Merge trees in discrete Morse theory

Benjamin Johnson, Nicholas A. Scoville

July 2020

Abstract

In this paper, we study merge trees induced by a discrete Morse function on a tree. Given a discrete Morse function, we provide a method to constructing an induced merge tree and define a new notion of equivalence of discrete Morse functions based on the induced merge tree. We then relate the matching number of a tree to a certain invariant of the induced merge tree. Finally, we count the number of merge trees that can be induced on a star graph and characterize the induced merge tree.

1 Introduction

Topological data analysis seeks to understand a set of data by studying topological properties of that data. One highly successful tool in this regard is persistent homology. Persistence has been used to study statistical mechanics \[20\], hypothesis testing \[6\], image analysis \[7\], complex networks \[19\], and many other phenomena. Recently, there has been interest in studying merge trees, a special kind of persistence \[15, 17\]. Part of the advantage of studying a merge tree instead of the persistence diagram is that the merge tree gives more detailed information about precisely which components merged with which other components. It tracks not only the lifetime of a component but its evolution as well.

In \[9\], Justin Curry studies functions on the unit interval that have the same persistent homology. In this smooth setting, Curry develops a merge tree associated to a Morse set, an abstraction of path components associated to a Morse function on a compact, connected manifold. He is then able to count merge trees under a suitable notion of equivalence. In this paper, we take up a similar problem in a purely discrete setting: that is, given a discrete Morse function on a tree (i.e. 1-dimensional abstract simplicial complex), we associate a tree, appropriately called a merge tree (Definition 9). We describe a method to obtain a merge tree from a discrete Morse function on a tree in Theorem 9. After defining a notion of equivalence of merge trees, we prove that a certain invariant of an induced merge tree of $T$ yields a lower bound for the matching number of $T$ in Proposition 13. Section 4 is devoted to comparing merge equivalent discrete Morse functions with other notions of equivalence of discrete Morse functions. Then in Section 5, we give a characterization of merge trees induced
by a discrete Morse function on a star graph. Finally, we share some future
directions in Section 6.

2 Background

2.1 Graphs and trees

Let $G = (V(G), E(G))$ be a finite, loopless graph without multi-edges (i.e. a
1-dimensional abstract simplicial complex). We call an edge or a vertex of $G$ a
simplex. If $e = uv$ is an edge, we say that the edge $e$ is incident with vertex $v$
and that $u$ and $v$ are adjacent. We use $|V(G)|$ to denote the number of vertices
of $G$ and $|E(G)|$ to denote the number of edges of $G$.

We work exclusively with trees in this paper. Here we recall several impor-
tant characterizations of trees. They will be utilized without further reference.

**Theorem 1.** (Characterization of trees) Let $G$ be a connected graph with $v$
vertices and $e$ edges. The following are equivalent:

a) Every two vertices of $G$ are connected by a unique path.

b) $v = e + 1$.

c) $G$ contains no cycles.

d) $b_1(G) = 0$ where $b_1$ is the first Betti number of $G$ ([10, Chapter II.4]).

e) The removal of any edge from $G$ results in a disconnected graph.

A connected graph that satisfies any of the above characterizations is called a
tree. Proofs of the equivalence of the statements may be found in any graph
theory textbook (e.g. [8, Chapter 2.2]). A disconnected graph $F$ such that each
component of $F$ is a tree is called a forest. For any vertex $v \in F$, we let $F[v]$ denote
the connected component of $F$ containing $v$. It immediately follows that
if $F$ is a forest with two distinct vertices $u, v \in F$, then there is a path between
two vertices $u$ and $v$ if and only if $F[u] = F[v]$.

2.2 Discrete Morse theory

Our references for the basics of discrete Morse theory are [11, 13, 14, 18]. There
are several different ways of viewing a discrete Morse function. For our purposes,
we make the following definition:

**Definition 2.** Let $G$ be a graph. A function $f : G \to \mathbb{R}$ is called weakly
increasing if $f(v) \leq f(e)$ whenever $v \subseteq e$. A discrete Morse function
$f : G \to \mathbb{R}$ is a weakly increasing function which is at most 2–1 and satis-
ifies the property that if $f(\sigma) = f(\tau)$, then either $\sigma \subseteq \tau$ or $\tau \subseteq \sigma$. Any simplex $\sigma$ on
which $f$ is 1–1 is called critical and the value $f(\sigma)$ is a critical value of $f$. 

