```
diary on         % begin recording

% FEMA appeal for The Town of Harpswell, Cumberland county, Maine
% TRANSECT ID: CM-158-1
% calculation by SJH, Ransom Consulting, Inc. 26-Feb-2020
% 100-year wave runup using TAW methodology
% including berm and weighted average with foreshore if necessary
%
% chk nld 20200220
%
% This script assumes that the incident wave conditions provided
% as input in the configuration section below are the
% appropriate values located at the end of the foreshore
% or toe of the slope on which the run-up is being calculated
% the script does not attempt to apply a depth limit or any other
% transformation to the incident wave conditions other than
% conversion of the peak wave period to the spectral mean wave
% as recommended in the references below
%
% references:
%
% Van der Meer, J.W., 2002. Technical Report Wave Run-up and
% Wave Overtopping at Dikes. TAW Technical Advisory Committee on
% Flood Defence, The Netherlands.
%
% FEMA. 2007,  Atlantic Ocean and Gulf of Mexico Coastal Guidelines Update
%
%
%

%-------------------------------------------------------------------------
% CONFIG
%-------------------------------------------------------------------------
fname='inpfiles/CM-158-1sta_ele_include.csv';   % file with station, elevation, include
                                        % third colunm is 0 for excluded points
imgname='logfiles/CM-158-1-runup';
SWEL=8.816;  % 100-yr still water level including wave setup.
H0=1.833;    % significant wave height at toe of structure
Tp=2.6668;    % peak period, 1/fma,
T0=Tp/1.1;

gamma_berm=1;   % this may get changed automatically below
gamma_rough=1;
gamma_beta=1;
gamma_perm=1;

setupAtToe=-0.0012402;
maxSetup=0.06101;   % only used in case of berm/shallow foreshore weighted average

plotTitle='Iterative TAW for CM-158-1'

plotTitle =

Iterative TAW for CM-158-1



% END CONFIG
%------------------------------

SWEL=SWEL+setupAtToe

SWEL =

                 8.8147598

SWEL_fore=SWEL+maxSetup

SWEL_fore =

                 8.8757698



% FIND WAVELENGTH USING DEEPWATER DISPERSION RELATION
% using English units
L0=32.15/(2*pi)*T0^2

L0 =

          30.0743764412918


% Find Hb (Munk, 1949)
%Hb=H0/(3.3*(H0/L0)^(1/3))
%Db=-Hb/.78+SWEL;  % depth at breaking

% The toe elevation here is only used to determine the average
% structure slope, it is not used to depth limit the wave height.
% Any depth limiting or other modification of the wave height
```

```
% to make it consitent with TAW guidance should be performed
% prior to the input of the significant wave height given above.
Ztoe=SWEL-1.5*H0

Ztoe =

                 6.0652598


% read the transect
[sta,dep,inc] = textread(fname,'%n%n%n%*[^\n]','delimiter',',','headerlines',0);

% remove unselected points
k=find(inc==0);
sta(k)=[];
dep(k)=[];

sta_org=sta;  % used for plotting purposes
dep_org=dep;


% initial guess at maximum run-up elevation to estimate slope
Z2=SWEL+1.5*H0

Z2 =

               11.5642598


% determine station at the max runup and -1.5*H0 (i.e. the toe)
top_sta=-999;
toe_sta=-999;
for kk=1:length(sta)-1
    if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
       top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
    end
    if  ((Ztoe > dep(kk)) & (Ztoe <= dep(kk+1)))    % here is the intersection of Ztoe with profile
       toe_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Ztoe)
    end
end

toe_sta =

          13.0058230890364


top_sta =

          25.2593016206604

% check to make sure we got them, if not extend the end slopes outward
S=diff(dep)./diff(sta);
if toe_sta==-999
   dy=dep(1)-Ztoe;
   toe_sta=sta(1)-dy/S(1)
end
if top_sta==-999
   dy=Z2-dep(end);
   top_sta=sta(end)+dy/S(end)
end

% just so the reader can tell the values aren't -999 anymore
top_sta

top_sta =

          25.2593016206604

toe_sta

toe_sta =

          13.0058230890364




% check for case where the toe of slope is below SWL-1.5*H0
% in this case interpolate setup from the setupAtToe(really setup as first station), and the max setup
% also un-include points seaward of SWL-1.5*H0
if Ztoe > dep(1)
   dd=SWEL_fore-dep;
   k=find(dd<0,1); % k is index of first land point
   staAtSWL=interp1(dep(k-1:k),sta(k-1:k),SWEL_fore);
   dsta=staAtSWL-sta(1);
   dsetup=maxSetup-setupAtToe;
   dsetdsta=dsetup/dsta;
   setup=setupAtToe+dsetdsta*(toe_sta-sta(1));
   sprintf('-!!- Location of SWEL-1.5*H0 is %4.1f ft landward of toe of slope',dsta)
   sprintf('-!!- Setup is interpolated between setup at toe of slope and max setup')
```

```
   sprintf('-!!-       setup is adjusted to %4.2f feet',setup)
   SWEL=SWEL-setupAtToe+setup;
   sprintf('-!!-       SWEL is adjusted to %4.2f feet',SWEL)
   k=find(dep < SWEL-1.5*H0)
   sta(k)=[];
   dep(k)=[];
else
   sprintf('-!!- The User has selected a starting point that is %4.2f feet above the elevation of SWEL-1.5H0\n',dep(1
   sprintf('-!!- This may be reasonable for some cases.  However the user may want to consider:\n')
   sprintf('-!!-   1) Selecting a starting point that is at or below %4.2f feet elevation, or\n', Ztoe)
   sprintf('-!!-   2) Reducing the incident wave height to a depth limited condition.\n')
end

ans =

-!!- Location of SWEL-1.5*H0 is 27.0 ft landward of toe of slope


ans =

-!!- Setup is interpolated between setup at toe of slope and max setup


ans =

-!!-       setup is adjusted to 0.05 feet


ans =

-!!-       SWEL is adjusted to 8.87 feet


k =

     1
     2
     3
     4
     5
     6
     7
     8
     9
    10
    11
    12
    13
    14
    15
    16
    17
    18
    19
    20
    21
    22
    23
    24



% now iterate converge on a runup elevation
tol=0.01;  % convergence criteria
R2del=999;
R2_new=3*H0; %initial guess
R2=R2_new;
iter=0;
R2_all=[];
topStaAll=[];
Berm_Segs=[];
TAW_ALWAYS_VALID=1;
while(abs(R2del) > tol && iter <= 25)
    iter=iter+1;
    sprintf ('!----------------- STARTING ITERATION %d -----------------!',iter)
    % elevation of toe of slope
    Ztoe
    % station of toe slope (relative to 0-NAVD88 shoreline
    toe_sta
    % station of top of slope/extent of 2% run-up
    top_sta
    % elevation of top of slope/extent of 2% run-up
    Z2
    % incident significant wave height
    H0
    % incident spectral peak wave period
    Tp
    % incident spectral mean wave period
    T0
```

```
R2=R2_new
Z2=R2+SWEL
% determine slope for this iteration
top_sta=-999;
for kk=1:length(sta)-1
   if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
      top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
      break;
   end
end
if top_sta==-999
   dy=Z2-dep(end);
   top_sta=sta(end)+dy/S(end)
end

% get the length of the slope (not accounting for berm)
Lslope=top_sta-toe_sta

% loop over profile segments to determine berm factor
% re-calculate influence of depth of berm based on this run-up elevation
% check for berm, berm width, berm height
berm_width=0;
rdh_sum=0;
Berm_Segs=[];
Berm_Heights=[];
for kk=1:length(sta)-1
   ddep=dep(kk+1)-dep(kk);
   dsta=sta(kk+1)-sta(kk);
   s=ddep/dsta;
   if  (s < 1/15)     % count it as a berm if slope is flatter than 1:15 (see TAW manual)
      sprintf ('Berm Factor Calculation: Iteration %d, Profile Segment: %d',iter,kk)
      berm_width=berm_width+dsta;   % tally the width of all berm segments
      % compute the rdh for this segment and weight it by the segment length
      dh=SWEL-(dep(kk)+dep(kk+1))/2
      if  dh < 0
          chi=R2;
      else
          chi=2* H0;
      end
      if (dh <= R2 & dh >=-2*H0)
         rdh=(0.5-0.5*cos(3.14159*dh/chi)) ;
      else
         rdh=1;
      end
      rdh_sum=rdh_sum + rdh * dsta
      Berm_Segs=[Berm_Segs, kk];
      Berm_Heights=[Berm_Heights, (dep(kk)+dep(kk+1))/2];
   end
   if dep(kk) >= Z2  % jump out of loop if we reached limit of run-up for this iteration
      break
   end
end
sprintf ('!------- End Berm Factor Calculation, Iter: %d ---------!',iter)
berm_width
rB=berm_width/Lslope
if (berm_width > 0)
   rdh_mean=rdh_sum/berm_width
else
   rdh_mean=1
end
gamma_berm=1- rB * (1-rdh_mean)
if gamma_berm > 1
   gamma_berm=1
end
if gamma_berm < 0.6
   gamma_berm =0.6
end
% Iribarren number
slope=(Z2-Ztoe)/(Lslope-berm_width)
Irb=(slope/(sqrt(H0/L0)))
% runup height
gamma_berm
gamma_perm
gamma_beta
gamma_rough
gamma=gamma_berm*gamma_perm*gamma_beta*gamma_rough

% check validity
TAW_VALID=1;
if (Irb*gamma_berm < 0.5 | Irb*gamma_berm > 10 )
   sprintf('!!! - - Iribaren number: %6.2f is outside the valid range (0.5-10), TAW NOT VALID - - !!!\n', Irb*gam
   TAW_VALID=0;
else
   sprintf('!!! - - Iribaren number: %6.2f is in the valid range (0.5-10), TAW RECOMMENDED - - !!!\n', Irb*gamma_
end
islope=1/slope;
if (slope < 1/8 | slope > 1)
   sprintf('!!! - - slope: 1:%3.1f V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!\n', islope)
   TAW_VALID=0;
```

```
    else
       sprintf('!!! - - slope: 1:%3.1f V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!\n', islope)
    end
    if TAW_VALID == 0
       TAW_ALWAYS_VALID=0;
    end

    if (Irb*gamma_berm < 1.8)
       R2_new=gamma*H0*1.77*Irb
    else
       R2_new=gamma*H0*(4.3-(1.6/sqrt(Irb)))
    end


    % check to see if we need to evaluate a shallow foreshore
    if berm_width > 0.25 * L0;
       disp ('!   Berm_width is greater than 1/4 wave length')
       disp ('!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on ber
       % do the foreshore calculation
       fore_H0=0.78*(SWEL_fore-min(Berm_Heights))
       % get upper slope
       fore_toe_sta=-999;
       fore_toe_dep=-999;
       for kk=length(dep)-1:-1:1
          ddep=dep(kk+1)-dep(kk);
          dsta=sta(kk+1)-sta(kk);
          s=ddep/dsta;
          if s < 1/15
             break
          end
          fore_toe_sta=sta(kk);
          fore_toe_dep=dep(kk);
          upper_slope=(Z2-fore_toe_dep)/(top_sta-fore_toe_sta)
       end
       fore_Irb=upper_slope/(sqrt(fore_H0/L0));
       fore_gamma=gamma_perm*gamma_beta*gamma_rough;
       if (fore_Irb < 1.8)
          fore_R2=fore_gamma*fore_H0*1.77*fore_Irb;
       else
          fore_R2=fore_gamma*fore_H0*(4.3-(1.6/sqrt(fore_Irb)));
       end
       if berm_width >= L0
          R2_new=fore_R2
          disp ('berm is wider than one wavelength, use full shallow foreshore solution');
       else
          w2=(berm_width-0.25*L0)/(0.75*L0)
          w1=1-w2
          R2_new=w2*fore_R2 + w1*R2_new
       end
    end % end berm width check
    % convergence criterion
    R2del=abs(R2-R2_new)
    R2_all(iter)=R2_new;
    % get the new top station (for plot purposes)
    Z2=R2_new+SWEL
    top_sta=-999;
    for kk=1:length(sta)-1
       if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
          top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
          break;
       end
    end
    if top_sta==-999
       dy=Z2-dep(end);
       top_sta=sta(end)+dy/S(end);
    end
    topStaAll(iter)=top_sta;
end
ans =
!----------------- STARTING ITERATION 1 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
Z2 =
               11.5642598
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
R2 =
                    5.499
Z2 =
          14.3667865524122
top_sta =
         -32.8054111085127
```

```
Lslope =
         -45.8112341975491
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
          0.431923830491403
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
          0.877877093142679
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 10
dh =
         -2.60941244758778
rdh_sum =
           1.33790298328708
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 11
dh =
         -2.65889844758778
rdh_sum =
           1.81203344910653
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 12
dh =
         -2.70838444758778
rdh_sum =
           2.30028916609791
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 13
dh =
         -2.75787044758778
rdh_sum =
           2.80267952054878
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 14
dh =
         -2.80735644758778
rdh_sum =
           3.31920260203565
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 15
dh =
         -2.85684244758778
rdh_sum =
           3.84984520495081
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 16
dh =
         -2.90632844758778
rdh_sum =
           4.39458283905656
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 17
dh =
         -2.95581444758778
rdh_sum =
           4.95337974905832
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 18
dh =
         -3.00530044758779
rdh_sum =
           5.52618894318109
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 19
dh =
         -3.01099444758778
rdh_sum =
           6.10060690908255
ans =
!------- End Berm Factor Calculation, Iter: 1 ---------!
berm_width =
    12
rB =
        -0.261944481745528
rdh_mean =
          0.508383909090213
gamma_berm =
           1.12877612215113
gamma_berm =
     1
slope =
         -0.143597120311335
Irb =
         -0.581651448469581
```

```
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:  -0.58 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:-7.0 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
         -1.88711577592919
!   Berm_width is greater than 1/4 wave length
!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on berm
fore_H0 =
             -1.951916616
{ Undefined function or variable 'upper_slope'.
