This lecture’s notes defines topological space and looks at the union find algorithm and its usages.

### 3.1 Topological Space in Point Set Topology

Let:

* $\mathbb{X}$: A point set, in its most simple version.
* $\mathcal{U}$: A collection of subsets of $\mathbb{X}$. The elements of $\mathcal{U}$ are called open sets.

**Definition 3.1.** $\mathcal{U}$ is a topology of $\mathbb{X}$ if:

1. $\mathbb{X}$, $\emptyset$ is in $\mathcal{U}$;
2. Any union of sets in $\mathcal{U}$ is in $\mathcal{U}$;
3. Any finite intersection of sets in $\mathcal{U}$ is in $\mathcal{U}$.

**Proof.** Prove that $\mathcal{U}$ is a topology of $\mathbb{X}$, where $\mathbb{X} = \{1, 2, 3\}$ and $\mathcal{U} = \{\emptyset, \{1, 2, 3\}\}$.

Following the definition of a topology:

1. Both $\emptyset$ and $\mathbb{X}$ ($\{1, 2, 3\}$) are in $\mathcal{U}$;
2. $\emptyset \cup \{1, 2, 3\} = \{\emptyset, \{1, 2, 3\}\}$ is in $\mathcal{U}$;
3. $\emptyset \cap \{1, 2, 3\} = \{\emptyset\}$ is in $\mathcal{U}$.

In this case, $\mathcal{U}$ is a trivial topology on $\mathbb{X}$.

**3.1.1 Topological Space**

**Definition 3.2.** The $(\mathbb{X}, \mathcal{U})$ is a topological space.

In some cases, the $\mathcal{U}$ is omitted as it is assumed that the $\mathcal{U}$ is understood. Saying that this space is a topological space attaches the relation from subsets to each set.

**Exercise:** Is $\mathcal{U} = \{\emptyset, \{1\}, \{2\}, \{3\}, \{1, 2\}, \{2, 3\}, \{1, 3\}, \{1, 2, 3\}\}$, also called the power set of $\mathbb{X}$, a topology of $\mathbb{X} = \{1, 2, 3\}$?

Seeing as any union of subsets in $\mathcal{U}$ is also in $\mathcal{U}$, it is a topology of $\mathbb{X}$. Because $\mathcal{U}$ contains all subsets of $\mathbb{X}$, it can be said that $\mathcal{U}$ is a discrete topology of $\mathbb{X}$.

**Exercise:** $\mathbb{R}$ is a real line and $\mathcal{B}$ is a collection of open sets in $\mathbb{R}$. $(\mathbb{R}, \mathcal{B})$ is a topological space and $\mathcal{B}$ is a topology of $\mathbb{R}$.

3-1
3.2 Open Sets in Euclidean Space

**Definition 3.3.** A subset of $\mathbb{R}^n$ is called open if given an point $x \in U$, $\exists$ a real number $\epsilon > 0$, such that for any point $y$ in $\mathbb{R}^n$ whose distance from $x < \epsilon$, $y$ is in $U$.

In this definition, $\epsilon$ represents the neighborhood size. As a point gets closer to the boundary, $\epsilon$ shrinks, but never reaches 0.
Definition 3.4. A closed set is a set whose complement is open.

3.2.1 Continuity

Definition 3.5. A function $f : X \rightarrow Y$, where $X$ and $Y$ are both topological spaces, is continuous if the preimage of every open set is open. For any open set $V \subseteq Y$, $f^{-1}(V) = \{x \in X | f(x) \in V\}$ is an open set of $X$.

Exercise: When we think of a continuous function, we think of a continuous curve. The figure below shows a function that is not continuous at 0.
For any interval \((\epsilon, \epsilon)\), where \(|\epsilon| < 1\), \(f^{-1}(\epsilon, \epsilon)\) is not an open set.

**Proof.** Given the above function, \(f(x) = 0\) for \((-\infty, 0]\) and \(f(x) = 1\) for \((0, \infty)\). The union of these sets is the entire x-axis. For simplicity, assume \(f : \mathbb{X} \rightarrow \mathbb{Y}\), where \(\mathbb{X} = \mathbb{Y} = \mathbb{R}\). To be continuous, the pre-image in \(\mathbb{X}\) of every open set in \(\mathbb{Y}\) must be open.

