

UNIVERSITY OF COPENHAGEN

Union-Find

Author: William Henrich Due

1 Definitions

Definition 1.1 (Union-Find). The union-find data structure U represents a partition of a set S if:

1. $a \neq \emptyset$ for all $a \in \mathcal{C}(U)$
2. $a \cap b = \emptyset$ for all $a, b \in \mathcal{C}(U)$ where $a \neq b$
3. $\bigcup_{a \in \mathcal{C}(U)} a = S$

where $\mathcal{C} : \mathbb{P}(U) \rightarrow \mathbb{P}(\mathbb{P}(S))$ converts the data structure U into a partition of S .

Definition 1.2 (Cycle). A cycle in a directed graph $G = (V, E)$ is a sequence of directed edges $e_1, e_2, \dots, e_m \in E$ where $e_i = (v_{i-1}, v_i)$ for $1 \leq i \leq m$ such that $v_0 = v_m$ and $m > 0$.

Definition 1.3 (Forest). A forest is a directed graph $F = (V, E)$ where V is a set of vertices and $E \subseteq V \times V$ is a set of directed edges such that:

1. There are no cycles.
2. Each node has at most one parent i.e. for all $(u, v_1), (u, v_2) \in E$ it holds that $v_1 = v_2$.

Definition 1.4 (Reachability). A node v is *reachable* from a node u in a directed graph $G = (V, E)$ if there exists a sequence of directed edges $e_1, e_2, \dots, e_m \in E$ (possibly $m = 0$) where $e_i = (v_{i-1}, v_i)$ for $1 \leq i \leq m$, such that $v_0 = u$ and $v_m = v$. We denote this by $u \rightsquigarrow v$.

Definition 1.5 (Union-Find Structure). The union-find data structure U is a forest $U = (V, E)$ where the vertices $V = S$. The edges $E \subseteq V \times V$ represent parent relations between elements in S such that $(u, v) \in E$ means that u has parent v . An element u with no parent is represented by $(u, 0) \in E$ where $0 \notin S$.

Definition 1.6 (Root). A node $v \in V$ in a union-find data structure $U = (V, E)$ is a root if it has no parent i.e. $\mathcal{R}_U(v) : |\{u : u \in V \wedge v \rightsquigarrow u\}| = 1$.

Definition 1.7 (Find Root). The root $r \in V$ of node $v \in V$ in a union-find data structure $U = (V, E)$ is defined as:

1. r is a root node and

-
2. there is path from v to r i.e. $v \rightsquigarrow r$.

Proposition 1.1 (Unique Root). In a union-find data structure $U = (V, E)$ each node $v \in V$ has exactly one root.

Proof. □

Proposition 1.2 (Representation of Union-Find). A cycleless union-find structure $U \in \mathbb{U}_n$ fulfills definition 1.1 for the following conversion function:

$$\mathcal{C}(U) := \{\{f(\pi_1(u)) : u \in U \wedge \mathcal{P}_U(\pi_1(u)) = p\} : p \in \mathbb{N}_{\leq n}\} \setminus \{\emptyset\}$$

where $f : \mathbb{N}_{\leq n} \rightarrow S$ is a bijective function mapping indices to elements in the set S .

Proof. Let $U \in \mathbb{U}_n$ be a union-find data structure fulfilling the three criteria in proposition 1.2. We will show that U fulfills the three criteria in definition 1.1.

1. By definition of \mathcal{C} it is clear that $a \neq \emptyset$ for all $a \in \mathcal{C}(D)$ since \mathcal{C} only includes non-empty sets.
2. Let $a, b \in \mathcal{C}(D)$ where $a \neq b$ then by definition $a = \{f(\pi_1(u)) : u \in U \wedge \mathcal{P}_U(\pi_1(u)) = p_a\}$ and $b = \{f(\pi_1(u)) : u \in U \wedge \mathcal{P}_U(\pi_1(u)) = p_b\}$ for some $p_a, p_b \in \mathbb{N}_{\leq n}$ where $p_a \neq p_b$. Since if $p_a = p_b$ then $a = b$ it follows that $a \cap b = \emptyset$.
3. Let $s \in S$ then since U is a union-find data structure of size n there exists $(i, p) \in U$ for some i and p . Let $r = \mathcal{P}_U(i)$ then by definition of \mathcal{C} it follows that $s \in \{f(\pi_1(u)) : u \in U \wedge \mathcal{P}_U(\pi_1(u)) = r\} \in \mathcal{C}(U)$. Since s was arbitrary it follows that $\bigcup_{a \in \mathcal{C}(U)} a = S$.