2
Example 3. Define the function $f$ on $T$ as follows:

Then $f$ is a discrete Morse function. Note that all values are critical.

Definition 4. Let $G$ be a graph. Given $a \in \mathbb{R}$ the level subcomplex $G_a$ is defined to be the induced subgraph of $G$ consisting of all simplices $\sigma$ with $f(\sigma) \leq a$. For each critical value $c_0 < \ldots < c_{m-1}$ of $f$, we consider the induced sequence of level subcomplexes $\{v\} = G_{c_0} \subset G_{c_1} \subset \ldots \subset G_{c_{m-1}}$. In the sequel, we will use the notation $G_{c_i - \epsilon}$ to denote the level subcomplex immediately preceding $G_{c_i}$; that is, $\epsilon$ is chosen so that $f(\sigma) < c_i - \epsilon < c_i$ for every $\sigma \in G$ such that $f(\sigma) < c_i$.

3 Merge trees

In this section we introduce merge trees, our main object of study.

3.1 Basics of merge trees

Definition 5. A binary tree is a rooted tree where each vertex has at most two children, and each child is designated as its left (L) or right (R) child. A binary tree is full if every vertex has 0 or 2 children. A merge tree is a full binary tree. A node with exactly one neighbor is a leaf node or leaf. Otherwise, a node with more than one neighbor is an internal node.

Although graphs and merge trees are different objects, they both look the same and consist of vertices and edges. To help distinguish them, we reserve the term "node" for merge trees and "vertex" for graphs.

Remark 6. We will view a merge tree upside down from how one normally views a binary tree; that is, the root is drawn at the bottom of the tree, as opposed to the top. However, we maintain the parent/child relationship language so that when considering the merge tree, a child will be above the parent.

Example 7. Consider the merge tree below:
To illustrate Remark 6, we say that $a$ and $b$ are children of $x$, even though $x$ is below $a$ and $b$.

**Remark 8.** Note that the left and right information is part of the definition of a merge tree so that

![Diagram showing different merge trees]

are different merge trees even though they are isomorphic as graphs. We will sometimes suppress the L and R labelings below, as the position on the page makes a node’s L or R status clear.

To any discrete Morse function on a tree, we are able to associate a merge tree through the following construction.

**Theorem 9.** Let $f: T \to \mathbb{R}$ be a discrete Morse function on a tree $T$. Then $f$ induces a merge tree $M_f = M$.

**Proof.** Let $f: T \to \mathbb{R}$ be a discrete Morse function with critical values $c_0 < c_1 < c_2 < \ldots < c_m$ and write $C := \{c_0, c_1, \ldots, c_m\}$. We will construct a merge tree whose node set is in 1–1 correspondence with $C$. In order to organize information, we will label each node of the merge tree by defining a function $f_M$ that takes in nodes of $M$ and yields real numbers. Furthermore, each node will be given a direction L or R. We construct $M$ inductively on the critical edges of $f$ in reverse order.

Begin by creating a node $n_{c_m}$ corresponding to the critical edge in $T$ labeled $c_m$. Define $f_M(n_{c_m}) := c_m$ along with direction L.

Inductively, let $n_{c_i}$ be a node of $M$ corresponding to a critical edge $uv \in C$. Create two child nodes of $n_{c_i}$ called $n_u$ and $n_v$. Define $f_M(n_u) := \max\{f(\sigma): \sigma \in T_{c_i-\epsilon}[u], \sigma \text{ critical}\}$ and $f_M(n_v) := \max\{f(\sigma): \sigma \in T_{c_i-\epsilon}[v], \sigma \text{ critical}\}$ (see Definition 4 for meaning of $T_{c_i-\epsilon}$). Note that the values of the child nodes can be values from a critical vertex or a critical edge. If $\min\{f(\sigma): \sigma \in T_{c_i-\epsilon}[u]\} <$
\[
\min\{f(\sigma): \sigma \in T_{c_i-\epsilon}[v], \sigma \text{ critical}\}, \text{ then give } n_{n_i} \text{ the same direction (L or R) as that of } n_{c_i} \text{ and give } n_v \text{ the opposite direction.}
\]

Continue over all critical edges to obtain \( M \).

