```
function BESTNODE(neighborsX, neighborsY)
    indx ← 0, pos ← 0, indn ← 0
    for i=0 to neighborsX length do
        if Nmapping[i] == 0 then
            for j=0 to xlength do
                m ← neighborsX[i]
                n ← neighborsY[i]
                Xc=xclone.indexof(m)
                Yc=yclone.indexof(n)
                if (Xc) == (Yc) then
                    indx ← x.indexof[m]
                    if indx==dest then
                        Dest is in the tr region itself
                    end if
                else
                    XCm=xclone.indexof(m)
                    YCn=yclone.indexof(n)
                    if (XCm) > (YCn) then
                        yclone[YCn] ← 99
                    else
                        xclone[XCm] ← 99
                    end if
                end if
            end for
            Ex=energy[indx]
            Px=pdr[indx]
            if (Ex >= tener) and (Px >= tpdr) then
                time ← distance[i]/pdr[indx]
                if time¡min then
                    flag ← 1, min ← time
                    pos ← indx, indn ← i
                end if
            end if
        end if
    end for
    if flag==1 then
        Nmapping[indn] ← 1
        nextnodex ← x[pos]
        nextnodey ← y[pos]
        src ← pos
        presize ← neighborsXlength
        p ← 1, flag ← 0
        Call NeighborsList()
    else
        No neighbors satisfying the threshold conditions
    end if
end function
```