}
TAW_iterative
plotTitle =
Iterative TAW for CM-158-1
SWEL =
                8.8147598
SWEL_fore =
                8.8757698
L0 =
          30.0743764412918
Ztoe =
                6.0652598
Z2 =
               11.5642598
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
top_sta =
          25.2593016206604
toe_sta =
          13.0058230890364
ans =
-!!- Location of SWEL-1.5*H0 is 27.0 ft landward of toe of slope
ans =
-!!- Setup is interpolated between setup at toe of slope and max setup
ans =
-!!-       setup is adjusted to 0.05 feet
ans =
-!!-       SWEL is adjusted to 8.87 feet
k =
     1
     2
     3
     4
     5
     6
     7
     8
     9
    10
    11
    12
    13
    14
    15
    16
    17
    18
    19
    20
    21
    22
    23
    24
ans =
!----------------- STARTING ITERATION 1 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
Z2 =
               11.5642598
H0 =
                    1.833
Tp =
                   2.6668
```

```
T0 =
         2.42436363636364
R2 =
                    5.499
Z2 =
         14.3667865524122
top_sta =
        -32.8054111085127
Lslope =
        -45.8112341975491
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 8
dh =
        -2.51044044758778
rdh_sum =
         0.431923830491403
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 9
dh =
        -2.55992644758778
rdh_sum =
         0.877877093142679
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 10
dh =
        -2.60941244758778
rdh_sum =
         1.33790298328708
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 11
dh =
        -2.65889844758778
rdh_sum =
         1.81203344910653
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 12
dh =
        -2.70838444758778
rdh_sum =
         2.30028916609791
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 13
dh =
        -2.75787044758778
rdh_sum =
         2.80267952054878
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 14
dh =
        -2.80735644758778
rdh_sum =
         3.31920260203565
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 15
dh =
        -2.85684244758778
rdh_sum =
         3.84984520495081
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 16
dh =
        -2.90632844758778
rdh_sum =
         4.39458283905656
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 17
dh =
        -2.95581444758778
rdh_sum =
         4.95337974905832
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 18
dh =
        -3.00530044758779
rdh_sum =
         5.52618894318109
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 19
dh =
        -3.01099444758778
rdh_sum =
         6.10060690908255
ans =
!------- End Berm Factor Calculation, Iter: 1 ---------!
berm_width =
    12
rB =
       -0.261944481745528
rdh_mean =
         0.508383909090213
```

```
gamma_berm =
          1.12877612215113
gamma_berm =
     1
slope =
        -0.143597120311335
Irb =
        -0.581651448469581
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:  -0.58 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:-7.0 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
         -1.88711577592919
!   Berm_width is greater than 1/4 wave length
!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on berm
fore_H0 =
              -1.951916616
{ Undefined function or variable 'upper_slope'.
Error in <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative', 'C:\FEMA-TransectAn
nt-weight:bold">TAW_iterative</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpsw
       fore_Irb=upper_slope/(sqrt(fore_H0/L0));
}
opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswell\Additional_Transects\4_taw\TAW_iterative.m',706,
clear all
close all
format long g
diary logfiles/CM-144-1-DIARY.txt  % open a diary file to record calculations
diary on         % begin recording
% FEMA appeal for The Town of Harpswell, Cumberland county, Maine
% TRANSECT ID: CM-158-1
% calculation by SJH, Ransom Consulting, Inc. 26-Feb-2020
% 100-year wave runup using TAW methodology
% including berm and weighted average with foreshore if necessary
%
% chk nld 20200220
%
% This script assumes that the incident wave conditions provided
% as input in the configuration section below are the
% appropriate values located at the end of the foreshore
% or toe of the slope on which the run-up is being calculated
% the script does not attempt to apply a depth limit or any other
% transformation to the incident wave conditions other than
% conversion of the peak wave period to the spectral mean wave
% as recommended in the references below
%
% references:
%
% Van der Meer, J.W., 2002. Technical Report Wave Run-up and
% Wave Overtopping at Dikes. TAW Technical Advisory Committee on
% Flood Defence, The Netherlands.
%
% FEMA. 2007,  Atlantic Ocean and Gulf of Mexico Coastal Guidelines Update
%
%
%
%------------------------------------------------------------------------
% CONFIG
%------------------------------------------------------------------------
fname='inpfiles/CM-158-1sta_ele_include.csv';   % file with station, elevation, include
                                    % third colunm is 0 for excluded points
imgname='logfiles/CM-158-1-runup';
SWEL=8.816;  % 100-yr still water level including wave setup.
H0=1.833;    % significant wave height at toe of structure
Tp=2.6668;    % peak period, 1/fma,
T0=Tp/1.1;
gamma_berm=1;   % this may get changed automatically below
gamma_rough=1;
gamma_beta=1;
gamma_perm=1;
setupAtToe=-0.0012402;
maxSetup=0.06101;   % only used in case of berm/shallow foreshore weighted average
plotTitle='Iterative TAW for CM-158-1'
plotTitle =
Iterative TAW for CM-158-1
% END CONFIG
%------------------------------
SWEL=SWEL+setupAtToe
SWEL =
                8.8147598
```

```
SWEL_fore=SWEL+maxSetup
SWEL_fore =
                8.8757698
% FIND WAVELENGTH USING DEEPWATER DISPERSION RELATION
% using English units
L0=32.15/(2*pi)*T0^2
L0 =
          30.0743764412918
% Find Hb (Munk, 1949)
%Hb=H0/(3.3*(H0/L0)^(1/3))
%Db=-Hb/.78+SWEL;  % depth at breaking
% The toe elevation here is only used to determine the average
% structure slope, it is not used to depth limit the wave height.
% Any depth limiting or other modification of the wave height
% to make it consitent with TAW guidance should be performed
% prior to the input of the significant wave height given above.
Ztoe=SWEL-1.5*H0
Ztoe =
                6.0652598
% read the transect
[sta,dep,inc] = textread(fname,'%n%n%n%*[^\n]','delimiter',',','headerlines',0);
% remove unselected points
k=find(inc==0);
sta(k)=[];
dep(k)=[];
sta_org=sta;  % used for plotting purposes
dep_org=dep;
% initial guess at maximum run-up elevation to estimate slope
Z2=SWEL+1.5*H0
Z2 =
               11.5642598
% determine station at the max runup and -1.5*H0 (i.e. the toe)
top_sta=-999;
toe_sta=-999;
for kk=1:length(sta)-1
    if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
       top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
    end
    if  ((Ztoe > dep(kk)) & (Ztoe <= dep(kk+1)))    % here is the intersection of Ztoe with profile
       toe_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Ztoe)
    end
end
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
% check to make sure we got them, if not extend the end slopes outward
S=diff(dep)./diff(sta);
if toe_sta==-999
   dy=dep(1)-Ztoe;
   toe_sta=sta(1)-dy/S(1)
end
if top_sta==-999
   dy=Z2-dep(end);
   top_sta=sta(end)+dy/S(end)
end
% just so the reader can tell the values aren't -999 anymore
top_sta
top_sta =
          25.2593016206604
toe_sta
toe_sta =
          13.0058230890364
% check for case where the toe of slope is below SWL-1.5*H0
% in this case interpolate setup from the setupAtToe(really setup as first station), and the max setup
% also un-include points seaward of SWL-1.5*H0
if Ztoe > dep(1)
   dd=SWEL_fore-dep;
   k=find(dd<0,1); % k is index of first land point
   staAtSWL=interp1(dep(k-1:k),sta(k-1:k),SWEL_fore);
   dsta=staAtSWL-sta(1);
   dsetup=maxSetup-setupAtToe;
   dsetdsta=dsetup/dsta;
   setup=setupAtToe+dsetdsta*(toe_sta-sta(1));
   sprintf('-!!- Location of SWEL-1.5*H0 is %4.1f ft landward of toe of slope',dsta)
   sprintf('-!!- Setup is interpolated between setup at toe of slope and max setup')
   sprintf('-!!-       setup is adjusted to %4.2f feet',setup)
   SWEL=SWEL-setupAtToe+setup;
   sprintf('-!!-       SWEL is adjusted to %4.2f feet',SWEL)
   k=find(dep < SWEL-1.5*H0)
   sta(k)=[];
   dep(k)=[];
else
   sprintf('-!!- The User has selected a starting point that is %4.2f feet above the elevation of SWEL-1.5H0\n',dep(1
   sprintf('-!!- This may be reasonable for some cases.  However the user may want to consider:\n')
   sprintf('-!!-   1) Selecting a starting point that is at or below %4.2f feet elevation, or\n', Ztoe)
   sprintf('-!!-   2) Reducing the incident wave height to a depth limited condition.\n')
end
ans =
-!!- Location of SWEL-1.5*H0 is 27.0 ft landward of toe of slope
```

```
ans =
-!!- Setup is interpolated between setup at toe of slope and max setup
ans =
-!!-       setup is adjusted to 0.05 feet
ans =
-!!-       SWEL is adjusted to 8.87 feet
k =
     1
     2
     3
     4
     5
     6
     7
     8
     9
    10
    11
    12
    13
    14
    15
    16
    17
    18
    19
    20
    21
    22
    23
    24
% now iterate converge on a runup elevation
tol=0.01;  % convergence criteria
R2del=999;
R2_new=3*H0; %initial guess
R2=R2_new;
iter=0;
R2_all=[];
topStaAll=[];
Berm_Segs=[];
TAW_ALWAYS_VALID=1;
while(abs(R2del) > tol && iter <= 25)
    iter=iter+1;
    sprintf ('!----------------- STARTING ITERATION %d -----------------!',iter)
    % elevation of toe of slope
    Ztoe
    % station of toe slope (relative to 0-NAVD88 shoreline
    toe_sta
    % station of top of slope/extent of 2% run-up
    top_sta
    % elevation of top of slope/extent of 2% run-up
    Z2
    % incident significant wave height
    H0
    % incident spectral peak wave period
    Tp
    % incident spectral mean wave period
    T0

    R2=R2_new
    Z2=R2+SWEL
    % determine slope for this iteration
    top_sta=-999;
    for kk=1:length(sta)-1
       if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
          top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
          break;
       end
    end
    if top_sta==-999
       dy=Z2-dep(end);
       top_sta=sta(end)+dy/S(end)
    end

    % get the length of the slope (not accounting for berm)
    Lslope=top_sta-toe_sta
    % loop over profile segments to determine berm factor
    % re-calculate influence of depth of berm based on this run-up elevation
    % check for berm, berm width, berm height
    berm_width=0;
    rdh_sum=0;
    Berm_Segs=[];
    Berm_Heights=[];
    for kk=1:length(sta)-1
       ddep=dep(kk+1)-dep(kk);
       dsta=sta(kk+1)-sta(kk);
       s=ddep/dsta;
       if  (s < 1/15)     % count it as a berm if slope is flatter than 1:15 (see TAW manual)
          sprintf ('Berm Factor Calculation: Iteration %d, Profile Segment: %d',iter,kk)
          berm_width=berm_width+dsta;   % tally the width of all berm segments
```

```
      % compute the rdh for this segment and weight it by the segment length
      dh=SWEL-(dep(kk)+dep(kk+1))/2
      if  dh < 0
          chi=R2;
      else
          chi=2* H0;
      end
      if (dh <= R2 & dh >=-2*H0)
         rdh=(0.5-0.5*cos(3.14159*dh/chi)) ;
      else
         rdh=1;
      end
      rdh_sum=rdh_sum + rdh * dsta
      Berm_Segs=[Berm_Segs, kk];
      Berm_Heights=[Berm_Heights, (dep(kk)+dep(kk+1))/2];
   end
   if dep(kk) >= Z2  % jump out of loop if we reached limit of run-up for this iteration
      break
   end
end
sprintf ('!------- End Berm Factor Calculation, Iter: %d ---------!',iter)
berm_width
rB=berm_width/Lslope
if (berm_width > 0)
   rdh_mean=rdh_sum/berm_width
else
   rdh_mean=1
end
gamma_berm=1- rB * (1-rdh_mean)
if gamma_berm > 1
   gamma_berm=1
end
if gamma_berm < 0.6
   gamma_berm =0.6
end
% Iribarren number
slope=(Z2-Ztoe)/(Lslope-berm_width)
Irb=(slope/(sqrt(H0/L0)))
% runup height
gamma_berm
gamma_perm
gamma_beta
gamma_rough
gamma=gamma_berm*gamma_perm*gamma_beta*gamma_rough
% check validity
TAW_VALID=1;
if (Irb*gamma_berm < 0.5 | Irb*gamma_berm > 10 )
   sprintf('!!! - - Iribaren number: %6.2f is outside the valid range (0.5-10), TAW NOT VALID - - !!!\n', Irb*gam
   TAW_VALID=0;
else
   sprintf('!!! - - Iribaren number: %6.2f is in the valid range (0.5-10), TAW RECOMMENDED - - !!!\n', Irb*gamma_
end
islope=1/slope;
if (slope < 1/8 | slope > 1)
   sprintf('!!! - - slope: 1:%3.1f V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!\n', islope)
   TAW_VALID=0;
else
   sprintf('!!! - - slope: 1:%3.1f V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!\n', islope)
end
if TAW_VALID == 0
   TAW_ALWAYS_VALID=0;
end

if (Irb*gamma_berm < 1.8)
   R2_new=gamma*H0*1.77*Irb
else
   R2_new=gamma*H0*(4.3-(1.6/sqrt(Irb)))
end
% check to see if we need to evaluate a shallow foreshore
if berm_width > 0.25 * L0;
   disp ('!   Berm_width is greater than 1/4 wave length')
   disp ('!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on ber
   % do the foreshore calculation
   fore_H0=0.78*(SWEL_fore-min(Berm_Heights))
   % get upper slope
   fore_toe_sta=-999;
   fore_toe_dep=-999;
   for kk=length(dep)-1:-1:1
      ddep=dep(kk+1)-dep(kk);
      dsta=sta(kk+1)-sta(kk);
      s=ddep/dsta;
      if s < 1/15
         break
      end
      fore_toe_sta=sta(kk);
      fore_toe_dep=dep(kk);
      upper_slope=(Z2-fore_toe_dep)/(top_sta-fore_toe_sta)
   end
   fore_Irb=upper_slope/(sqrt(fore_H0/L0));
   fore_gamma=gamma_perm*gamma_beta*gamma_rough;
```

```
       if (fore_Irb < 1.8)
          fore_R2=fore_gamma*fore_H0*1.77*fore_Irb;
       else
          fore_R2=fore_gamma*fore_H0*(4.3-(1.6/sqrt(fore_Irb)));
       end
       if berm_width >= L0
          R2_new=fore_R2
          disp ('berm is wider than one wavelength, use full shallow foreshore solution');
       else
          w2=(berm_width-0.25*L0)/(0.75*L0)
          w1=1-w2
          R2_new=w2*fore_R2 + w1*R2_new
       end
    end % end berm width check
    % convergence criterion
    R2del=abs(R2-R2_new)
    R2_all(iter)=R2_new;
    % get the new top station (for plot purposes)
    Z2=R2_new+SWEL
    top_sta=-999;
    for kk=1:length(sta)-1
       if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
          top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
          break;
       end
    end
    if top_sta==-999
       dy=Z2-dep(end);
       top_sta=sta(end)+dy/S(end);
    end
    topStaAll(iter)=top_sta;
end
ans =
!----------------- STARTING ITERATION 1 -----------------!