It can be difficult to build the induced merge tree starting from the “top down” or the smallest value of the discrete Morse function since when adding new nodes to the merge tree, it is often unclear where a node is placed on the merge tree. This is because where it is placed depends on which component(s) it ends up merging to and when. Fortunately, Theorem 9 is starting from the “bottom up” or the largest value of the discrete Morse function. We give an example below.

**Example 10.** To illustrate the construction of Theorem 1, we will take the discrete Morse function from Example 3. We begin by identifying the critical edge values and placing them in reverse order: 10, 9, 8, 5, 3. The largest value is 10, so it corresponds to a node in \( M \) with label 10 and direction L:

\[
10L
\]

We then look at the level subcomplex \( T_{10-\epsilon} \) and identify the largest value in each of the trees that were incident with 10.

\[
\begin{array}{c}
1 \\
8 \\
9 \\
0
\end{array}
\quad
\begin{array}{ccc}
3 \\
6 \\
5 \\
4
\end{array}
\]

In this case, the two values are 3 and 9. To determine which is to the left and which is to the right, we look for the tree with the minimum value. In this case, 0 < 1 so that 9 shares the same direction as 10. We thus obtain

\[
9L
\quad
3R
\]

\[
10L
\]

\[
5
\]
We move next to 9, and consider the level subcomplex $T_{9-\epsilon}$:

Now 9 was connected to 6 and 0, and the maximum value on each of their trees is 8 and 5, respectively, so these will be the labels of the two new nodes above 9. To see which one shares the direction with 9, we see that the tree with the vertex 0 has minimum value, so 5 shares the same direction as 9. We then obtain

Now in $T_{8-\epsilon}$, 8 was connected to the isolated vertex 7 and isolated vertex 6. Hence the two new nodes connected to 8 will be 6 and 7. Since $6 < 7$, 6 and 8 share the same direction yielding
The next critical edge value is 5, so we consider the level subcomplex $T_{5-\epsilon}$:

```
\begin{center}
\begin{tikzpicture}
\node (1) at (0,0) {$1$};
\node (2) at (2,0) {$2$};
\node (3) at (1,-1) {};\node (4) at (-1,-1) {3};
\draw (1) -- (2);
\end{tikzpicture}
\end{center}
```

The edge 5 was connected to isolated vertices 0 and 4, yielding

```
\begin{center}
\begin{tikzpicture}
\node (0) at (0,0) {0};\node (4) at (2,0) {4};
\node (5) at (0,-1) {};\node (6) at (-2,-1) {};\node (7) at (1,-1) {5};\node (8) at (2,-1) {6};\node (9) at (3,-1) {7};\node (10) at (0,-2) {8};\node (11) at (2,-2) {9};\node (12) at (4,-2) {10};
\draw (0) -- (5);\draw (5) -- (9);\draw (5) -- (6);\draw (6) -- (4);\draw (12) -- (9);\draw (8) -- (11);\draw (7) -- (12);\draw (7) -- (11);\draw (11) -- (4);
\end{tikzpicture}
\end{center}
```

Finally, 3 is connected to 2 and 1, giving us the merge tree induced by the discrete Morse function:

```
\begin{center}
\begin{tikzpicture}
\node (0) at (0,0) {0};\node (4) at (2,0) {4};\node (9) at (0,-1) {};\node (10) at (-2,-1) {};\node (7) at (1,-1) {7};\node (6) at (2,-1) {6};\node (8) at (3,-1) {8};\node (2) at (4,-1) {2};\node (1) at (5,-1) {1};\node (3) at (0,-2) {3};
\draw (0) -- (9);\draw (9) -- (10);\draw (9) -- (7);\draw (7) -- (4);\draw (7) -- (8);\draw (8) -- (6);\draw (8) -- (2);\draw (8) -- (1);\draw (7) -- (10);
\end{tikzpicture}
\end{center}
```
Definition 3.1. Two discrete Morse functions $f, g: T \to \mathbb{R}$ are merge equivalent if they induce the same unlabeled binary tree; that is, if there is a rooted graph isomorphism $\phi: M_f \to M_g$ such that node $v$ is a left (right) child of node $u$ if and only if node $\phi(v)$ is a left (right) child of node $\phi(u)$.

We will compare this notion of equivalence of discrete Morse functions with other notions of equivalence of discrete Morse functions in Section 4.

