**Find minimal cost function:**

function [minValues, argmins] = find\_minimum(arr)%define a function to find minimal total cost

der1 = diff(arr);%Difference and approximate derivative. If der1<=0, then the function is decreasing; If der1>=0, then the function is increasing.

idx\_neg = find(der1<=0);% The function is decreasing.

idx\_pos = find(der1>0);% The function is increasing.

idx = ismember(idx\_pos, idx\_neg+1);% If the position of idx\_neg+1 same with idx\_pos, then it is the minimal location.

argmins = idx\_pos(idx);% Return the changed actual start

minValues = arr(argmins);% Return the cost value

% Add special case for the start and the end of array

if isempty(minValues)% If it is empty in minimal total cost, it may have several local minimal total cost

if arr(1) <= arr(2)% Special case: if the first number in array(the value of total cost) smaller than the second

minValues = [arr(1), minValues];% Add the first number into minimal cost set

argmins = [1, argmins];% Add changed actual start=1 into minimal changed actual start set

end

if arr(end) <= arr(end-1)% If the last number of total cost is smaller than the last second

minValues = [minValues, arr(end)];% Add the last number of total cost into minimal cost set

argmins = [argmins,size(arr,2)];% Add changed actual start=the end number into minimal changed actual start set

end

end

end