Ztoe =
                6.0652598
toe_sta =
           13.0058230890364
top_sta =
           25.2593016206604
Z2 =
               11.5642598
H0 =
                    1.833
Tp =
                   2.6668
T0 =
           2.42436363636364
R2 =
                    5.499
Z2 =
           14.3667865524122
top_sta =
           81.8920210243749
Lslope =
           68.8861979353385
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 8
dh =
          -2.51044044758778
rdh_sum =
           0.431923830491403
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 9
dh =
          -2.55992644758778
rdh_sum =
           0.877877093142679
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 10
dh =
          -2.60941244758778
rdh_sum =
           1.33790298328708
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 11
dh =
          -2.65889844758778
rdh_sum =
           1.81203344910653
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 12
dh =
          -2.70838444758778
rdh_sum =
           2.30028916609791
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 13
dh =
          -2.75787044758778
```

```
rdh_sum =
          2.80267952054878
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 14
dh =
         -2.80735644758778
rdh_sum =
          3.31920260203565
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 15
dh =
         -2.85684244758778
rdh_sum =
          3.84984520495081
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 16
dh =
         -2.90632844758778
rdh_sum =
          4.39458283905656
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 17
dh =
         -2.95581444758778
rdh_sum =
          4.95337974905832
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 18
dh =
         -3.00530044758779
rdh_sum =
          5.52618894318109
ans =
!------- End Berm Factor Calculation, Iter: 1 ---------!
berm_width =
    11
rB =
         0.159683656954407
rdh_mean =
         0.502380813016463
gamma_berm =
          0.92053834845179
slope =
         0.143411159283347
Irb =
         0.580898198675621
gamma_berm =
          0.92053834845179
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
          0.92053834845179
ans =
!!! - - Iribaren number:   0.53 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:7.0 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          1.73491278099679
!   Berm_width is greater than 1/4 wave length
!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on berm
fore_H0 =
             -1.951916616
{ Undefined function or variable 'upper_slope'.
}
clear all
close all
format long g
diary logfiles/CM-144-1-DIARY.txt  % open a diary file to record calculations
diary on         % begin recording
% FEMA appeal for The Town of Harpswell, Cumberland county, Maine
% TRANSECT ID: CM-158-1
% calculation by SJH, Ransom Consulting, Inc. 26-Feb-2020
% 100-year wave runup using TAW methodology
% including berm and weighted average with foreshore if necessary
%
% chk nld 20200220
%
% This script assumes that the incident wave conditions provided
% as input in the configuration section below are the
% appropriate values located at the end of the foreshore
% or toe of the slope on which the run-up is being calculated
% the script does not attempt to apply a depth limit or any other
% transformation to the incident wave conditions other than
% conversion of the peak wave period to the spectral mean wave
% as recommended in the references below
%
```

```
% references:
%
% Van der Meer, J.W., 2002. Technical Report Wave Run-up and
% Wave Overtopping at Dikes. TAW Technical Advisory Committee on
% Flood Defence, The Netherlands.
%
% FEMA. 2007,  Atlantic Ocean and Gulf of Mexico Coastal Guidelines Update
%
%
%
%-------------------------------------------------------------------------
% CONFIG
%-------------------------------------------------------------------------
fname='inpfiles/CM-158-1sta_ele_include.csv';   % file with station, elevation, include
                                        % third colunm is 0 for excluded points
imgname='logfiles/CM-158-1-runup';
SWEL=8.816;  % 100-yr still water level including wave setup.
H0=1.833;    % significant wave height at toe of structure
Tp=2.6668;    % peak period, 1/fma,
T0=Tp/1.1;
gamma_berm=1;   % this may get changed automatically below
gamma_rough=1;
gamma_beta=1;
gamma_perm=1;
setupAtToe=-0.0012402;
maxSetup=0.06101;   % only used in case of berm/shallow foreshore weighted average
plotTitle='Iterative TAW for CM-158-1'
plotTitle =
Iterative TAW for CM-158-1
% END CONFIG
%------------------------------
SWEL=SWEL+setupAtToe
SWEL =
                8.8147598
SWEL_fore=SWEL+maxSetup
SWEL_fore =
                8.8757698
% FIND WAVELENGTH USING DEEPWATER DISPERSION RELATION
% using English units
L0=32.15/(2*pi)*T0^2
L0 =
          30.0743764412918
% Find Hb (Munk, 1949)
%Hb=H0/(3.3*(H0/L0)^(1/3))
%Db=-Hb/.78+SWEL;  % depth at breaking
% The toe elevation here is only used to determine the average
% structure slope, it is not used to depth limit the wave height.
% Any depth limiting or other modification of the wave height
% to make it consitent with TAW guidance should be performed
% prior to the input of the significant wave height given above.
Ztoe=SWEL-1.5*H0
Ztoe =
                6.0652598
% read the transect
[sta,dep,inc] = textread(fname,'%n%n%n%*[^\n]','delimiter',',','headerlines',0);
% remove unselected points
k=find(inc==0);
sta(k)=[];
dep(k)=[];
sta_org=sta;  % used for plotting purposes
dep_org=dep;
% initial guess at maximum run-up elevation to estimate slope
Z2=SWEL+1.5*H0
Z2 =
              11.5642598
% determine station at the max runup and -1.5*H0 (i.e. the toe)
top_sta=-999;
toe_sta=-999;
for kk=1:length(sta)-1
    if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
       top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
    end
    if  ((Ztoe > dep(kk)) & (Ztoe <= dep(kk+1)))    % here is the intersection of Ztoe with profile
       toe_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Ztoe)
    end
end
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
% check to make sure we got them, if not extend the end slopes outward
S=diff(dep)./diff(sta);
if toe_sta==-999
   dy=dep(1)-Ztoe;
   toe_sta=sta(1)-dy/S(1)
end
if top_sta==-999
   dy=Z2-dep(end);
   top_sta=sta(end)+dy/S(end)
end
```

```
% just so the reader can tell the values aren't -999 anymore
top_sta
top_sta =
          25.2593016206604
toe_sta
toe_sta =
          13.0058230890364
% check for case where the toe of slope is below SWL-1.5*H0
% in this case interpolate setup from the setupAtToe(really setup as first station), and the max setup
% also un-include points seaward of SWL-1.5*H0
if Ztoe > dep(1)
   dd=SWEL_fore-dep;
   k=find(dd<0,1); % k is index of first land point
   staAtSWL=interp1(dep(k-1:k),sta(k-1:k),SWEL_fore);
   dsta=staAtSWL-sta(1);
   dsetup=maxSetup-setupAtToe;
   dsetdsta=dsetup/dsta;
   setup=setupAtToe+dsetdsta*(toe_sta-sta(1));
   sprintf('-!!- Location of SWEL-1.5*H0 is %4.1f ft landward of toe of slope',dsta)
   sprintf('-!!- Setup is interpolated between setup at toe of slope and max setup')
   sprintf('-!!-       setup is adjusted to %4.2f feet',setup)
   SWEL=SWEL-setupAtToe+setup;
   sprintf('-!!-       SWEL is adjusted to %4.2f feet',SWEL)
   k=find(dep < SWEL-1.5*H0)
   sta(k)=[];
   dep(k)=[];
else
   sprintf('-!!- The User has selected a starting point that is %4.2f feet above the elevation of SWEL-1.5H0\n',dep(1
   sprintf('-!!- This may be reasonable for some cases.  However the user may want to consider:\n')
   sprintf('-!!-   1) Selecting a starting point that is at or below %4.2f feet elevation, or\n', Ztoe)
   sprintf('-!!-   2) Reducing the incident wave height to a depth limited condition.\n')
end
ans =
-!!- Location of SWEL-1.5*H0 is 27.0 ft landward of toe of slope
ans =
-!!- Setup is interpolated between setup at toe of slope and max setup
ans =
-!!-       setup is adjusted to 0.05 feet
ans =
-!!-       SWEL is adjusted to 8.87 feet
k =
     1
     2
     3
     4
     5
     6
     7
     8
     9
    10
    11
    12
    13
    14
    15
    16
    17
    18
    19
    20
    21
    22
    23
    24
% now iterate converge on a runup elevation
tol=0.01;  % convergence criteria
R2del=999;
R2_new=3*H0; %initial guess
R2=R2_new;
iter=0;
R2_all=[];
topStaAll=[];
Berm_Segs=[];
TAW_ALWAYS_VALID=1;
while(abs(R2del) > tol && iter <= 25)
    iter=iter+1;
    sprintf ('!----------------- STARTING ITERATION %d -----------------!',iter)
    % elevation of toe of slope
    Ztoe
    % station of toe slope (relative to 0-NAVD88 shoreline
    toe_sta
    % station of top of slope/extent of 2% run-up
    top_sta
    % elevation of top of slope/extent of 2% run-up
    Z2
    % incident significant wave height
    H0
    % incident spectral peak wave period
    Tp
```

```
% incident spectral mean wave period
T0

R2=R2_new
Z2=R2+SWEL
% determine slope for this iteration
top_sta=-999;
for kk=1:length(sta)-1
   if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
      top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
      break;
   end
end
if top_sta==-999
   dy=Z2-dep(end);
   top_sta=sta(end)+dy/S(end)
end

% get the length of the slope (not accounting for berm)
Lslope=top_sta-toe_sta
% loop over profile segments to determine berm factor
% re-calculate influence of depth of berm based on this run-up elevation
% check for berm, berm width, berm height
berm_width=0;
rdh_sum=0;
Berm_Segs=[];
Berm_Heights=[];
for kk=1:length(sta)-1
   ddep=dep(kk+1)-dep(kk);
   dsta=sta(kk+1)-sta(kk);
   s=ddep/dsta;
   if  (s < 1/15)     % count it as a berm if slope is flatter than 1:15 (see TAW manual)
      sprintf ('Berm Factor Calculation: Iteration %d, Profile Segment: %d',iter,kk)
      berm_width=berm_width+dsta;   % tally the width of all berm segments
      % compute the rdh for this segment and weight it by the segment length
      dh=SWEL-(dep(kk)+dep(kk+1))/2
      if  dh < 0
          chi=R2;
      else
          chi=2* H0;
      end
      if (dh <= R2 & dh >=-2*H0)
         rdh=(0.5-0.5*cos(3.14159*dh/chi)) ;
      else
         rdh=1;
      end
      rdh_sum=rdh_sum + rdh * dsta
      Berm_Segs=[Berm_Segs, kk];
      Berm_Heights=[Berm_Heights, (dep(kk)+dep(kk+1))/2];
   end
   if dep(kk) >= Z2  % jump out of loop if we reached limit of run-up for this iteration
      break
   end
end
sprintf ('!------- End Berm Factor Calculation, Iter: %d ---------!',iter)
berm_width
rB=berm_width/Lslope
if (berm_width > 0)
   rdh_mean=rdh_sum/berm_width
else
   rdh_mean=1
end
gamma_berm=1- rB * (1-rdh_mean)
if gamma_berm > 1
   gamma_berm=1
end
if gamma_berm < 0.6
   gamma_berm =0.6
end
% Iribarren number
slope=(Z2-Ztoe)/(Lslope-berm_width)
Irb=(slope/(sqrt(H0/L0)))
% runup height
gamma_berm
gamma_perm
gamma_beta
gamma_rough
gamma=gamma_berm*gamma_perm*gamma_beta*gamma_rough
% check validity
TAW_VALID=1;
if (Irb*gamma_berm < 0.5 | Irb*gamma_berm > 10 )
   sprintf('!!! - - Iribaren number: %6.2f is outside the valid range (0.5-10), TAW NOT VALID - - !!!\n', Irb*gam
