Algorithm for Finding Granuloma Clusters

Because the number of granulomas is typically in the dozens, this algorithm as designed doesn’t require the terms to have memory pre-allocation.

First, we make a list of ID’s of all coupled intersections of individual granulomas.

Ex. *IndivIntersectList*=[1,2 ; 1,3 ; 1,4 ; 2,3 ; 2,7 ; 6,10]

means the following:

*IndivGran(1)* intersects with *IndivGran(2)*, *IndivGran(3)*, and *IndivGran(4)*
*IndivGran(2)* intersects with *IndivGran(1)*, *IndivGran(3)*, and *IndivGran(7)*
*IndivGran(3)* intersects with *IndivGran(1)* and *IndivGran(2)*
*IndivGran(4)* intersects with *IndivGran(1)*
*IndivGran(6)* intersects with *IndivGran(10)*
*IndivGran(7)* intersects with *IndivGran(2)*

and the rest of the individual granulomas don’t intersect any others.

Algorithm:

*IndivIntersectList*: list of ID’s of pairs of intersecting individual granulomas

*LenIntersectList*: number of pairs in *IndivIntersectList*

*TotIndivGrans*: total number of individual granulomas

*IndivGranObject*: data structure of all individual granulomas

*ti*: time

*PositionX(ti), PositionY(ti), PositionZ(ti)*: x-, y-, z-coordinates of center of granuloma at time *ti*

*Radius(ti)*: radius of individual granuloma at time *ti*

*IndivIntersectList*=[ ]

*LenIntersectList*=0

if *TotIndivGrans*>1

for ID1=1 to ID1=TotIndivGrans-1

  x_ID1=IndivGranObject(ID1).PositionX(ti)
  y_ID1=IndivGranObject(ID1).PositionY(ti)
  z_ID1=IndivGranObject(ID1).PositionZ(ti)
  Radius_ID1=IndivGranObject(ID1).Radius(ti)

for ID2=ID1+1 to ID2=TotIndivGrans

  x_ID2=IndivGranObject(ID2).PositionX(ti)
  y_ID2=IndivGranObject(ID2).PositionY(ti)
  z_ID2=IndivGranObject(ID2).PositionZ(ti)
  Radius_ID2=IndivGranObject(ID2).Radius(ti)

  TempDist=sqrt((x_ID1-x_ID2)^2+(y_ID1-y_ID2)^2+(z_ID1-z_ID2)^2)
  if TempDist<==(Radius_ID1+Radius_ID2)
```
LenIntersectList = LenIntersectList + 1
IndivIntersectList(LenIntersectList, 1) = ID_1
IndivIntersectList(LenIntersectList, 2) = ID_2
```

end “if” loop
end “for” loop
end “for” loop
end “if” loop

Next, we build a list of individual granuloma ID’s that make up all granuloma clusters.
Ex. with the example above, `IndivIntersectList = [1,2 ; 1,3 ; 1,4 ; 2,3 ; 2,7 ; 6,10]` with
`TotIndivGrans = 10`
we have `GranulomaClusterList = { [1,2,3,4,7] , [6,10] , 5 , 8 , 9 }`
which means that we have a total of 5 granuloma “clusters”: a cluster made up by individual granulomas of ID’s 1,2,3,4, and 7; a cluster made up by individual granulomas of ID’s 6 and 10; a “cluster” made up by individual granulomas of ID 5; a “cluster” made up by individual granulomas of ID 8; and a “cluster” made up by individual granulomas of ID 9.

Algorithm:

GranulomaClusterList: the main output of the algorithm: a list of individual granuloma ID’s that make up each granuloma cluster

LenGranulomaClusterList: the total number of granuloma clusters

IndivIntersectList: list of ID’s of pairs of intersecting individual granulomas (from above)

LenIntersectList: number of pairs in `IndivIntersectList` (from above)

IndivIntersectListTEMP: begins a copy of `IndivIntersectList`, but empties as the algorithm progresses

LenIntersectList: number of pairs in `IndivIntersectListTEMP`

TotIndivGrans: total number of individual granulomas

IndivGranIDList: list of all ID’s for individual granulomas

UnassignedIDList: list of all individual granuloma ID’s that haven’t been assigned to a granuloma cluster

LenUnassignedIDList: number of individual granuloma ID’s that haven’t been assigned to a granuloma cluster—it begins the same as `IndivGranIDList`, but empties as the algorithm progresses

AssignedIDList: list of all individual granuloma ID’s that have been assigned to a granuloma cluster—each individual granuloma ID must belong to either `UnassignedIDList` or `AssignedIDList`, but not both

LenUnassignedIDList: number of individual granuloma ID’s that have been assigned to a granuloma cluster
**TempGranClusterList**: a list of individual granuloma ID’s that each iteration is dynamically constructed, added to *GranulomaClusterList*, then purged before repeating

**LenTempGranClusterList**: number of individual granuloma ID’s in *GranulomaClusterList*

**TempSearchList**: a dynamically-changing list of individual granuloma ID’s that is a part of the current cluster but still pending a move to *TempGranClusterList*—each iteration it begins at 1, then gets added to and one by one moves its elements to *TempGranClusterList* until it is empty

**LenTempSearchList**: number of individual granuloma ID’s that are in *TempSearchList*

**TempSearchID**: a single individual granuloma ID from *TempSearchList* that is being checked against to see whether there are other individual granuloma ID’s paired with *TempSearchID* in *IndivIntersectListTEMP*

**RemoveList**: a list of pairs of items: (1) the index for which pair of ID’s *IndivIntersectListTEMP* that should be removed from *IndivIntersectListTEMP*, and (2) the 2nd ID associated with the index from (1)—each iteration it begins empty then grows dynamically

