimensions $p \neq d$ and has rank one for $p = d$. Using integer coefficients, the latter group has two generators, and a choice of one is called an orientation of $\mathbb{M}$ at x. Making consistent choices at all points, an element of $\mathsf{H}_d(\mathbb{M})$ is a fundamental class of $\mathbb{M}$ if its image under the induced map to $\mathsf{H}_d(\mathbb{M}, \mathbb{M} - \{x\})$ is the chosen orientation. This class is used to define Poincaré duality; see Hatcher [82, Section 3.3]. The idea of using this construction in combination with persistent homology appears for the first time in [17]. Given a finite point set $S \subseteq \mathbb{R}^d$, the paper uses persistent versions of local homology towards reconstructing a stratified space that best approximates S. Basic tools in this study are the persistent kernels and cokernels of maps from one filtration to another. The algorithms for these are similar to but more involved than those for ordinary persistence [37].

The problem of reconstructing shapes from sequences of images is studied in the general field of computer vision; see e.g. [72, 86]. The idea of carving out the shape from a block of voxels is due to Kutulakos and Seitz [98], but see also [100]. Given such a reconstruction, we can use standard algorithms for Delaunay complexes [54], alpha shapes [63], and persistent homology [60] to characterize the distribution of the root in space.

Exercises

The credit assignment reflects a subjective assessment of difficulty. A typical question can be answered using knowledge of the material combined with some thought and analysis.

1. **Antipodal functions** (two credits). A function $f : \mathbb{S}^1 \to \mathbb{R}$ is *antipodal* if $f(s) = f(-s)$ for all $s \in \mathbb{S}^1$. Equivalently, the function defined by $g(s) = f(s) - f(-s)$ is the zero function.

   (i) Design a measure for quantifying the distance of a function from being antipodal.

   (ii) Prove that your measure is stable or, alternatively, change your measure such that it is stable.

2. **Lipschitz function on the sphere** (three credits). Let d be a positive integer constant and $f : \mathbb{S}^d \to \mathbb{R}$ a Lipschitz function. Note that the d-dimensional volume of $\mathbb{S}^d$ is bounded from above by a constant that depends on d.

   (i) Prove that for $q > d$, the degree-q total persistence of f is bounded from above by a constant.

   (ii) Use the result in (i) to show that for $q > d + 1$, the degree-q total persistence measuring Lipschitz functions on $\mathbb{S}^d$ is stable.

3. **Fast pairing** (two credits). Let $f : \mathbb{S}^1 \to \mathbb{R}$ be a continuous, piecewise linear function specified by its values at the vertices of a triangulating n-gon.

   (i) Assuming $f(x_i) \neq f(x_j)$ for all pairs of vertices $x_i \neq x_j$ of the n-gon, characterize the vertices that are paired by extended persistence.

   (ii) Furthermore assuming the vertices are given in the order of increasing function value, show that the extended persistence pairing can be computed in time at most some constant times n.

4. **Inflection points and bitangent lines** (one credit). Let $\gamma : \mathbb{S}^1 \to \mathbb{R}$ be a smooth embedding of the circle in the plane. Suppose the curvature vanishes only at a finite number of points. Show that there are only a finite number of lines that are tangent to the curve at two or more points.

5. **Labeling regions** (two credits). Consider the Watershed Algorithm for segmenting a triangulated 2-manifold given in Section IX.3.

   (i) Modify the algorithm so it labels the simplices in each region with the index of the generating maximum.

   (ii) Define the *i-th region* as the union of the interiors of the simplices labeled i by the modified Watershed Algorithm. Prove that it is homeomorphic to an open disk.

6. **Ordering the pixels** (two credits). Let $n = 2^k$ and consider an n-by-n array of pixels p_i^j for $1 \leq i, j \leq n$. We define what it means to list the pixels in *Z-order*. For $k = 1$, we have four pixels which we arrange as $p_1^1, p_1^2, p_2^1, p_2^2$. For $k > 1$, we decompose the array into four equal blocks and list the upper left, the upper right, the lower left, the lower right blocks in this sequence and each in Z-order.

   (i) Assume the pixels are listed in lexicographic ordering of their index pairs, (i, j). Write an algorithm that rearranges the pixels in Z-order.

   (ii) Write computer programs that translate back and forth between the row-column index pairs of a pixel and its index in Z-order.

7. **Isomorphic relative homology groups** (three credits). Let $\mathbb{Y}$ be a d-manifold with boundary and let $\mathrm{bd}\,\mathbb{Y} = \mathbb{A} \cup \mathbb{B}$ be a decomposition of the boundary into two $(d-1)$-manifolds with common, $(d-2)$-dimensional boundary and disjoint interiors. Prove that $\mathsf{H}_{d-p}(\mathbb{Y}, \mathbb{A}) \simeq \mathsf{H}_p(\mathbb{Y}, \mathbb{B})$ for all dimensions p.

8. **Distance function** (two credits). Let S be a finite set of points in $\mathbb{R}^d$ and let $f : \mathbb{R}^d \to \mathbb{R}$ be defined by $f(x) = \min_{u \in S} \|x - u\|$. Recall from Chapter III that Alpha(r) is the alpha complex defined by S and a radius $r \geq 0$.

   (i) Let $r \leq s$ and consider the diagram of homology groups in which all four maps are induced by inclusion:

$$\begin{tikzcd}
\mathsf{H}_p(f^{-1}[0,r]) \arrow[r] & \mathsf{H}_p(f^{-1}[0,s]) \\
\mathsf{H}_p(\mathrm{Alpha}(r)) \arrow[r] \arrow[u] & \mathsf{H}_p(\mathrm{Alpha}(s)). \arrow[u]
\end{tikzcd}$$

   Prove that the vertical maps are isomorphisms and that the diagram commutes.

   (ii) Show that the persistence diagrams of f are the same as those of the sequence of alpha complexes.