   TAW_VALID=0;
else
   sprintf('!!! - - Iribaren number: %6.2f is in the valid range (0.5-10), TAW RECOMMENDED - - !!!\n', Irb*gamma_
end
islope=1/slope;
if (slope < 1/8 | slope > 1)
   sprintf('!!! - - slope: 1:%3.1f V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!\n', islope)
   TAW_VALID=0;
```

```
    else
       sprintf('!!! - - slope: 1:%3.1f V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!\n', islope)
    end
    if TAW_VALID == 0
       TAW_ALWAYS_VALID=0;
    end

    if (Irb*gamma_berm < 1.8)
       R2_new=gamma*H0*1.77*Irb
    else
       R2_new=gamma*H0*(4.3-(1.6/sqrt(Irb)))
    end
    % check to see if we need to evaluate a shallow foreshore
    if berm_width > 0.25 * L0;
       disp ('!   Berm_width is greater than 1/4 wave length')
       disp ('!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on ber
       % do the foreshore calculation
       fore_H0=0.78*(SWEL_fore-min(Berm_Heights))
       % get upper slope
       fore_toe_sta=-999;
       fore_toe_dep=-999;
       for kk=length(dep)-1:-1:1
          ddep=dep(kk+1)-dep(kk);
          dsta=sta(kk+1)-sta(kk);
          s=ddep/dsta;
          if s < 1/15
             break
          end
          fore_toe_sta=sta(kk);
          fore_toe_dep=dep(kk);
          upper_slope=(Z2-fore_toe_dep)/(top_sta-fore_toe_sta)
       end
       fore_Irb=upper_slope/(sqrt(fore_H0/L0));
       fore_gamma=gamma_perm*gamma_beta*gamma_rough;
       if (fore_Irb < 1.8)
          fore_R2=fore_gamma*fore_H0*1.77*fore_Irb;
       else
          fore_R2=fore_gamma*fore_H0*(4.3-(1.6/sqrt(fore_Irb)));
       end
       if berm_width >= L0
          R2_new=fore_R2
          disp ('berm is wider than one wavelength, use full shallow foreshore solution');
       else
          w2=(berm_width-0.25*L0)/(0.75*L0)
          w1=1-w2
          R2_new=w2*fore_R2 + w1*R2_new
       end
    end % end berm width check
    % convergence criterion
    R2del=abs(R2-R2_new)
    R2_all(iter)=R2_new;
    % get the new top station (for plot purposes)
    Z2=R2_new+SWEL
    top_sta=-999;
    for kk=1:length(sta)-1
       if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
          top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
          break;
       end
    end
    if top_sta==-999
       dy=Z2-dep(end);
       top_sta=sta(end)+dy/S(end);
    end
    topStaAll(iter)=top_sta;
end
ans =
!----------------- STARTING ITERATION 1 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
Z2 =
               11.5642598
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
R2 =
                    5.499
Z2 =
          14.3667865524122
top_sta =
          81.8920210243749
Lslope =
          68.8861979353385
```

```
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
          0.431923830491403
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
          0.877877093142679
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 10
dh =
         -2.60941244758778
rdh_sum =
           1.33790298328708
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 11
dh =
         -2.65889844758778
rdh_sum =
           1.81203344910653
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 12
dh =
         -2.70838444758778
rdh_sum =
           2.30028916609791
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 13
dh =
         -2.75787044758778
rdh_sum =
           2.80267952054878
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 14
dh =
         -2.80735644758778
rdh_sum =
           3.31920260203565
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 15
dh =
         -2.85684244758778
rdh_sum =
           3.84984520495081
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 16
dh =
         -2.90632844758778
rdh_sum =
           4.39458283905656
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 17
dh =
         -2.95581444758778
rdh_sum =
           4.95337974905832
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 18
dh =
         -3.00530044758779
rdh_sum =
           5.52618894318109
ans =
!------- End Berm Factor Calculation, Iter: 1 ---------!
berm_width =
    11
rB =
          0.159683656954407
rdh_mean =
          0.502380813016463
gamma_berm =
           0.92053834845179
slope =
          0.143411159283347
Irb =
          0.580898198675621
gamma_berm =
           0.92053834845179
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
           0.92053834845179
```

```
ans =
!!! - - Iribaren number:   0.53 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:7.0 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          1.73491278099679
!   Berm_width is greater than 1/4 wave length
!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on berm
fore_H0 =
              -1.951916616
{ Undefined function or variable ’upper_slope’.
}
TAW_iterative_writer
tawfilename =
TAW_iterative.m
fid =
     3
ans =
     0
fid =
     3
ans =
     0
fid =
     3
ans =
     0
fid2 =
     4
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Trans
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
```

```
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Trans
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
```

```
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Trans
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
ans =
```

```
     0
clear all
close all
format long g
diary logfiles/CM-144-1-DIARY.txt  % open a diary file to record calculations
diary on         % begin recording
% FEMA appeal for The Town of Harpswell, Cumberland county, Maine
% TRANSECT ID: CM-158-1
% calculation by SJH, Ransom Consulting, Inc. 26-Feb-2020
% 100-year wave runup using TAW methodology
% including berm and weighted average with foreshore if necessary
%
% chk nld 20200220
%
% This script assumes that the incident wave conditions provided
% as input in the configuration section below are the
% appropriate values located at the end of the foreshore
% or toe of the slope on which the run-up is being calculated
% the script does not attempt to apply a depth limit or any other
% transformation to the incident wave conditions other than
% conversion of the peak wave period to the spectral mean wave
% as recommended in the references below
%
% references:
%
% Van der Meer, J.W., 2002. Technical Report Wave Run-up and
% Wave Overtopping at Dikes. TAW Technical Advisory Committee on
% Flood Defence, The Netherlands.
%
% FEMA. 2007,  Atlantic Ocean and Gulf of Mexico Coastal Guidelines Update
%
%
%
%------------------------------------------------------------------------
% CONFIG
%------------------------------------------------------------------------
fname='inpfiles/CM-158-1sta_ele_include.csv';   % file with station, elevation, include
                                        % third colunm is 0 for excluded points
imgname='logfiles/CM-158-1-runup';
SWEL=8.816;  % 100-yr still water level including wave setup.
H0=1.833;    % significant wave height at toe of structure
Tp=2.6668;    % peak period, 1/fma,
T0=Tp/1.1;
gamma_berm=1;   % this may get changed automatically below
gamma_rough=1;
gamma_beta=1;
gamma_perm=1;
setupAtToe=-0.0012402;
maxSetup=0.06101;   % only used in case of berm/shallow foreshore weighted average
plotTitle='Iterative TAW for CM-158-1'
plotTitle =
Iterative TAW for CM-158-1
% END CONFIG
%------------------------------
SWEL=SWEL+setupAtToe
SWEL =
                8.8147598
SWEL_fore=SWEL+maxSetup
SWEL_fore =
                8.8757698
% FIND WAVELENGTH USING DEEPWATER DISPERSION RELATION
% using English units
L0=32.15/(2*pi)*T0^2
L0 =
          30.0743764412918
% Find Hb (Munk, 1949)
%Hb=H0/(3.3*(H0/L0)^(1/3))
%Db=-Hb/.78+SWEL;  % depth at breaking
% The toe elevation here is only used to determine the average
% structure slope, it is not used to depth limit the wave height.
% Any depth limiting or other modification of the wave height
% to make it consitent with TAW guidance should be performed
% prior to the input of the significant wave height given above.
Ztoe=SWEL-1.5*H0
Ztoe =
                6.0652598
% read the transect
[sta,dep,inc] = textread(fname,'%n%n%n%*[^\n]','delimiter',',','headerlines',0);
% remove unselected points
k=find(inc==0);
sta(k)=[];
dep(k)=[];
sta_org=sta;  % used for plotting purposes
dep_org=dep;
% initial guess at maximum run-up elevation to estimate slope
Z2=SWEL+1.5*H0
Z2 =
               11.5642598
% determine station at the max runup and -1.5*H0 (i.e. the toe)
top_sta=-999;
```

```
toe_sta=-999;
for kk=1:length(sta)-1
    if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
       top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
    end
    if  ((Ztoe > dep(kk)) & (Ztoe <= dep(kk+1)))    % here is the intersection of Ztoe with profile
       toe_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Ztoe)
    end
end
toe_sta =
         13.0058230890364
top_sta =
         25.2593016206604
% check to make sure we got them, if not extend the end slopes outward
S=diff(dep)./diff(sta);
if toe_sta==-999
   dy=dep(1)-Ztoe;
   toe_sta=sta(1)-dy/S(1)
end
if top_sta==-999
   dy=Z2-dep(end);
   top_sta=sta(end)+dy/S(end)
end
% just so the reader can tell the values aren't -999 anymore
top_sta
top_sta =
         25.2593016206604
toe_sta
toe_sta =
         13.0058230890364
% check for case where the toe of slope is below SWL-1.5*H0
% in this case interpolate setup from the setupAtToe(really setup as first station), and the max setup
% also un-include points seaward of SWL-1.5*H0
if Ztoe > dep(1)
   dd=SWEL_fore-dep;
   k=find(dd<0,1); % k is index of first land point
   staAtSWL=interp1(dep(k-1:k),sta(k-1:k),SWEL_fore);
   dsta=staAtSWL-sta(1);
   dsetup=maxSetup-setupAtToe;
   dsetdsta=dsetup/dsta;
   setup=setupAtToe+dsetdsta*(toe_sta-sta(1));
   sprintf('-!!- Location of SWEL-1.5*H0 is %4.1f ft landward of toe of slope',dsta)
   sprintf('-!!- Setup is interpolated between setup at toe of slope and max setup')
   sprintf('-!!-       setup is adjusted to %4.2f feet',setup)
   SWEL=SWEL-setupAtToe+setup;
   sprintf('-!!-       SWEL is adjusted to %4.2f feet',SWEL)
   k=find(dep < SWEL-1.5*H0)
   sta(k)=[];
   dep(k)=[];
else
   sprintf('-!!- The User has selected a starting point that is %4.2f feet above the elevation of SWEL-1.5H0\n',dep(1
   sprintf('-!!- This may be reasonable for some cases.  However the user may want to consider:\n')
   sprintf('-!!-   1) Selecting a starting point that is at or below %4.2f feet elevation, or\n', Ztoe)
   sprintf('-!!-   2) Reducing the incident wave height to a depth limited condition.\n')
end
ans =
-!!- Location of SWEL-1.5*H0 is 27.0 ft landward of toe of slope
ans =
-!!- Setup is interpolated between setup at toe of slope and max setup
ans =
-!!-       setup is adjusted to 0.05 feet
ans =
-!!-       SWEL is adjusted to 8.87 feet
k =
     1
     2
     3
     4
     5
     6
     7
     8
     9
    10
    11
    12
    13
    14
    15
    16
    17
    18
    19
    20
    21
    22
    23
    24
% now iterate converge on a runup elevation
tol=0.01;  % convergence criteria
```

```
R2del=999;
R2_new=3*H0; %initial guess