Consider an open set \((-\epsilon, \epsilon) \subseteq \mathbb{Y}\). Its pre-image is the set of all points \(x \in \mathbb{X}\) such that \(f(x)\) is in the open set \((-\epsilon, \epsilon)\). Look at any such open set in \(\mathbb{Y}\) where \(|\epsilon| > 1\), eg. the interval \((-\epsilon_2, \epsilon_2)\) formed by parentheses at the top and bottom. This set contains both 0 and 1. Since the function \(f\) maps all \(x \in \mathbb{X}\) to either 0 or 1 in \(\mathbb{Y}\), the pre-image of any such set is the entire x-axis (the open interval \((-\infty, \infty) = \mathbb{X} = \mathbb{R}\).

However, when \(|\epsilon| < 1\), as in the case of \((-\epsilon_1, \epsilon_1)\) - the interval formed by middle two parentheses - the set contains 0 but not 1. Since the set of points \(x \in \mathbb{X}\) that map to 0 is \((-\infty, 0]\) which is not an open set in \(\mathbb{X}\), the pre-image of an open set in \(\mathbb{Y}\) is not open and thus the function is not continuous.

If we allow an infinite number of open sets of any size, the intersection of those sets will be a single point. This makes every point an open set and continuous, which is not what we want.

**Definition 3.6.** A path in topological space is a continuous function from \([0, 1] \rightarrow \mathbb{X}\).
Definition 3.7. The topological space is path-connected if every pair of points is connected by a path.

Definition 3.8. Separation is when a path is partitioned into two nonempty, open subsets.

Definition 3.9. If a path has no separation it is connected. This is a weaker relationship than path-connected.

Exercise: The Topologist’s Sine Curve is an example of a function that is connected, but not path connected. The Curve is modeled by, \( f(x) = \begin{cases} 0 & \text{if } x = 0 \\ \sin(1/x) & \text{if } x > 0 \end{cases} \). As you approach 0 from the right, the \( \sin(1/x) \) function oscillates so much, you never actually reach the (0,0) point. Therefore, the function is not path-connected because a path does not exist between (0,0) and the rest of the curve. [EL15]

3.3 Union Find Algorithm or the Disjoint Sets Data Structure

The union find algorithm is an algorithm that decides connectedness.

This algorithm represents each set as a tree element.

Exercise:

![Figure 3.9: A possible tree of the set \{a, b, c, d, e\}.

![Figure 3.10: An array representation of the same tree.

The benefit of this algorithm over depth first search or breadth first search is that it stores the graph as a data structure containing a collection of sets. This works as a reversed tree, since instead of traversing from the root to the children, it traverses from the children to the root.

The union find algorithm has three main operations:

1. \( \text{MakeSet}(x) \): Create a set that contains the single element \( x \). This creates a single node that has a pointer to itself.

2. \( \text{Find}(x) \): Find the root of the tree containing \( x \). For the tree in Figure 3.9, \( \text{Find}(e) = \text{Find}(b) = a \). This works it way from \( x \) to its parent, and that parent’s parent, until the root is reached.
3. *Union(x,y)*: Make root of one tree containing *x* to be root of another tree containing *y*.

![Figure 3.11: The Union(a,p).](image)

**3.3.1 Run Time**

With *Union(x,y)*, the run time is affected based on if larger trees are being attached to smaller tree or vice versa. If singletons are unioned together, creating a tree like the one below, *Find(e)* will operate at the worst run time possible.

![Figure 3.12: When singletons are unioned together, the run time for Find(e) will be O(n).](image)

The run times for the union find operations in *O-*notation, where α is a very slow growing function that functions as a constant and it assumed that the root is already known for *Union(x,y)* [B09]:

<table>
<thead>
<tr>
<th></th>
<th>MakeSet(x)</th>
<th>Find(x)</th>
<th>Union(x,y)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Worst Case</td>
<td><em>O</em>(1)</td>
<td><em>O</em>(1)</td>
<td><em>O</em>(log n)</td>
</tr>
<tr>
<td>Amortized</td>
<td><em>O</em>(1)</td>
<td><em>O</em>(α(n))</td>
<td><em>O</em>(α(n))</td>
</tr>
</tbody>
</table>

**3.3.2 Reducing Run Time**

There are two "hacks" that can be used to reduce run time with union find operations:

1. Union By Rank: Always hang the smaller tree on the larger tree. This requires extra storage for the rank/depth of tree.
   Instead of *Union(a,p)* in Figure 10, the union by rank hack knows that the tree containing *p* is smaller. Therefore, it will instead perform *Union(p,a)*.
2. Path Compression: When using $\text{Find}(x)$, connect all nodes on the path from $x$ to the root directly to the root. This shrinks the height of the tree and increases the efficiency of $\text{Find}(x)$.
References