□

Definition 1.8 (Substitute Parent). The function for substitution a parent of an element i to p in a cycleless union-find data structure $U \in \mathbb{U}_n$ is denoted $\mathcal{S}_U : \mathbb{N}_{\leq n} \times \mathbb{N}_{\leq n} \rightarrow \mathbb{P}(\mathbb{U}_n)$ and defined as:

$$\mathcal{S}_U(i, p) := (U \setminus \{(i, p')\}) \cup \{(i, p)\} \text{ where } (i, p') \in U$$

Definition 1.9 (Equivalence Set). The set of equivalent indices of an element i in a cycleless union-find data structure $U \in \mathbb{U}_n$ is defined as:

$$\mathcal{E}_U(i) := \{j : j \in \mathbb{N}_{\leq n} \wedge \mathcal{P}_U(j) = \mathcal{P}_U(i)\}$$

Definition 1.10 (Parent). A element $i \in \mathbb{N}_{\leq n}$ has a parent $p \in \mathbb{N}_{\leq n} \cup \{0\}$ if

Definition 1.11 (Equivalent Union-Find Structures). Two cycleless union-find data structures $U, U' \in \mathbb{U}_n$ are equivalent if:

$$\mathcal{E}_{U'}(i) = \mathcal{E}_U(i) \text{ for all } i \in \mathbb{N}_{\leq n}$$

Lemma 1.1 (Cycleless substitution). A parent substitution $\mathcal{S}_U(i, p)$ of a cycleless union-find data structure $U \in \mathbb{U}_n$ will result in a cycleless union-find data structure if:

$$\mathcal{P}_U(i) \neq \mathcal{P}_U(p)$$

Proof. □

Definition 1.12 (Well-formed Union). For a cycleless union-find data structure $U \in \mathbb{U}_n$ the union of two elements $i \sim j$ where $i, j \in \mathbb{N}_{\leq n}$ is well-formed if it results in a cycleless union-find data structure $U' \in \mathbb{U}_n$ such that:

1. $\mathcal{E}_{U'}(i) = \mathcal{E}_U(i) \cup \mathcal{E}_U(j)$ and
2. $\mathcal{E}_{U'}(k) = \mathcal{E}_U(k)$ for all $k \in \mathbb{N}_{\leq n} \setminus (\mathcal{E}_U(i) \cup \mathcal{E}_U(j))$.

Definition 1.13 (Sequential Unions). Given a sequence of union relations $i_1 \sim j_1, i_2 \sim j_2, \dots, i_m \sim j_m$ where $i_k, j_k \in \mathbb{N}_{\leq n}$ for all $1 \leq k \leq m$ and a initial $U \in \mathbb{U}_n$. The final union-find data structure $U' \in \mathbb{U}_n$ is defined as:

$$\begin{aligned} U_0 &:= \{(i, 0) : i \in \mathbb{N}_{\leq n}\} \\ (q_k, p_k) &:= (\mathcal{P}_{U_{k-1}}(i_k), \mathcal{P}_{U_{k-1}}(j_k)) \text{ for } 1 \leq k \leq m \\ U_k &:= \begin{cases} U_{k-1} & \text{if } p_k = q_k \\ \mathcal{S}_{U_{k-1}}(q_k, p_k) & \text{if } p_k \neq q_k \end{cases} \text{ for } 1 \leq k \leq m \\ U' &:= U_m \end{aligned}$$

Proposition 1.3 (Sequential Unions is Well-formed). Given a sequence of union relations $i_1 \sim j_1, i_2 \sim j_2, \dots, i_m \sim j_m$ where $i_k, j_k \in \mathbb{N}_{\leq n}$ for all $1 \leq k \leq m$ and a cycleless union-find data structure $U \in \mathbb{U}_n$. The final union-find data structure $U' \in \mathbb{U}_n$ defined in definition 1.13 is well-formed.

Proof. To prove that U' is well-formed we need to show that each union in the sequence is well-formed. This can be shown by induction on the index k of the union in the sequence.

- For U_0 it is clear that $\mathcal{E}_{U_0}(i) = \{i\}$ for all $i \in \mathbb{N}_{\leq n}$ since all elements are their own root.

-
- Assume that U_{k-1} is well-formed for some $1 \leq k \leq m$. We will show that U_k is well-formed. Let $q_k = \mathcal{P}_{U_{k-1}}(i_k)$ and $p_k = \mathcal{P}_{U_{k-1}}(i_k)$.
 1. If $p_k = q_k$ then by definition $U_k = U_{k-1}$ and thus U_k is well-formed by the inductive hypothesis.
 2. If $p_k \neq q_k$ then since q_k is a root $(q_k, 0) \in U_{k-1}$ and by definition of \mathcal{S} .

$$U_k := (U_{k-1} \setminus \{(q_k, 0)\}) \cup \{(q_k, p_k)\} \text{ where } (q_k, 0) \in U_{k-1}$$

Now $p_k = \mathcal{P}_{U_k}(q_k)$ hence $\mathcal{P}_{U_k}(q) = p_k$ for all $q \in \mathcal{E}_{U_{k-1}}(q_k)$. And by definition of p_k being a root it follows that $\mathcal{P}_{U_k}(p) = p_k$ for all $p \in \mathcal{E}_{U_{k-1}}(p_k)$. Thus $\mathcal{E}_{U_k}(i_k) = \mathcal{E}_{U_{k-1}}(i_k) \cup \mathcal{E}_{U_{k-1}}(j_k)$ and since all other elements are unaffected by the union it follows that $\mathcal{E}_{U_k}(l) = \mathcal{E}_{U_{k-1}}(l)$ for all $l \in \mathbb{N}_{\leq n} \setminus (\mathcal{E}_U(i) \cup \mathcal{E}_U(j))$.

Finally since U_{k-1} is cycleless and q_k is a root it follows that U_k is cycleless and thus U_k is well-formed.

□

Definition 1.14 (Parallel Unions).