R2=R2_new;
iter=0;
R2_all=[];
topStaAll=[];
Berm_Segs=[];
TAW_ALWAYS_VALID=1;
while(abs(R2del) > tol && iter <= 25)
    iter=iter+1;
    sprintf ('!----------------- STARTING ITERATION %d -----------------!',iter)
    % elevation of toe of slope
    Ztoe
    % station of toe slope (relative to 0-NAVD88 shoreline
    toe_sta
    % station of top of slope/extent of 2% run-up
    top_sta
    % elevation of top of slope/extent of 2% run-up
    Z2
    % incident significant wave height
    H0
    % incident spectral peak wave period
    Tp
    % incident spectral mean wave period
    T0

    R2=R2_new
    Z2=R2+SWEL
    % determine slope for this iteration
    top_sta=-999;
    for kk=1:length(sta)-1
       if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
          top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
          break;
       end
    end
    if top_sta==-999
       dy=Z2-dep(end);
       top_sta=sta(end)+dy/S(end)
    end

    % get the length of the slope (not accounting for berm)
    Lslope=top_sta-toe_sta
    % loop over profile segments to determine berm factor
    % re-calculate influence of depth of berm based on this run-up elevation
    % check for berm, berm width, berm height
    berm_width=0;
    rdh_sum=0;
    Berm_Segs=[];
    Berm_Heights=[];
    for kk=1:length(sta)-1
       ddep=dep(kk+1)-dep(kk);
       dsta=sta(kk+1)-sta(kk);
       s=ddep/dsta;
       if  (s < 1/15)     % count it as a berm if slope is flatter than 1:15 (see TAW manual)
          sprintf ('Berm Factor Calculation: Iteration %d, Profile Segment: %d',iter,kk)
          berm_width=berm_width+dsta;   % tally the width of all berm segments
          % compute the rdh for this segment and weight it by the segment length
          dh=SWEL-(dep(kk)+dep(kk+1))/2
          if  dh < 0
              chi=R2;
          else
              chi=2* H0;
          end
          if (dh <= R2 & dh >=-2*H0)
             rdh=(0.5-0.5*cos(3.14159*dh/chi)) ;
          else
             rdh=1;
          end
          rdh_sum=rdh_sum + rdh * dsta
          Berm_Segs=[Berm_Segs, kk];
          Berm_Heights=[Berm_Heights, (dep(kk)+dep(kk+1))/2];
       end
       if dep(kk) >= Z2  % jump out of loop if we reached limit of run-up for this iteration
          break
       end
    end
    sprintf ('!------- End Berm Factor Calculation, Iter: %d ---------!',iter)
    berm_width
    rB=berm_width/Lslope
    if (berm_width > 0)
       rdh_mean=rdh_sum/berm_width
    else
       rdh_mean=1
    end
    gamma_berm=1- rB * (1-rdh_mean)
    if gamma_berm > 1
       gamma_berm=1
    end
    if gamma_berm < 0.6
```

```
      gamma_berm =0.6
   end
   % Iribarren number
   slope=(Z2-Ztoe)/(Lslope-berm_width)
   Irb=(slope/(sqrt(H0/L0)))
   % runup height
   gamma_berm
   gamma_perm
   gamma_beta
   gamma_rough
   gamma=gamma_berm*gamma_perm*gamma_beta*gamma_rough
   % check validity
   TAW_VALID=1;
   if (Irb*gamma_berm < 0.5 | Irb*gamma_berm > 10 )
      sprintf('!!! - - Iribaren number: %6.2f is outside the valid range (0.5-10), TAW NOT VALID - - !!!\n', Irb*gam
      TAW_VALID=0;
   else
      sprintf('!!! - - Iribaren number: %6.2f is in the valid range (0.5-10), TAW RECOMMENDED - - !!!\n', Irb*gamma_
   end
   islope=1/slope;
   if (slope < 1/8 | slope > 1)
      sprintf('!!! - - slope: 1:%3.1f V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!\n', islope)
      TAW_VALID=0;
   else
      sprintf('!!! - - slope: 1:%3.1f V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!\n', islope)
   end
   if TAW_VALID == 0
      TAW_ALWAYS_VALID=0;
   end

   if (Irb*gamma_berm < 1.8)
      R2_new=gamma*H0*1.77*Irb
   else
      R2_new=gamma*H0*(4.3-(1.6/sqrt(Irb)))
   end
   % check to see if we need to evaluate a shallow foreshore
   if berm_width > 0.25 * L0;
      disp ('!   Berm_width is greater than 1/4 wave length')
      disp ('!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on ber
      % do the foreshore calculation
      fore_H0=0.78*(SWEL_fore-min(Berm_Heights))
      % get upper slope
      fore_toe_sta=-999;
      fore_toe_dep=-999;
      for kk=length(dep)-1:-1:1
         ddep=dep(kk+1)-dep(kk);
         dsta=sta(kk+1)-sta(kk);
         s=ddep/dsta;
         if s < 1/15
            break
         end
         fore_toe_sta=sta(kk);
         fore_toe_dep=dep(kk);
         upper_slope=(Z2-fore_toe_dep)/(top_sta-fore_toe_sta)
      end
      fore_Irb=upper_slope/(sqrt(fore_H0/L0));
      fore_gamma=gamma_perm*gamma_beta*gamma_rough;
      if (fore_Irb < 1.8)
         fore_R2=fore_gamma*fore_H0*1.77*fore_Irb;
      else
         fore_R2=fore_gamma*fore_H0*(4.3-(1.6/sqrt(fore_Irb)));
      end
      if berm_width >= L0
         R2_new=fore_R2
         disp ('berm is wider than one wavelength, use full shallow foreshore solution');
      else
         w2=(berm_width-0.25*L0)/(0.75*L0)
         w1=1-w2
         R2_new=w2*fore_R2 + w1*R2_new
      end
   end % end berm width check
   % convergence criterion
   R2del=abs(R2-R2_new)
   R2_all(iter)=R2_new;
   % get the new top station (for plot purposes)
   Z2=R2_new+SWEL
   top_sta=-999;
   for kk=1:length(sta)-1
      if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
         top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
         break;
      end
   end
   if top_sta==-999
      dy=Z2-dep(end);
      top_sta=sta(end)+dy/S(end);
   end
   topStaAll(iter)=top_sta;
end
ans =
```

```
!----------------- STARTING ITERATION 1 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
Z2 =
               11.5642598
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
R2 =
                    5.499
Z2 =
          14.3667865524122
top_sta =
          81.8920210243749
Lslope =
          68.8861979353385
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
          0.431923830491403
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
          0.877877093142679
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 10
dh =
         -2.60941244758778
rdh_sum =
           1.33790298328708
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 11
dh =
         -2.65889844758778
rdh_sum =
           1.81203344910653
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 12
dh =
         -2.70838444758778
rdh_sum =
           2.30028916609791
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 13
dh =
         -2.75787044758778
rdh_sum =
           2.80267952054878
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 14
dh =
         -2.80735644758778
rdh_sum =
           3.31920260203565
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 15
dh =
         -2.85684244758778
rdh_sum =
           3.84984520495081
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 16
dh =
         -2.90632844758778
rdh_sum =
           4.39458283905656
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 17
dh =
         -2.95581444758778
rdh_sum =
           4.95337974905832
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 18
dh =
         -3.00530044758779
rdh_sum =
           5.52618894318109
ans =
```

```
!------- End Berm Factor Calculation, Iter: 1 ---------!
berm_width =
    11
rB =
         0.159683656954407
rdh_mean =
         0.502380813016463
gamma_berm =
          0.92053834845179
slope =
         0.143411159283347
Irb =
         0.580898198675621
gamma_berm =
          0.92053834845179
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
          0.92053834845179
ans =
!!! - - Iribaren number:   0.53 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:7.0 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          1.73491278099679
!   Berm_width is greater than 1/4 wave length
!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on berm
fore_H0 =
              -1.951916616
{ Undefined function or variable 'upper_slope'.
}
TAW_iterative_writer
tawfilename =
TAW_iterative.m
fid =
     3
ans =
     0
fid =
     3
ans =
     0
fid =
     3
ans =
     0
fid2 =
     4
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Trans
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
```

```
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Trans
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
```

```
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Trans
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
```

```
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
ans =
     0
clear all
close all
format long g
diary logfiles/CM-144-1-DIARY.txt  % open a diary file to record calculations
diary on         % begin recording
% FEMA appeal for The Town of Harpswell, Cumberland county, Maine
% TRANSECT ID: CM-158-1
% calculation by SJH, Ransom Consulting, Inc. 26-Feb-2020
% 100-year wave runup using TAW methodology
% including berm and weighted average with foreshore if necessary
%
% chk nld 20200220
%
% This script assumes that the incident wave conditions provided
% as input in the configuration section below are the
% appropriate values located at the end of the foreshore
% or toe of the slope on which the run-up is being calculated
% the script does not attempt to apply a depth limit or any other
% transformation to the incident wave conditions other than
% conversion of the peak wave period to the spectral mean wave
% as recommended in the references below
%
% references:
%
% Van der Meer, J.W., 2002. Technical Report Wave Run-up and
% Wave Overtopping at Dikes. TAW Technical Advisory Committee on
% Flood Defence, The Netherlands.
%
% FEMA. 2007,  Atlantic Ocean and Gulf of Mexico Coastal Guidelines Update
%
%
%
%--------------------------------------------------------------------------
% CONFIG
%--------------------------------------------------------------------------
fname=’inpfiles/CM-158-1sta_ele_include.csv’;   % file with station, elevation, include
                                      % third colunm is 0 for excluded points
imgname=’logfiles/CM-158-1-runup’;
SWEL=8.816;  % 100-yr still water level including wave setup.
H0=1.833;    % significant wave height at toe of structure
Tp=2.6668;    % peak period, 1/fma,
T0=Tp/1.1;
gamma_berm=1;   % this may get changed automatically below
gamma_rough=1;
gamma_beta=1;
gamma_perm=1;
setupAtToe=-0.0012402;
maxSetup=0.06101;   % only used in case of berm/shallow foreshore weighted average
plotTitle=’Iterative TAW for CM-158-1’
plotTitle =
Iterative TAW for CM-158-1
% END CONFIG
%--------------------------------
SWEL=SWEL+setupAtToe
SWEL =
                 8.8147598
SWEL_fore=SWEL+maxSetup
SWEL_fore =
                 8.8757698
% FIND WAVELENGTH USING DEEPWATER DISPERSION RELATION
% using English units
L0=32.15/(2*pi)*T0^2
L0 =
          30.0743764412918
% Find Hb (Munk, 1949)
%Hb=H0/(3.3*(H0/L0)^(1/3))
```

```
%Db=-Hb/.78+SWEL;  % depth at breaking
% The toe elevation here is only used to determine the average
% structure slope, it is not used to depth limit the wave height.
% Any depth limiting or other modification of the wave height
% to make it consitent with TAW guidance should be performed
% prior to the input of the significant wave height given above.