In addition, our main goal of Section 5 will be to count the number of merge equivalent discrete Morse functions on a star graph.

Remark 11. Definition 3.1 defines two merge trees to be equivalent if they share the same tree structure, ignoring the lifespan of a component and the order in which components were born and died. One could define a notion of equivalence that takes this order into account, thereby defining a chiral merge tree. This was defined and studied in the smooth setting by Curry [9]. We pose this adaption in the discrete setting as an open problem in Section 6.

3.2 Relation to Matching number

Recall that a matching in a graph is a set of edges such that no two edges share a common vertex. A matching is said to be maximum if it is a matching that contains the largest possible number of edges. The matching number of $G$, denoted $\nu(G)$, is the size of a maximum matching. We give a relationship between the matching number of a tree $T$ and the induced merge tree of any discrete Morse function on $T$ in Proposition 13. First a definition.

Definition 12. Let $M$ be a merge tree. An internal node of $M$ that is adjacent to exactly two leaves is called an impasse. The value $i(M)$ is the number of impasses of $M$.

Lemma 3.1. Every merge tree with more than one vertex has at least one impasse. That is, $i(M) \geq 1$ for every merge tree $M$.

Proof. Suppose we have a merge tree $M$ without an impasse. Therefore, all internal nodes of $M$ must have at least one internal node as a child. Consequently, each of those internal nodes must now have an internal node as a child. This continues on indefinitely, contradicting the fact that $M$ is finite.

Proposition 13. Let $f: T \to \mathbb{R}$ be discrete Morse function, $M$ the induced merge tree of $f$. Then the set of edges of $T$ corresponding to the set of impasses of $M$ form a matching of $T$. In particular, $i(M) \leq \nu(T)$.

Proof. Let $x, y$ be two impasses of $M$. In particular, $x, y$ are not leaves and correspond to edges $e_x, e_y$, respectively, in $T$. We must show that $e_x$ and $e_y$ do not share a vertex. Suppose by contradiction that $e_x = uv$ and $e_y = uw$. Then the two leaves of $x$ must be nodes corresponding to $u$ and $v$, say $n_u$ and $n_v$. Likewise for $y$. But then $n_u$ is adjacent to both $x$ and $y$, contradicting the fact that $n_u$ is a leaf. Thus $e_x$ and $e_y$ do not share a vertex in common. It follows that the corresponding set of edges forms a matching, hence $i(M) \leq \nu(T)$. \qed

8
Example The inequality in Proposition 13 can be strict. Indeed, consider the following tree $T$ with discrete Morse function $f$.

Then $f$ induces the merge tree $M$ given by

But clearly $i(M) = 1 < 2 = \nu(T)$.

4 Comparison with other notions of equivalence

There are several other notions of equivalence of discrete Morse functions in the literature. In this section, we compare merge equivalence with these other notions.

4.1 Forman equivalence

Definition 4.1. Let $f$ be a discrete Morse function on $G$. The induced gradient vector field $V_f$ is defined by

$$V_f := \{(\sigma^{(p)}, \tau^{(p+1)}) : \sigma < \tau, f(\sigma) \geq f(\tau)\}.$$

Recall that two discrete Morse functions $f, g : G \rightarrow \mathbb{R}$ are Forman equivalent if and only if $V_f = V_g$ [3]. It is easy to see that neither Forman equivalence nor merge equivalence implies the other.

Example 14. Suppose we have the following discrete Morse functions:

$$\begin{array}{cccc}
0 & 3 & 1 & 4 & 2 \\
0 & 4 & 1 & 3 & 2
\end{array}$$
All simplices for both functions are critical, and hence the gradient vector field induced by both these functions has no arrows. Thus these functions are Forman equivalent. However, the merge trees are given by

and

respectively. Thus they are not merge equivalent.

4.2 Homological equivalence

Given a graph with a discrete Morse function, one may study the Betti numbers of the level subcomplexes induced by the critical values. This gives rise to a non-negative sequence of integers. Such a sequence is a homological sequence and two discrete Morse functions are homologically equivalent if they induce the same homological sequence. See [2, 5, 4, 1].

Example 15. We show that homological equivalence and merge equivalence do not imply each other. First, consider the first two discrete Morse functions from Example 14. It is easy to see that they both induce the homological sequence 1, 2, 3, 2, 1, hence they are homologically equivalent. However, their merge trees were shown in that same example to be different, hence they are not merge equivalent.

