

---

**Algorithm 1:** Reference based trimming

---

**Input:** *RefSeq* is a reference genome

*qReads* are reads to check for eccDNA, *q* is a number of reads

**Output:** *T* is an array containing readid, coordinates, and strands of trimmed regions (*r, t1, t2, ts*)

**Initialize:** *A, T*  $\leftarrow$  *Array*(*size* : 0)

```
1 for i  $\leftarrow$  0, ..., q do
2   A  $\leftarrow$  getAlign(RefSeq, qReadsq)
3   for j  $\leftarrow$  0, ..., lengthOf(A) do
4     if lengthOf(A[j]) > 1 then
5       if hasRepeatPattern(A[j]) then
6         | T(r, t1, t2, ts)  $\leftarrow$  A[j]
7       else if A[j](a1, a2) overlap A[j+1](a1, a2) then
8         | if getOverlapLength(A[j](a1, a2), A[j+1](a1, a2))  $\leq$  50 then
9           | | T(r, t1, t2, ts)  $\leftarrow$  A[j]
10          else
11            | if A[j](as) == A[j+1](as) then
12              | | T(r, t1, t2, ts)  $\leftarrow$  merge(A[j], A[j+1])
13            else
14              | if lengthOf(A[j](a1, a2))  $\geq$  lengthOf(A[j+1](a1, a2))
15                | then
16                  | | T(r, t1, t2, ts)  $\leftarrow$  A[j]
17                else
18                  | | T(r, t1, t2, ts)  $\leftarrow$  A[j+1]
19                end if
20              end if
21            end if
22          else
23            | if getGapLength(A[j](a1, a2), A[j+1](a1, a2))  $\leq$  50 then
24              | | T(r, t1, t2, ts)  $\leftarrow$  A[j]
25            end if
26          end if
27        end for
28      end for
```

---

---

**Algorithm 2:** Regions and linkages identification

---

**Input:**  $T$  is a 2D array containing coordinates and strands of trimmed regions grouped by reads  $(t1, t2, ts)$ ,  $r$  is a number of reads,  $t$  is a number of trimmed regions for each read

**Output:**  $M$  is an array of coordinates and strands of merged regions  $(m1, m2, ms)$

**Initialize:**  $M \leftarrow \text{Array}(\text{size} : 0)$   
 $\text{temp} \leftarrow \text{Array}(\text{size} : 3)$

```
1 for  $i \leftarrow 0, \dots, r$  do
2   if  $t > 1$  then
3     for  $j \leftarrow 0, \dots, t$  do
4        $\text{temp}_{(t1, t2, ts)} \leftarrow T_{[ij]}$ 
5       if  $\text{temp}_{(t1, t2, ts)}$  overlap  $M_{(m1, m2, ms)}$  then
6         if  $\text{temp}_{(t1)} < M_{(m1)}$  then
7            $M_{(m1)} \leftarrow \text{temp}_{(t1)}$ 
8         end if
9         if  $\text{temp}_{(t2)} > M_{(m2)}$  then
10           $M_{(m2)} \leftarrow \text{temp}_{(t2)}$ 
11        end if
12      else
13         $M \leftarrow \text{temp}_{(t1, t2, ts)}$ 
14      end if
15    end for
16  end if
17 end for
```

---

---

**Algorithm 3:** Graph construction for eccDNA identification

---

**Input:**  $M$  is an array of coordinates and strands of merged regions  
( $m1, m2, ms$ ),  $m$  is a number of merge regions  
 $T$  is an array containing readid, coordinates, and strands of  
trimmed regions ( $r, t1, t2, ts$ )

**Output:**  $K$  is an array of coordinates, strands, and status of eccDNAs  
( $k1, k2, ks, s$ )

**Initialize:**  $R, tempK \leftarrow Array(size : 0)$   
 $tempM \leftarrow Array(size : 3)$

```
1 for  $i \leftarrow 0, \dots, m$  do
2    $tempM_{(m1, m2, ms)} \leftarrow M_{[i]}$ 
3    $R \leftarrow getOverlapRead(T, tempM_{(m1, m2, ms)})$ 
4   if  $lengthOf(R) > 1$  then
5     for  $j \leftarrow 0, \dots, lengthOf(R)$  do
6        $tempML_{(ml1, ml2, mls)} \leftarrow R_{[j]}$ 
7        $tempMR_{(mr1, mr2, mrs)} \leftarrow R_{[j+1]}$ 
8       if  $getAlignReadSpan(R_{[j]}, tempML_{(ml2)}) \geq 200$  and
           $getAlignReadSpan(R_{[j+1]}, tempMR_{(mr1)}) \geq 200$  then
9         if  $tempML_{(mls)} == tempMR_{(mrs)}$  then
10           $tempK \leftarrow connectGraphMergeRegion(R_{[j]}, R_{[j+1]})$ 
11          end if
12        end if
13      end for
14    end if
15  end for
16  for  $k \leftarrow 0, \dots, lengthOf(tempK)$  do
17     $s \leftarrow checkGraphCircularStatus(tempK_{[k]})$ 
18     $K_{(k1, k2, ks, s)} \leftarrow combineGraphRegion(tempK_{[k]}, s)$ 
19  end for
```

---

---

**Algorithm 4: EccDNA assembly and annotation**

---

**Input:**  $K$  is an array of coordinates, strands, and status of eccDNAs  
( $k1, k2, ks, s$ ),  $k$  is a number of eccDNAs  
 $T$  is an array containing coordinates and strands of trimmed regions ( $t1, t2, ts$ ),  $t$  is a number of trimmed regions

**Output:**  $eRef$  is a sequence file of combined reagrions for each eccDNA  
 $eReads$  is a sequence file of overlapping trimmed regions for each eccDNA  
 $eCon$  is a draft assembled sequence file for each eccDNA  
 $eAnnot$  are tables containing annotations for each eccDNA (Repeats, CpG islands, genes, exons, introns, and variants)

**Initialize:**  $tempK, tempT \leftarrow Array(size : 3)$   
 $eRef, eReads, eCon, eAnnot \leftarrow emptyfile$

```
1 for  $i \leftarrow 0, \dots, k$  do
2    $tempK_{(k1, k2, ks)} \leftarrow K_{[i]}$ 
3    $eAnnot \leftarrow getOverlapRepeat(tempK_{(k1, k2, ks)})$ 
4    $eAnnot \leftarrow getOverlapCpG(tempK_{(k1, k2, ks)})$ 
5    $eAnnot \leftarrow getOverlapGeneExonIntron(tempK_{(k1, k2, ks)})$ 
6   for  $j \leftarrow 0, \dots, t$  do
7      $tempT_{(t1, t2, ts)} \leftarrow T_{[i]}$ 
8     if  $tempT_{(t1, t2, ts)}$  overlap  $tempK_{(k1, k2, ks)}$  then
9        $eReads \leftarrow getSequence(tempT_{(t1, t2, ts)})$ 
10    end if
11  end for
12   $eCon \leftarrow getAssembledSequence(eRef, eReads)$ 
13   $eAnnot \leftarrow getOverlapVariant(eRef, eCon)$ 
14 end for
```

---