Ztoe=SWEL-1.5*H0
Ztoe =
                6.0652598
% read the transect
[sta,dep,inc] = textread(fname,'%n%n%n%*[^\n]','delimiter',',','headerlines',0);
% remove unselected points
k=find(inc==0);
sta(k)=[];
dep(k)=[];
sta_org=sta;  % used for plotting purposes
dep_org=dep;
% initial guess at maximum run-up elevation to estimate slope
Z2=SWEL+1.5*H0
Z2 =
               11.5642598
% determine station at the max runup and -1.5*H0 (i.e. the toe)
top_sta=-999;
toe_sta=-999;
for kk=1:length(sta)-1
    if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
       top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
    end
    if  ((Ztoe > dep(kk)) & (Ztoe <= dep(kk+1)))    % here is the intersection of Ztoe with profile
       toe_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Ztoe)
    end
end
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
% check to make sure we got them, if not extend the end slopes outward
S=diff(dep)./diff(sta);
if toe_sta==-999
   dy=dep(1)-Ztoe;
   toe_sta=sta(1)-dy/S(1)
end
if top_sta==-999
   dy=Z2-dep(end);
   top_sta=sta(end)+dy/S(end)
end
% just so the reader can tell the values aren't -999 anymore
top_sta
top_sta =
          25.2593016206604
toe_sta
toe_sta =
          13.0058230890364
% check for case where the toe of slope is below SWL-1.5*H0
% in this case interpolate setup from the setupAtToe(really setup as first station), and the max setup
% also un-include points seaward of SWL-1.5*H0
if Ztoe > dep(1)
   dd=SWEL_fore-dep;
   k=find(dd<0,1); % k is index of first land point
   staAtSWL=interp1(dep(k-1:k),sta(k-1:k),SWEL_fore);
   dsta=staAtSWL-sta(1);
   dsetup=maxSetup-setupAtToe;
   dsetdsta=dsetup/dsta;
   setup=setupAtToe+dsetdsta*(toe_sta-sta(1));
   sprintf('-!!- Location of SWEL-1.5*H0 is %4.1f ft landward of toe of slope',dsta)
   sprintf('-!!- Setup is interpolated between setup at toe of slope and max setup')
   sprintf('-!!-       setup is adjusted to %4.2f feet',setup)
   SWEL=SWEL-setupAtToe+setup;
   sprintf('-!!-       SWEL is adjusted to %4.2f feet',SWEL)
   k=find(dep < SWEL-1.5*H0)
   sta(k)=[];
   dep(k)=[];
else
   sprintf('-!!- The User has selected a starting point that is %4.2f feet above the elevation of SWEL-1.5H0\n',dep(1
   sprintf('-!!- This may be reasonable for some cases.  However the user may want to consider:\n')
   sprintf('-!!-   1) Selecting a starting point that is at or below %4.2f feet elevation, or\n', Ztoe)
   sprintf('-!!-   2) Reducing the incident wave height to a depth limited condition.\n')
end
ans =
-!!- Location of SWEL-1.5*H0 is 27.0 ft landward of toe of slope
ans =
-!!- Setup is interpolated between setup at toe of slope and max setup
ans =
-!!-       setup is adjusted to 0.05 feet
ans =
-!!-       SWEL is adjusted to 8.87 feet
k =
     1
     2
     3
```

```
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
% now iterate converge on a runup elevation
tol=0.01;  % convergence criteria
R2del=999;
R2_new=3*H0; %initial guess
R2=R2_new;
iter=0;
R2_all=[];
topStaAll=[];
Berm_Segs=[];
TAW_ALWAYS_VALID=1;
while(abs(R2del) > tol && iter <= 25)
    iter=iter+1;
    sprintf ('!----------------- STARTING ITERATION %d -----------------!',iter)
    % elevation of toe of slope
    Ztoe
    % station of toe slope (relative to 0-NAVD88 shoreline
    toe_sta
    % station of top of slope/extent of 2% run-up
    top_sta
    % elevation of top of slope/extent of 2% run-up
    Z2
    % incident significant wave height
    H0
    % incident spectral peak wave period
    Tp
    % incident spectral mean wave period
    T0

    R2=R2_new
    Z2=R2+SWEL
    % determine slope for this iteration
    top_sta=-999;
    for kk=1:length(sta)-1
       if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
          top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
          break;
       end
    end
    if top_sta==-999
       dy=Z2-dep(end);
       top_sta=sta(end)+dy/S(end)
    end

    % get the length of the slope (not accounting for berm)
    Lslope=top_sta-toe_sta
    % loop over profile segments to determine berm factor
    % re-calculate influence of depth of berm based on this run-up elevation
    % check for berm, berm width, berm height
    berm_width=0;
    rdh_sum=0;
    Berm_Segs=[];
    Berm_Heights=[];
    for kk=1:length(sta)-1
       ddep=dep(kk+1)-dep(kk);
       dsta=sta(kk+1)-sta(kk);
       s=ddep/dsta;
       if  (s < 1/15)     % count it as a berm if slope is flatter than 1:15 (see TAW manual)
          sprintf ('Berm Factor Calculation: Iteration %d, Profile Segment: %d',iter,kk)
          berm_width=berm_width+dsta;   % tally the width of all berm segments
          % compute the rdh for this segment and weight it by the segment length
          dh=SWEL-(dep(kk)+dep(kk+1))/2
          if  dh < 0
              chi=R2;
          else
              chi=2* H0;
          end
          if (dh <= R2 & dh >=-2*H0)
             rdh=(0.5-0.5*cos(3.14159*dh/chi)) ;
          else
```

```
         rdh=1;
      end
      rdh_sum=rdh_sum + rdh * dsta
      Berm_Segs=[Berm_Segs, kk];
      Berm_Heights=[Berm_Heights, (dep(kk)+dep(kk+1))/2];
   end
   if dep(kk) >= Z2  % jump out of loop if we reached limit of run-up for this iteration
      break
   end
end
sprintf ('!------- End Berm Factor Calculation, Iter: %d ---------!',iter)
berm_width
rB=berm_width/Lslope
if (berm_width > 0)
   rdh_mean=rdh_sum/berm_width
else
   rdh_mean=1
end
gamma_berm=1- rB * (1-rdh_mean)
if gamma_berm > 1
   gamma_berm=1
end
if gamma_berm < 0.6
   gamma_berm =0.6
end
% Iribarren number
slope=(Z2-Ztoe)/(Lslope-berm_width)
Irb=(slope/(sqrt(H0/L0)))
% runup height
gamma_berm
gamma_perm
gamma_beta
gamma_rough
gamma=gamma_berm*gamma_perm*gamma_beta*gamma_rough
% check validity
TAW_VALID=1;
if (Irb*gamma_berm < 0.5 | Irb*gamma_berm > 10 )
   sprintf('!!! - - Iribaren number: %6.2f is outside the valid range (0.5-10), TAW NOT VALID - - !!!\n', Irb*gam
   TAW_VALID=0;
else
   sprintf('!!! - - Iribaren number: %6.2f is in the valid range (0.5-10), TAW RECOMMENDED - - !!!\n', Irb*gamma_
end
islope=1/slope;
if (slope < 1/8 | slope > 1)
   sprintf('!!! - - slope: 1:%3.1f V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!\n', islope)
   TAW_VALID=0;
else
   sprintf('!!! - - slope: 1:%3.1f V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!\n', islope)
end
if TAW_VALID == 0
   TAW_ALWAYS_VALID=0;
end

if (Irb*gamma_berm < 1.8)
   R2_new=gamma*H0*1.77*Irb
else
   R2_new=gamma*H0*(4.3-(1.6/sqrt(Irb)))
end
% check to see if we need to evaluate a shallow foreshore
if berm_width > 0.25 * L0;
   disp ('!   Berm_width is greater than 1/4 wave length')
   disp ('!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on ber
   % do the foreshore calculation
   fore_H0=0.78*(SWEL_fore-min(Berm_Heights))
   % get upper slope
   fore_toe_sta=-999;
   fore_toe_dep=-999;
   for kk=length(dep)-1:-1:1
      ddep=dep(kk+1)-dep(kk);
      dsta=sta(kk+1)-sta(kk);
      s=ddep/dsta;
      if s < 1/15
         break
      end
      fore_toe_sta=sta(kk);
      fore_toe_dep=dep(kk);
      upper_slope=(Z2-fore_toe_dep)/(top_sta-fore_toe_sta)
   end
   fore_Irb=upper_slope/(sqrt(fore_H0/L0));
   fore_gamma=gamma_perm*gamma_beta*gamma_rough;
   if (fore_Irb < 1.8)
      fore_R2=fore_gamma*fore_H0*1.77*fore_Irb;
   else
      fore_R2=fore_gamma*fore_H0*(4.3-(1.6/sqrt(fore_Irb)));
   end
   if berm_width >= L0
      R2_new=fore_R2
      disp ('berm is wider than one wavelength, use full shallow foreshore solution');
   else
      w2=(berm_width-0.25*L0)/(0.75*L0)
```

```
          w1=1-w2
          R2_new=w2*fore_R2 + w1*R2_new
       end
    end % end berm width check
    % convergence criterion
    R2del=abs(R2-R2_new)
    R2_all(iter)=R2_new;
    % get the new top station (for plot purposes)
    Z2=R2_new+SWEL
    top_sta=-999;
    for kk=1:length(sta)-1
       if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
          top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
          break;
       end
    end
    if top_sta==-999
       dy=Z2-dep(end);
       top_sta=sta(end)+dy/S(end);
    end
    topStaAll(iter)=top_sta;
end
ans =
!----------------- STARTING ITERATION 1 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
Z2 =
               11.5642598
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
R2 =
                    5.499
Z2 =
          14.3667865524122
top_sta =
         -32.8054111085127
Lslope =
         -45.8112341975491
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
          0.431923830491403
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
          0.877877093142679
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 10
dh =
         -2.60941244758778
rdh_sum =
          1.33790298328708
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 11
dh =
         -2.65889844758778
rdh_sum =
          1.81203344910653
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 12
dh =
         -2.70838444758778
rdh_sum =
          2.30028916609791
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 13
dh =
         -2.75787044758778
rdh_sum =
          2.80267952054878
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 14
dh =
         -2.80735644758778
rdh_sum =
          3.31920260203565
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 15
```

```
dh =
         -2.85684244758778
rdh_sum =
          3.84984520495081
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 16
dh =
         -2.90632844758778
rdh_sum =
          4.39458283905656
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 17
dh =
         -2.95581444758778
rdh_sum =
          4.95337974905832
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 18
dh =
         -3.00530044758779
rdh_sum =
          5.52618894318109
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 19
dh =
         -3.01099444758778
rdh_sum =
          6.10060690908255
ans =
!------- End Berm Factor Calculation, Iter: 1 ---------!
berm_width =
    12
rB =
        -0.261944481745528
rdh_mean =
         0.508383909090213
gamma_berm =
          1.12877612215113
gamma_berm =
     1
slope =
        -0.143597120311335
Irb =
        -0.581651448469581
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:  -0.58 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:-7.0 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
         -1.88711577592919
!   Berm_width is greater than 1/4 wave length
!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on berm
fore_H0 =
              -1.951916616
{ Undefined function or variable 'upper_slope'.
}
TAW_iterative
plotTitle =
Iterative TAW for CM-158-1
SWEL =
                 8.8147598
SWEL_fore =
                 8.8757698
L0 =
          30.0743764412918
Ztoe =
                 6.0652598
Z2 =
                11.5642598
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
top_sta =
          25.2593016206604
toe_sta =
          13.0058230890364
ans =
-!!- Location of SWEL-1.5*H0 is 27.0 ft landward of toe of slope
ans =
```

```
-!!- Setup is interpolated between setup at toe of slope and max setup
ans =
-!!-       setup is adjusted to 0.05 feet
ans =
-!!-       SWEL is adjusted to 8.87 feet
k =
     1
     2
     3
     4
     5
     6
     7
     8
     9
    10
    11
    12
    13
    14
    15
    16
    17
    18
    19
    20
    21
    22
    23
    24
ans =
!----------------- STARTING ITERATION 1 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
Z2 =
               11.5642598
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
R2 =
                    5.499
Z2 =
          14.3667865524122
top_sta =
         -32.8054111085127
Lslope =
         -45.8112341975491
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
          0.431923830491403
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
          0.877877093142679
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 10
dh =
         -2.60941244758778
rdh_sum =
           1.33790298328708
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 11
dh =
         -2.65889844758778
rdh_sum =
           1.81203344910653
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 12
dh =
         -2.70838444758778
rdh_sum =
           2.30028916609791
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 13
dh =
         -2.75787044758778
rdh_sum =
           2.80267952054878
```

```
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 14
dh =
         -2.80735644758778
rdh_sum =
          3.31920260203565
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 15
dh =
         -2.85684244758778
rdh_sum =
          3.84984520495081
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 16
dh =
         -2.90632844758778
rdh_sum =
          4.39458283905656
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 17
dh =
         -2.95581444758778
rdh_sum =
          4.95337974905832
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 18
dh =
         -3.00530044758779
rdh_sum =
          5.52618894318109
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 19
dh =
         -3.01099444758778
rdh_sum =
          6.10060690908255
ans =
!------- End Berm Factor Calculation, Iter: 1 ---------!
berm_width =
    12
rB =
        -0.261944481745528
rdh_mean =
         0.508383909090213
gamma_berm =
          1.12877612215113
gamma_berm =
     1
slope =
        -0.143597120311335
Irb =
        -0.581651448469581
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:  -0.58 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:-7.0 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
         -1.88711577592919
!   Berm_width is greater than 1/4 wave length
!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on berm
fore_H0 =
             -1.951916616
{ Undefined function or variable 'upper_slope'.
Error in <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative', 'C:\FEMA-TransectAn
nt-weight:bold">TAW_iterative</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpsw
       fore_Irb=upper_slope/(sqrt(fore_H0/L0));
}
TAW_iterative_writer
tawfilename =
TAW_iterative.m
fid =
     3
ans =
     0
fid =
     3
ans =
     0
fid =
     3
```

```
ans =
     0
fid2 =
     4
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Trans
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Trans
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
```

```
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Trans
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
```

```
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
ans =
     0
TAW_iterative
plotTitle =
Iterative TAW for CM-158-1
SWEL =
                 8.8147598
SWEL_fore =
                 8.8757698
L0 =
          30.0743764412918
Ztoe =
                 6.0652598
Z2 =
                11.5642598
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
top_sta =
          25.2593016206604
toe_sta =
          13.0058230890364
ans =
-!!- Location of SWEL-1.5*H0 is 27.0 ft landward of toe of slope
ans =
```

```
-!!- Setup is interpolated between setup at toe of slope and max setup
ans =
-!!-       setup is adjusted to 0.05 feet
ans =
-!!-       SWEL is adjusted to 8.87 feet
k =
     1
     2
     3
     4
     5
     6
     7
     8
     9
    10
    11
    12
    13
    14
    15
    16
    17
    18
    19
    20
    21
    22
    23
    24
ans =
!----------------- STARTING ITERATION 1 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
Z2 =
               11.5642598
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
R2 =
                    5.499
Z2 =
          14.3667865524122
top_sta =
          81.8920210243731
Lslope =
          68.8861979353367
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
         0.431923830491403
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
         0.877877093142679
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 10
dh =
         -2.60941244758778
rdh_sum =
          1.33790298328708
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 11
dh =
         -2.65889844758778
rdh_sum =
          1.81203344910653
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 12
dh =
         -2.70838444758778
rdh_sum =
          2.30028916609791
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 13
dh =
         -2.75787044758778
rdh_sum =
          2.80267952054878
```

```
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 14
dh =
         -2.80735644758778
rdh_sum =
          3.31920260203565
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 15
dh =
         -2.85684244758778
rdh_sum =
          3.84984520495081
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 16
dh =
         -2.90632844758778
rdh_sum =
          4.39458283905656
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 17
dh =
         -2.95581444758778
rdh_sum =
          4.95337974905832
ans =
!------- End Berm Factor Calculation, Iter: 1 ---------!
berm_width =
    10
rB =
         0.145166960867647
rdh_mean =
         0.495337974905832
gamma_berm =
         0.926739747551768
slope =
         0.140975764160019
Irb =
         0.571033438867008
gamma_berm =
         0.926739747551768
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
         0.926739747551768
ans =
!!! - - Iribaren number:   0.53 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:7.1 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          1.71693977662046
!   Berm_width is greater than 1/4 wave length
!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on berm
fore_H0 =
             -1.951916616
{ Undefined function or variable 'upper_slope'.