Now suppose we have the following functions

The homological sequence for these functions is given by 1, 2, 3, 2, 1 and 1, 2, 1, 2, 1, respectively so that they are not homologically equivalent. But they
both induce the merge tree

so that they are merge equivalent.

### 4.3 Persistence equivalence

Another notion of equivalence of discrete Morse functions, closely related to merge equivalence, is persistence equivalence. Two discrete Morse functions are persistent equivalent if they induce the same persistence diagram. See [16] for more details.

**Example 16.** Suppose we have the following discrete Morse functions:

These functions both create the same persistence diagram,

but different merge trees.

The same functions in Example 15 which show that merge equivalence does not imply homological equivalence also shows that merge equivalence does not imply persistence equivalence.

### 5 Merge tree of a star graph

**Definition 17.** Let $n \geq 2$ be an integer. The **star graph on $n$ vertices** is defined by $S_n = K_{1,n-1}$ ([12, p. 17]). We call the unique vertex $c \in S_n$ of degree $n - 1$ the **center of $S_n$$ or **center vertex**.
In this section, we consider discrete Morse functions on a star graph with every simplex critical. We will call such a function a critical discrete Morse function. We first define the kind of merge tree that, it turns out, can be induced by a critical discrete Morse function on a star graph.

**Definition 18.** A merge tree $M$ is called thin if $i(M) = 1$, i.e., $M$ has a unique impasse.

Observe that a thin merge tree can be characterized by the fact that it has a unique path from the root node to the unique impasse with the property that every edge not on the path and incident with the path is part of a leaf. We are thus able to determine a unique thin merge tree through a sequence of Ls and Rs where the L or R is specifying the next child to travel to, i.e., the direction that this path takes. We make this notion precise in the following definition.

**Definition 5.1.** Let $M$ be a thin merge tree with $n$ leaves, and let $P_M$ denote the unique path $r = m_1, m_2, \ldots, m_{n-1}$ from the root node $r$ to the unique impasse $m_{n-1}$. Define a function $d_M = d: \{m_1, \ldots, m_{n-2}\} \rightarrow \{L, R\}$ by $d(m_i) = L$ (or $R$) if $m_{i+1}$ is the left (or right) child of $m_i$, $1 \leq i \leq n-2$. The LR sequence of $M$, denoted $D_M = D$, is the sequence $D: \{0, 1, 2, 3, \ldots, n-2\} \rightarrow \{L, R\}$ by $D(0) = L$ and $D_M(i) := d(m_i)$ for all $1 \leq i \leq n-2$.

The choice that $D(0) = L$ is arbitrary, but chosen to be consistent with the construction in Theorem 9. We will also need this in the proof of Proposition 22 to determine when the sequence $D$ switches direction.

**Example 19.** We illustrate the terms and notation in Definition 5.1. Consider the merge tree $M$
Then \( d(m_1) = d(m_2) = d(m_4) = d(m_7) = L \) while \( d(m_3) = d(m_5) = d(m_6) = R \). We then have that the LR sequence of \( M \) is \((L)LLRLRRL\). Note that the 0 term of the sequence is always L, which we will put in parentheses when it is included.

By choosing a direction to travel, we easily count all thin merge trees with \( n \) internal nodes.

**Proposition 20.** There are exactly \( 2^n - 1 \) thin merge trees with \( n \) internal nodes.

**Proof.** We will count the number of ways to construct a thin merge tree with \( n \) internal nodes. Beginning with the root node, we will choose which of its neighbors, left or right, is the next internal node, forcing the other node to be a leaf, since by definition, a thin tree has only one internal node incident to two leaves with all other internal nodes being incident to one leaf and one internal node. At each internal node, we choose the next internal node to be on the left or the right. This choice is made for every internal node except for the very last internal node, which ends with the node adjacent to two leaves. Since this choice is made for all internal nodes except the impasse, there are exactly \( 2^n - 1 \) thin merge trees with \( n \) internal nodes.

The goal of the remainder of the section is to show that thin merge trees with \( n + 1 \) leaves are in bijective correspondence with the merge equivalent discrete Morse functions on \( S_n \). As promised, the next proposition tells us that the merge tree induced by a discrete Morse function on a star graph is always thin.
Proposition 21. If $T = S_n$ is a star graph and $f : S_n \to \mathbb{R}$ any discrete Morse function, then $i(M_f) = 1$.