**LenRemoveList**: the number of pairs of items in *RemoveList*

**RemoveList2**: LenRemoveList2:
**TempValI**: AddToSearchList:
**RemoveFromList**: AddFromList:

**IndivGranIDList=1, 2, …, TotIndivGrans**

If LenIntersectList==0
  GranulomaClusterList=IndivIDGranList
  LenGranulomaClusterList=TotIndivGrans
else
  IndivIntersectListTEMP=IndivIntersectList; LenIntersectListTEMP=LenIntersectList
  UnassignedIDList=IndivIDGranList; LenUnassignedIDList=TotIndivGrans
  AssignedIDList=zeros(size(UnassignedIDList)); LenAssignedIDList=0
  GranulomaClusterList=[]; LenGranulomaClusterList=0

while LenUnassignedIDList>0
  LenTempGranClusterList=0
  TempGranClusterList=[];

  LenTempSearchList=1
  TempSearchList(LenTempSearchList)=UnassignedIDList(1)
  UnassignedIDList(1)=[];
  LenUnassignedIDList=LenUnassignedIDList-1
  AssignedIDList(LenAssignedIDList)=TempSearchList(LenTempSearchList);
  LenAssignedIDList=LenAssignedIDList+1
while LenTempSearchList>0
    TempSearchList=sort(TempSearchList)
    TempSearchID=TempSearchList(1)
    TempSearchList(1)=[]
    LenTempSearchList=LenTempSearchList-1
    LenGranulomaClusterList=LenGranulomaClusterList + 1
    GranulomaClusterList(LenGranulomaClusterList)=TempSearchID

    if LenIntersectListTEMP>0
        RemoveIDList=[]; LenRemoveList=0
        for IntersectListIndex=1 to IntersectListIndex=LenIntersectListTEMP
            if IndivIntersectListTEMP(IntersectListIndex,1)==TempSearchID
                LenTempSearchList=LenTempSearchList+1
                TempSearchList(LenTempSearchList)=IndivIntersectListTEMP(IntersectListIndex,2)
                LenRemoveList=LenRemoveList+1
                RemoveList(LenRemoveList,1)=IntersectListIndex
                RemoveList(LenRemoveList,2)=TempSearchList(LenTempSearchList)
            end "if"
            if IndivIntersectListTEMP(IntersectListIndex,1)>TempSearchID
                break
            end "if"
        end "for"

        if LenRemoveList>0
            for RemoveListIndex=1 to RemoveListIndex=LenRemoveList
                for IntersectListIndex=1 to IntersectListIndex=LenIntersectListTEMP
                    if IndivIntersectListTEMP(IntersectListIndex,2)==RemoveList(LenRemoveList,2)
                        && IndivIntersectListTEMP(IntersectListIndex,1)==TempSearchID
                            AddToSearchList='yes'
                            if LenTempSearchList>0
                                for SearchListIndex=1 to SearchListIndex=LenTempSearchList
                                    if TempSearchList(SearchListIndex)==IndivIntersectListTEMP(IntersectListIndex,1)
                                        AddToSearchList='yes'
                                        break
                                    end "if"
                                end "for"
                            end "if"
                        if AddToSearchList=='yes'
                            LenTempSearchList=LenTempSearchList+1
                        end "if"
                    end "if"
                end "for"
            end "for"
        end "if"
    end "if"
end "if"
LenRemoveList = LenRemoveList + 1
RemoveList(LenRemoveList, 1) = IntersectListIndex
RemoveList(LenRemoveList, 2) = TempSearchList(LenTempSearchList)
end “if”
end “for”
end “for”
end “if”

if LenRemoveList > 0
    IndivIntersectListTEMP(RemoveList(~, 1), ~) = [ ]
    LenIntersectListTEMP = LenIntersectListTEMP - LenRemoveList
    RemoveList2 = zeros(size(RemoveList))
    LenRemoveList2 = 0
    for RemoveListIndex = 1 to RemoveListIndex = LenRemoveList
        TempVal = RemoveList(RemoveListIndex, 2)
        for UnassignedListIndex = 1 to UnassignedListIndex = LenUnassignedList
            if UnassignedList(UnassignedListIndex) == TempVal
                RemoveFromList = ‘yes’
                if RemoveListIndex > 1
                    for RemoveListIndex2 = 1 to RemoveListIndex2 = RemoveListIndex - 1
                        if RemoveList(RemoveListIndex2, 2) == TempVal
                            RemoveFromList = ‘no’
                            break
                        end “if”
                    end “for”
                end “if”
                if RemoveFromList == ‘yes’
                    LenRemoveList2 = LenRemoveList2 + 1
                    RemoveList2(LenRemoveList2, 1) = UnassignedListIndex
                    RemoveList2(LenRemoveList2, 2) = TempVal
                    break
                end “if”
            end “if”
        end “for”
    end “if”
else
    RemoveList2 = RemoveList2(1:LenRemoveList2, ~)
    UnassignedList(RemoveList2(~, 1)) = [ ]
    LenUnassignedList = LenUnassignedList - LenRemoveList2
end “if”

AssignedList(LenAssignedList + 1:LenAssignedList + LenRemoveList2) = Transpose(RemoveList2(~, 2))
LenAssignedList = LenAssignedList + LenRemoveList2
end “if”

end “if”

end “while”

\[ \text{LenGranulomaClusterList} = \text{LenGranulomaClusterList} + 1 \]

\[ \text{GranulomaClusterList}[\text{LenGranulomaClusterList}] = \text{IndivIntersectListTEMP} \]

end “while”

end “if”