Error in <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative', 'C:\FEMA-TransectAn
nt-weight:bold">TAW_iterative</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpsw
       fore_Irb=upper_slope/(sqrt(fore_H0/L0));
}
opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswell\Additional_Transects\4_taw\TAW_iterative.m',706,
length(dep)
ans =
    18
upper_slope
{ Undefined function or variable 'upper_slope'.
}
TAW_iterative_writer
tawfilename =
TAW_iterative.m
fid =
     3
ans =
     0
fid =
     3
ans =
     0
fid =
     3
ans =
     0
fid2 =
     4
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Trans
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
```

```
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Trans
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
```

```
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Trans
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
```

```
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
ans =
     0
clear all
close all
format long g
diary logfiles/CM-144-1-DIARY.txt  % open a diary file to record calculations
diary on         % begin recording
% FEMA appeal for The Town of Harpswell, Cumberland county, Maine
% TRANSECT ID: CM-158-1
% calculation by SJH, Ransom Consulting, Inc. 26-Feb-2020
% 100-year wave runup using TAW methodology
% including berm and weighted average with foreshore if necessary
%
% chk nld 20200220
%
% This script assumes that the incident wave conditions provided
% as input in the configuration section below are the
% appropriate values located at the end of the foreshore
% or toe of the slope on which the run-up is being calculated
% the script does not attempt to apply a depth limit or any other
% transformation to the incident wave conditions other than
% conversion of the peak wave period to the spectral mean wave
% as recommended in the references below
%
% references:
%
% Van der Meer, J.W., 2002. Technical Report Wave Run-up and
% Wave Overtopping at Dikes. TAW Technical Advisory Committee on
% Flood Defence, The Netherlands.
%
% FEMA. 2007,  Atlantic Ocean and Gulf of Mexico Coastal Guidelines Update
%
%
```

```
%
%------------------------------------------------------------------------
% CONFIG
%------------------------------------------------------------------------
fname='inpfiles/CM-158-1sta_ele_include.csv';   % file with station, elevation, include
                                    % third colunm is 0 for excluded points
imgname='logfiles/CM-158-1-runup';
SWEL=8.816;  % 100-yr still water level including wave setup.
H0=1.833;    % significant wave height at toe of structure
Tp=2.6668;    % peak period, 1/fma,
T0=Tp/1.1;
gamma_berm=1;   % this may get changed automatically below
gamma_rough=1;
gamma_beta=1;
gamma_perm=1;
setupAtToe=-0.0012402;
maxSetup=0.06101;   % only used in case of berm/shallow foreshore weighted average
plotTitle='Iterative TAW for CM-158-1'
plotTitle =
Iterative TAW for CM-158-1
% END CONFIG
%------------------------------
SWEL=SWEL+setupAtToe
SWEL =
                8.8147598
SWEL_fore=SWEL+maxSetup
SWEL_fore =
                8.8757698
% FIND WAVELENGTH USING DEEPWATER DISPERSION RELATION
% using English units
L0=32.15/(2*pi)*T0^2
L0 =
          30.0743764412918
% Find Hb (Munk, 1949)
%Hb=H0/(3.3*(H0/L0)^(1/3))
%Db=-Hb/.78+SWEL;  % depth at breaking
% The toe elevation here is only used to determine the average
% structure slope, it is not used to depth limit the wave height.
% Any depth limiting or other modification of the wave height
% to make it consitent with TAW guidance should be performed
% prior to the input of the significant wave height given above.
Ztoe=SWEL-1.5*H0
Ztoe =
                6.0652598
% read the transect
[sta,dep,inc] = textread(fname,'%n%n%n%*[^\n]','delimiter',',','headerlines',0);
% remove unselected points
k=find(inc==0);
sta(k)=[];
dep(k)=[];
sta_org=sta;  % used for plotting purposes
dep_org=dep;
% initial guess at maximum run-up elevation to estimate slope
Z2=SWEL+1.5*H0
Z2 =
               11.5642598
% determine station at the max runup and -1.5*H0 (i.e. the toe)
top_sta=-999;
toe_sta=-999;
for kk=1:length(sta)-1
    if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
       top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
    end
    if  ((Ztoe > dep(kk)) & (Ztoe <= dep(kk+1)))    % here is the intersection of Ztoe with profile
       toe_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Ztoe)
    end
end
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
% check to make sure we got them, if not extend the end slopes outward
S=diff(dep)./diff(sta);
if toe_sta==-999
   dy=dep(1)-Ztoe;
   toe_sta=sta(1)-dy/S(1)
end
if top_sta==-999
   dy=Z2-dep(end);
   top_sta=sta(end)+dy/S(end)
end
% just so the reader can tell the values aren't -999 anymore
top_sta
top_sta =
          25.2593016206604
toe_sta
toe_sta =
          13.0058230890364
% check for case where the toe of slope is below SWL-1.5*H0
% in this case interpolate setup from the setupAtToe(really setup as first station), and the max setup
```

```
% also un-include points seaward of SWL-1.5*H0
if Ztoe > dep(1)
   dd=SWEL_fore-dep;
   k=find(dd<0,1); % k is index of first land point
   staAtSWL=interp1(dep(k-1:k),sta(k-1:k),SWEL_fore);
   dsta=staAtSWL-sta(1);
   dsetup=maxSetup-setupAtToe;
   dsetdsta=dsetup/dsta;
   setup=setupAtToe+dsetdsta*(toe_sta-sta(1));
   sprintf('-!!- Location of SWEL-1.5*H0 is %4.1f ft landward of toe of slope',dsta)
   sprintf('-!!- Setup is interpolated between setup at toe of slope and max setup')
   sprintf('-!!-       setup is adjusted to %4.2f feet',setup)
   SWEL=SWEL-setupAtToe+setup;
   sprintf('-!!-       SWEL is adjusted to %4.2f feet',SWEL)
   k=find(dep < SWEL-1.5*H0)
   sta(k)=[];
   dep(k)=[];
else
   sprintf('-!!- The User has selected a starting point that is %4.2f feet above the elevation of SWEL-1.5H0\n',dep(1
   sprintf('-!!- This may be reasonable for some cases.  However the user may want to consider:\n')
   sprintf('-!!-   1) Selecting a starting point that is at or below %4.2f feet elevation, or\n', Ztoe)
   sprintf('-!!-   2) Reducing the incident wave height to a depth limited condition.\n')
end
ans =
-!!- Location of SWEL-1.5*H0 is  9.0 ft landward of toe of slope
ans =
-!!- Setup is interpolated between setup at toe of slope and max setup
ans =
-!!-       setup is adjusted to 0.03 feet
ans =
-!!-       SWEL is adjusted to 8.85 feet
k =
     1
     2
     3
     4
     5
     6
% now iterate converge on a runup elevation
tol=0.01;  % convergence criteria
R2del=999;
R2_new=3*H0; %initial guess
R2=R2_new;
iter=0;
R2_all=[];
topStaAll=[];
Berm_Segs=[];
TAW_ALWAYS_VALID=1;
while(abs(R2del) > tol && iter <= 25)
    iter=iter+1;
    sprintf ('!----------------- STARTING ITERATION %d -----------------!',iter)
    % elevation of toe of slope
    Ztoe
    % station of toe slope (relative to 0-NAVD88 shoreline
    toe_sta
    % station of top of slope/extent of 2% run-up
    top_sta
    % elevation of top of slope/extent of 2% run-up
    Z2
    % incident significant wave height
    H0
    % incident spectral peak wave period
    Tp
    % incident spectral mean wave period
    T0

    R2=R2_new
    Z2=R2+SWEL
    % determine slope for this iteration
    top_sta=-999;
    for kk=1:length(sta)-1
       if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
          top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
          break;
       end
    end
    if top_sta==-999
       dy=Z2-dep(end);
       top_sta=sta(end)+dy/S(end)
    end

    % get the length of the slope (not accounting for berm)
    Lslope=top_sta-toe_sta
    % loop over profile segments to determine berm factor
    % re-calculate influence of depth of berm based on this run-up elevation
    % check for berm, berm width, berm height
    berm_width=0;
    rdh_sum=0;
    Berm_Segs=[];
    Berm_Heights=[];
```

```
for kk=1:length(sta)-1
   ddep=dep(kk+1)-dep(kk);
   dsta=sta(kk+1)-sta(kk);
   s=ddep/dsta;
   if  (s < 1/15)     % count it as a berm if slope is flatter than 1:15 (see TAW manual)
      sprintf ('Berm Factor Calculation: Iteration %d, Profile Segment: %d',iter,kk)
      berm_width=berm_width+dsta;   % tally the width of all berm segments
      % compute the rdh for this segment and weight it by the segment length
      dh=SWEL-(dep(kk)+dep(kk+1))/2
      if  dh < 0
          chi=R2;
      else
          chi=2* H0;
      end
      if (dh <= R2 & dh >=-2*H0)
         rdh=(0.5-0.5*cos(3.14159*dh/chi)) ;
      else
         rdh=1;
      end
      rdh_sum=rdh_sum + rdh * dsta
      Berm_Segs=[Berm_Segs, kk];
      Berm_Heights=[Berm_Heights, (dep(kk)+dep(kk+1))/2];
   end
   if dep(kk) >= Z2  % jump out of loop if we reached limit of run-up for this iteration
      break
   end
end
sprintf ('!------- End Berm Factor Calculation, Iter: %d ---------!',iter)
berm_width
rB=berm_width/Lslope
if (berm_width > 0)
   rdh_mean=rdh_sum/berm_width
else
   rdh_mean=1
end
gamma_berm=1- rB * (1-rdh_mean)
if gamma_berm > 1
   gamma_berm=1
end
if gamma_berm < 0.6
   gamma_berm =0.6
end
% Iribarren number
slope=(Z2-Ztoe)/(Lslope-berm_width)
Irb=(slope/(sqrt(H0/L0)))
% runup height
gamma_berm
gamma_perm
gamma_beta
gamma_rough
gamma=gamma_berm*gamma_perm*gamma_beta*gamma_rough
% check validity
TAW_VALID=1;
if (Irb*gamma_berm < 0.5 | Irb*gamma_berm > 10 )
   sprintf('!!! - - Iribaren number: %6.2f is outside the valid range (0.5-10), TAW NOT VALID - - !!!\n', Irb*gam
   TAW_VALID=0;
else
   sprintf('!!! - - Iribaren number: %6.2f is in the valid range (0.5-10), TAW RECOMMENDED - - !!!\n', Irb*gamma_
end
islope=1/slope;
if (slope < 1/8 | slope > 1)
   sprintf('!!! - - slope: 1:%3.1f V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!\n', islope)
   TAW_VALID=0;
else
   sprintf('!!! - - slope: 1:%3.1f V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!\n', islope)
end
if TAW_VALID == 0
   TAW_ALWAYS_VALID=0;
end

if (Irb*gamma_berm < 1.8)
   R2_new=gamma*H0*1.77*Irb
else
   R2_new=gamma*H0*(4.3-(1.6/sqrt(Irb)))
end
% check to see if we need to evaluate a shallow foreshore
if berm_width > 0.25 * L0;
   disp ('!   Berm_width is greater than 1/4 wave length')
   disp ('!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on ber
   % do the foreshore calculation
   fore_H0=0.78*(SWEL_fore-min(Berm_Heights))
   % get upper slope
   fore_toe_sta=-999;
   fore_toe_dep=-999;
   for kk=length(dep)-1:-1:1
      ddep=dep(kk+1)-dep(kk);
      dsta=sta(kk+1)-sta(kk);
      s=ddep/dsta;
      if s < 1/15
         break
```

```
          end
          fore_toe_sta=sta(kk);
          fore_toe_dep=dep(kk);
          upper_slope=(Z2-fore_toe_dep)/(top_sta-fore_toe_sta)
       end
       fore_Irb=upper_slope/(sqrt(fore_H0/L0));
       fore_gamma=gamma_perm*gamma_beta*gamma_rough;
       if (fore_Irb < 1.8)
          fore_R2=fore_gamma*fore_H0*1.77*fore_Irb;
       else
          fore_R2=fore_gamma*fore_H0*(4.3-(1.6/sqrt(fore_Irb)));
       end
       if berm_width >= L0
          R2_new=fore_R2
          disp ('berm is wider than one wavelength, use full shallow foreshore solution');
       else
          w2=(berm_width-0.25*L0)/(0.75*L0)
          w1=1-w2
          R2_new=w2*fore_R2 + w1*R2_new
       end
    end % end berm width check
    % convergence criterion
    R2del=abs(R2-R2_new)
    R2_all(iter)=R2_new;
    % get the new top station (for plot purposes)
    Z2=R2_new+SWEL
    top_sta=-999;
    for kk=1:length(sta)-1
       if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
          top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
          break;
       end
    end
    if top_sta==-999
       dy=Z2-dep(end);
       top_sta=sta(end)+dy/S(end);
    end
    topStaAll(iter)=top_sta;
end
ans =
!----------------- STARTING ITERATION 1 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
Z2 =
               11.5642598
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
R2 =
                    5.499
Z2 =
          14.3483549038871
top_sta =
          81.5195591457582
Lslope =
          68.5137360567218
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 8
dh =
         -2.52887209611292
rdh_sum =
         0.437143499285073
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 9
dh =
         -2.57835809611292
rdh_sum =
         0.888333831772447
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 10
dh =
         -2.62784409611292
rdh_sum =
          1.35361000721475
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 11
dh =
         -2.67733009611292
rdh_sum =
          1.83299977765117
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 12
dh =
```

```
         -2.72681609611292
rdh_sum =
          2.32651961522747
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 13
dh =
         -2.77630209611292
rdh_sum =
          2.83417469903102
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 14
dh =
         -2.82578809611292
rdh_sum =
           3.3559589109516
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 15
dh =
         -2.87527409611292
rdh_sum =
          3.89185484057113
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 16
dh =
         -2.92476009611292
rdh_sum =
          4.44183379907843
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 17
dh =
         -2.97424609611292
rdh_sum =
          5.00585584219796
ans =
!------- End Berm Factor Calculation, Iter: 1 ---------!