Proof. By Proposition 13, $i(M_f) \leq \nu(S_n) = 1$. To see that we have equality, apply Lemma 3.1.

Finally, given a thin merge tree with $n$ internal nodes, we need to construct a critical discrete Morse function on a star graph whose induced merge tree is the given thin merge tree.

Proposition 22. Suppose $M$ is a thin merge tree. Then there is a star graph $S_n$ and a discrete Morse function $f : S_n \to M$ such that $M_f = M$.

Proof. Let $M$ be a thin merge tree with $n+1$ leaves and choose $T = S_n$. We construct $f : S_n \to \mathbb{R}$ such that $M_f = M$. Consider the path $r = m_1, m_2, \ldots, m_{n-1}$ from the root node $r$ to the unique impasse $m_{n-1}$ of $M$. Now let

$$m_{i_1}, m_{i_2}, \ldots, m_{i_k}$$

be the (possibly empty) nodes on the path such that $d_M(m_{i_j}) \neq d_M(m_{i_{j-1}})$, that is, the nodes in the path that switch direction. To keep track of the labeling and correspondence between $M$ and $S_n$, we will also label nodes of $M$. To that end, label the unique leaf of $m_{i_j}$ with label $j$ for $1 \leq j \leq k-1$ and either of the leaves of the unique impasse $m_{n-1}$ with label $k$. Now choose $k-1$ corresponding non-center vertices of $S_n$ and label them $1, 2, \ldots, k-1$. Choose the center vertex of $S_n$ to label $k$. Traversing the path from $m_{n-1}$ to $m_1$, label each unlabeled leaf $k+1, k+2, \ldots, k+\ell = n+1$ in order and label the remaining vertices of $S_n$ by $k+1, \ldots, k+\ell$. Finally, we know that each internal node corresponds to the edge in $S_n$. Traversing the path from $m_{n-1}$ to $m_1$ again, label in order each internal node $k+\ell+1, k+\ell+2, \ldots, k+\ell+d = 2n+1$. The node $m_k$ is labeled $k+\ell+1$ and has children labeled $k$ and $k+1$. Hence label the edge in $S_n$ connecting vertices $k$ and $k+1$ with label $k+\ell+1$. Continuing along the path $m_{n-1}$ to $m_1$, suppose we are at node $n_{k-i}$ with label $k+\ell+i$. Then $m_{k-i}$ is adjacent to a leaf. This leaf corresponds to a vertex in $S_n$ which is incident to a unique edge. Label this edge $k+\ell+i$. This completes the labeling of $S_n$. It remains to show that the labeling described above is a critical discrete Morse function whose induced merge tree is the given thin merge tree $M$. By construction, the vertices of $S_n$ are labeled first, and since each subsequent label is strictly greater than the previous label, each vertex has a value strictly less than each edge, and hence $f$ is a critical discrete Morse function. By Proposition 21, the merge tree induced by a discrete Morse function on a star graph is a thin merge tree. Hence it suffices to show that the LR sequence of $M$ agrees with the LR sequence of $M_f$, i.e., $D_M = D_{M_f}$.

We proceed inductively on the LR sequence, showing that $D_M(i+1) = D_M(i)$ if and only if $D_{M_f}(i+1) = D_{M_f}(i)$ for $0 \leq i \leq n-2$, i.e., the LR sequence for $M_f$ changes directions precisely when the LR sequence for $M$ does. By both constructions, the root node for both $M$ and $M_f$ is given direction $L$ by definition, i.e, $D_M(0) = D_{M_f}(0) = L$. Inductively, suppose the LR sequence
for $M$ agrees with the LR sequence for $M_f$ up to $i$, $0 \leq i \leq n - 2$, and consider the node $m_{i+1}$ on the path $P_M$. Let $e$ be the edge of $S_n$ corresponding to node $m_{i+1}$. Then $T_{f(e) - e}$ is a forest where the components of the endpoints of $e$ are a star graph $S_{i+1}$ and an isolated vertex $v$. Now this $v$ induces a leaf node in the construction of $M_f$. Hence the path $P(M_f)$ will follow the direction of the node whose label is the maximum value of the star graph $S_{i+1}$. By the construction of Theorem 9, the forest component with the smallest value will share the direction of the previous node. We proceed by cases.

**Case 1:** $D_{M_f}(i + 1) = D_{M_f}(i)$
Suppose $D_{M_f}(i + 1) = D_{M_f}(i)$. Then we need to show that there is a vertex of $S_{i+1}$ with label less than that of $v$. Since $d_M(m_i) = d_M(m_{i+1})$, it follows by construction that $m_{i+1}$ is not any of the $m_{ij}$ from Equation (1). Thus, by the labeling of $M$, the leaf node of $m_{i+1}$ is given a value $t$ which is strictly greater than the value of either leaf of the impasse $m_k$. But by construction, $f(v) = t$ and the values assigned to the leaf nodes of $m_k$ are assigned to corresponding vertices of $S_{i+1}$. We conclude that $D_{M_f}(i + 1) = D_{M_f}(i)$.

**Case 2:** $D_{M_f}(i + 1) \neq D_{M_f}(i)$
Now suppose that $D_{M_f}(i + 1) \neq D_{M_f}(i)$. We need to show that $f(v) < f(u)$ for all $u \in S_{i+1}$. Since $d_M(m_i) \neq d_M(m_{i+1})$, then $m_{i+1} = m_{ij}$ from Equation (1) for some $j$. Hence by construction the leaf node of $m_{i+1}$ is given a value strictly less than the other $m_{ik}$ for all $k > j$. Since none of the leaf nodes of $m_{ij}$, $p < k$, have corresponding vertices in $S_{i+1}$, the value $f(v) < f(u)$ for all $u \in S_{i+1}$. Thus $D_{M_f}(i + 1) \neq D_{M_f}(i)$. This completes the proof.

**Corollary 23.** For any star graph $S_n$, there are exactly $2^{n-1}$ possible merge trees induced by a critical discrete Morse function on $S_n$.

**Example** We give an example illustrating the construction of Proposition 22. We will find a star graph and discrete Morse function on that graph that induces the merge tree

![Merge Tree Diagram]
We first observe that because there are 6 leaves, we choose \( T := S_5 \). Beginning at the root vertex of \( M \), we arrive at the impasse through the sequence of moves (L)LRRL. At each switch from L to R or R to L, we label the corresponding leaf with the next integer value yielding 1, 2, 3, 4, 5, 6. This corresponds to vertices in \( S_5 \), with the last value given to the center vertex.

Now traverse the path in \( M \) from the impasse to the root, labeling the leaves 4, 5, ... and 6. This corresponds to the same labels on the remaining leaves of \( S_5 \):
Traverse the same path in $M$, labeling the internal nodes 7, 8, 9, 10, 11.

Finally, each edge of $S_5$ is labeled with the same label as the internal nodes of $M$. If edge $e \in S_5$ is incident with non-center vertex labeled $a$, then the leaf in $M$ labeled $a$ is incident with an interior node labeled $b$. Thus define $f(e) = b$ so that the discrete Morse function is

6 \hspace{1cm} Future directions and open questions

In this final section, we share some ideas for future directions that one could pursue.

We were able to compute all merge trees induced by a discrete Morse function on a star graph. What other classes of merge trees can be realized? Can...
one characterize the set of all merge equivalent discrete Morse functions on a caterpillar graph, regular tree, binary tree, or other class of trees? Conversely, given a class of merge trees with some special property, can we find a characterization of the graphs whose set of all merge equivalent discrete Morse functions induces this class of merge trees?

Can any merge tree be realized by a discrete Morse function on some graph? We conjecture that this can be done with the right discrete Morse function on a path. Conversely, we conjecture that if $T$ is a tree containing a vertex with degree greater than 2, then there exists a merge tree that cannot be realized by any discrete Morse function on $T$.

We have seen that there is a one to one correspondence between thin merge trees on $n+1$ leaves and merge equivalent discrete Morse functions on $S_n$. Thin merge trees, however, do not characterize star graphs, as Example 3.2 shows that a non-star graph can induce a thin merge tree. Is there a substantive inverse problem in this setting? That is, given some class of merge trees induced by a discrete Morse function on a tree or a tree up to some notion of equivalence, can we use this class of merge trees to reconstruct the isomorphism type of the tree?

There are variations of equivalence classes of merge trees that can be considered. For example, one could require that there be a total ordering on the vertices of the merge tree, a so-called chiral merge tree [9]. Or one could think of each edge as having a length by putting some weight on the edges. One could also relax the condition that merge trees require a distinction between the left and right children.

References

[1] M. Agiorgousis, B. Green, A. Onderdonk, N. A. Scoville, and K. Rich, Homological sequences in discrete Morse theory, Topology Proc. 54 (2019), 283–294.

[2] R. Ayala, L. M. Fernández, D. Fernández-Ternero, and J. A. Vilches, Discrete Morse theory on graphs, Topology Appl. 156 (2009), no. 18, 3091–3100.

[3] R. Ayala, L. M. Fernández, and J. A. Vilches, Characterizing equivalent discrete Morse functions, Bull. Braz. Math. Soc. (N.S.) 40 (2009), no. 2, 225–235.

[4] R. Ayala, D. Fernández-Ternero, and J. A. Vilches, Counting excellent discrete Morse functions on compact orientable surfaces, Image-A: Applicable Mathematics in Image Engineering 1 (2010), no. 1, 49–56.

[5] R. Ayala, D. Fernández-Ternero, and J. A. Vilches, The number of excellent discrete Morse functions on graphs, Discrete Appl. Math. 159 (2011), no. 16, 1676–1688.
[6] A. J. Blumberg, I. Gal, M. A. Mandell, and M. Pancia, Robust statistics, hypothesis testing, and confidence intervals for persistent homology on metric measure spaces, Found. Comput. Math. 14 (2014), no. 4, 745–789.

[7] G. Carlsson, Topology and data, Bull. Amer. Math. Soc. (N.S.) 46 (2009), no. 2, 255–308.

[8] G. Chartrand, L. Lesniak, and P. Zhang, Graphs & digraphs, sixth ed., Textbooks in Mathematics, CRC Press, Boca Raton, FL, 2016.

[9] J. Curry, The fiber of the persistence map for functions on the interval, J. Appl. Comput. Topol. 2 (2018), no. 3-4, 301–321.

[10] D. L. Ferrario and R. A. Piccinini, Simplicial structures in topology, CMS Books in Mathematics/Ouvrages de Mathématiques de la SMC, Springer, New York, 2011, Translated from the 2009 Italian original by Maria Nair Piccinini.

[11] R. Forman, Discrete Morse theory and the cohomology ring, Trans. Amer. Math. Soc. 354 (2002), no. 12, 5063–5085.

[12] F. Harary, Graph theory, Addison-Wesley Publishing Co., Reading, Mass.-Menlo Park, Calif.-London, 1969.

[13] K. Knudson, Morse theory: Smooth and discrete, World Scientific Publishing Co. Pte. Ltd., Hackensack, NJ, 2015.

[14] D. N. Kozlov, Organized Collapse: An Introduction to Discrete Morse Theory, Graduate studies in mathematics, vol. 207, American Mathematical Society, Providence, RI, 2020.

[15] A. G. Landge, P. Bremer, A. Gyulassy, and Valerio Pascucci, Notes on the distributed computation of merge trees on CW-complexes, Topological methods in data analysis and visualization. IV, Math. Vis., Springer, Cham, 2017, pp. 333–348.

[16] Y. Liu and N. A. Scoville, The realization problem for discrete Morse functions on trees, Algebra Colloquium 27 (2020), no. 3, 455–468.

[17] P. Oesterling, C. Heine, G. H. Weber, D. Morozov, and G. Scheuermann, Computing and visualizing time-varying merge trees for high-dimensional data, Topological methods in data analysis and visualization. IV, Math. Vis., Springer, Cham, 2017, pp. 87–101.

[18] N. A. Scoville, Discrete Morse theory, Student Mathematical Library, vol. 90, American Mathematical Society, Providence, RI, 2019. MR 3970274

[19] N. A. Scoville and K. Yegnesh, A Persistent Homological Analysis of Network Data Flow Malfunctions, Journal of Complex Networks 6 (December 2017), no. 1, 884–892.
[20] G. Spreemann, B. Dunn, M. B. Botnan, and N. A. Baas, *Using persistent homology to reveal hidden covariates in systems governed by the kinetic Ising model*, Phys. Rev. E 97 (2018), no. 3, 032313, 14.