berm_width =
    10
rB =
          0.14595613340544
rdh_mean =
         0.500585584219796
gamma_berm =
         0.927107402905785
slope =
         0.141558130826882
Irb =
         0.573392360930257
gamma_berm =
         0.927107402905785
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
         0.927107402905785
ans =
!!! - - Iribaren number:   0.53 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:7.1 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          1.72471636007975
!   Berm_width is greater than 1/4 wave length
!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on berm
fore_H0 =
            -1.951916616
{ Undefined function or variable 'upper_slope'.
}
TAW_iterative
plotTitle =
Iterative TAW for CM-158-1
SWEL =
                8.8147598
SWEL_fore =
                8.8757698
L0 =
         30.0743764412918
Ztoe =
                6.0652598
Z2 =
               11.5642598
toe_sta =
         13.0058230890364
top_sta =
         25.2593016206604
top_sta =
         25.2593016206604
toe_sta =
         13.0058230890364
```

```
ans =
-!!- Location of SWEL-1.5*H0 is  9.0 ft landward of toe of slope
ans =
-!!- Setup is interpolated between setup at toe of slope and max setup
ans =
-!!-       setup is adjusted to 0.03 feet
ans =
-!!-       SWEL is adjusted to 8.85 feet
k =
     1
     2
     3
     4
     5
     6
ans =
!----------------- STARTING ITERATION 1 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
Z2 =
               11.5642598
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
R2 =
                    5.499
Z2 =
          14.3483549038871
top_sta =
          81.5195591457582
Lslope =
          68.5137360567218
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 8
dh =
         -2.52887209611292
rdh_sum =
         0.437143499285073
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 9
dh =
         -2.57835809611292
rdh_sum =
         0.888333831772447
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 10
dh =
         -2.62784409611292
rdh_sum =
          1.35361000721475
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 11
dh =
         -2.67733009611292
rdh_sum =
          1.83299977765117
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 12
dh =
         -2.72681609611292
rdh_sum =
          2.32651961522747
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 13
dh =
         -2.77630209611292
rdh_sum =
          2.83417469903102
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 14
dh =
         -2.82578809611292
rdh_sum =
           3.3559589109516
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 15
dh =
         -2.87527409611292
rdh_sum =
          3.89185484057113
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 16
dh =
```

```
         -2.92476009611292
rdh_sum =
          4.44183379907843
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 17
dh =
         -2.97424609611292
rdh_sum =
          5.00585584219796
ans =
!------- End Berm Factor Calculation, Iter: 1 ---------!
berm_width =
    10
rB =
          0.14595613340544
rdh_mean =
         0.500585584219796
gamma_berm =
         0.927107402905785
slope =
         0.141558130826882
Irb =
         0.573392360930257
gamma_berm =
         0.927107402905785
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
         0.927107402905785
ans =
!!! - - Iribaren number:   0.53 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:7.1 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          1.72471636007975
!   Berm_width is greater than 1/4 wave length
!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on berm
fore_H0 =
             -1.951916616
{ Undefined function or variable 'upper_slope'.
Error in <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative', 'C:\FEMA-TransectAn
nt-weight:bold">TAW_iterative</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpsw
       fore_Irb=upper_slope/(sqrt(fore_H0/L0));
}
opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswell\Additional_Transects\4_taw\TAW_iterative.m',706,
sta(kk)
ans =
    30
dep(kk)
ans =
                11.798858
fore_to3
{ Undefined function or variable 'fore_to3'.
}
fore_toe_dep
fore_toe_dep =
  -999
fore_toe_sta
fore_toe_sta =
  -999
uiopen('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswell\Additional_Transects\4_taw\TAW_iterative.m',1)
TAW_iterative
diary on         % begin recording
fname='inpfiles/CM-158-1sta_ele_include.csv';   % file with station, elevation, include
% third colunm is 0 for excluded points
imgname='logfiles/CM-158-1-runup';
SWEL=8.816;  % 100-yr still water level including wave setup.
H0=1.833;    % significant wave height at toe of structure
Tp=2.6668;    % peak period, 1/fma,
T0=Tp/1.1;
% FIND WAVELENGTH USING DEEPWATER DISPERSION RELATION
% using English units
L0=32.15/(2*pi)*T0^2
L0 =
          30.0743764412918
% Find Hb (Munk, 1949)
%-----------------------------
SWEL=SWEL+setupAtToe
{ Undefined function or variable 'setupAtToe'.
}
fname='inpfiles/CM-158-1sta_ele_include.csv';   % file with station, elevation, include
                                    % third colunm is 0 for excluded points
imgname='logfiles/CM-158-1-runup';
SWEL=8.816;  % 100-yr still water level including wave setup.
H0=1.833;    % significant wave height at toe of structure
Tp=2.6668;    % peak period, 1/fma,
```

```
T0=Tp/1.1;
gamma_berm=1;   % this may get changed automatically below
gamma_rough=1;
gamma_beta=1;
gamma_perm=1;
setupAtToe=-0.0012402;
maxSetup=0.06101;   % only used in case of berm/shallow foreshore weighted average
plotTitle='Iterative TAW for CM-158-1'
plotTitle =
Iterative TAW for CM-158-1
% END CONFIG
%------------------------------
SWEL=SWEL+setupAtToe
SWEL =
                8.8147598
SWEL_fore=SWEL+maxSetup
SWEL_fore =
                8.8757698
% FIND WAVELENGTH USING DEEPWATER DISPERSION RELATION
% using English units
L0=32.15/(2*pi)*T0^2
L0 =
          30.0743764412918
Ztoe=SWEL-1.5*H0
Ztoe =
                6.0652598
% read the transect
[sta,dep,inc] = textread(fname,'%n%n%n%*[^\n]','delimiter',',','headerlines',0);
% remove unselected points
k=find(inc==0);
sta(k)=[];
dep(k)=[];
sta_org=sta;  % used for plotting purposes
dep_org=dep;
% initial guess at maximum run-up elevation to estimate slope
Z2=SWEL+1.5*H0
Z2 =
               11.5642598
% determine station at the max runup and -1.5*H0 (i.e. the toe)
top_sta=-999;
toe_sta=-999;
for kk=1:length(sta)-1
    if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
       top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
    end
    if  ((Ztoe > dep(kk)) & (Ztoe <= dep(kk+1)))    % here is the intersection of Ztoe with profile
       toe_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Ztoe)
    end
end
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
plot(sta)
plot(dep)
% check to make sure we got them, if not extend the end slopes outward
S=diff(dep)./diff(sta);
if toe_sta==-999
   dy=dep(1)-Ztoe;
   toe_sta=sta(1)-dy/S(1)
end
if top_sta==-999
   dy=Z2-dep(end);
   top_sta=sta(end)+dy/S(end)
end
% just so the reader can tell the values aren't -999 anymore
top_sta
top_sta =
          25.2593016206604
toe_sta
toe_sta =
          13.0058230890364
% check for case where the toe of slope is below SWL-1.5*H0
% in this case interpolate setup from the setupAtToe(really setup as first station), and the max setup
% also un-include points seaward of SWL-1.5*H0
if Ztoe > dep(1)
   dd=SWEL_fore-dep;
   k=find(dd<0,1); % k is index of first land point
   staAtSWL=interp1(dep(k-1:k),sta(k-1:k),SWEL_fore);
   dsta=staAtSWL-sta(1);
   dsetup=maxSetup-setupAtToe;
   dsetdsta=dsetup/dsta;
   setup=setupAtToe+dsetdsta*(toe_sta-sta(1));
   sprintf('-!!- Location of SWEL-1.5*H0 is %4.1f ft landward of toe of slope',dsta)
   sprintf('-!!- Setup is interpolated between setup at toe of slope and max setup')
   sprintf('-!!-       setup is adjusted to %4.2f feet',setup)
   SWEL=SWEL-setupAtToe+setup;
   sprintf('-!!-       SWEL is adjusted to %4.2f feet',SWEL)
   k=find(dep < SWEL-1.5*H0)
   sta(k)=[];
```

```
   dep(k)=[];
else
   sprintf('-!!- The User has selected a starting point that is %4.2f feet above the elevation of SWEL-1.5H0\n',dep(1
   sprintf('-!!- This may be reasonable for some cases.  However the user may want to consider:\n')
   sprintf('-!!-   1) Selecting a starting point that is at or below %4.2f feet elevation, or\n', Ztoe)
   sprintf('-!!-   2) Reducing the incident wave height to a depth limited condition.\n')
end
ans =
-!!- Location of SWEL-1.5*H0 is 27.0 ft landward of toe of slope
ans =
-!!- Setup is interpolated between setup at toe of slope and max setup
ans =
-!!-       setup is adjusted to 0.05 feet
ans =
-!!-       SWEL is adjusted to 8.87 feet
k =
     1
     2
     3
     4
     5
     6
     7
     8
     9
    10
    11
    12
    13
    14
    15
    16
    17
    18
    19
    20
    21
    22
    23
    24
% now iterate converge on a runup elevation
tol=0.01;  % convergence criteria
R2del=999;
R2_new=3*H0; %initial guess
R2=R2_new;
iter=0;
R2_all=[];
topStaAll=[];
Berm_Segs=[];
TAW_ALWAYS_VALID=1;
sprintf ('!----------------- STARTING ITERATION %d -----------------!',iter)
ans =
!----------------- STARTING ITERATION 0 -----------------!
    % elevation of toe of slope
    Ztoe
Ztoe =
                6.0652598
    % station of toe slope (relative to 0-NAVD88 shoreline
    toe_sta
toe_sta =
          13.0058230890364
    % station of top of slope/extent of 2% run-up
    top_sta
top_sta =
          25.2593016206604
    % elevation of top of slope/extent of 2% run-up
    Z2
Z2 =
               11.5642598
    % incident significant wave height
    H0
H0 =
                    1.833
    % incident spectral peak wave period
    Tp
Tp =
                   2.6668
    % incident spectral mean wave period
    T0
T0 =
          2.42436363636364

    R2=R2_new
R2 =
                    5.499
    Z2=R2+SWEL
Z2 =
          14.3667865524122
    % determine slope for this iteration
    top_sta=-999;
```

```
    for kk=1:length(sta)-1
       if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
          top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
          break;
       end
    end
    if top_sta==-999
       dy=Z2-dep(end);
       top_sta=sta(end)+dy/S(end)
    end
top_sta =
         -32.8054111085127
% get the length of the slope (not accounting for berm)
    Lslope=top_sta-toe_sta
Lslope =
         -45.8112341975491
    % loop over profile segments to determine berm factor
    % re-calculate influence of depth of berm based on this run-up elevation
    % check for berm, berm width, berm height
    berm_width=0;
    rdh_sum=0;
    Berm_Segs=[];
    Berm_Heights=[];
    for kk=1:length(sta)-1
       ddep=dep(kk+1)-dep(kk);
       dsta=sta(kk+1)-sta(kk);
       s=ddep/dsta;
       if  (s < 1/15)     % count it as a berm if slope is flatter than 1:15 (see TAW manual)
          sprintf ('Berm Factor Calculation: Iteration %d, Profile Segment: %d',iter,kk)
          berm_width=berm_width+dsta;   % tally the width of all berm segments
          % compute the rdh for this segment and weight it by the segment length
          dh=SWEL-(dep(kk)+dep(kk+1))/2
          if  dh < 0
              chi=R2;
          else
              chi=2* H0;
          end
          if (dh <= R2 & dh >=-2*H0)
             rdh=(0.5-0.5*cos(3.14159*dh/chi)) ;
          else
             rdh=1;
          end
          rdh_sum=rdh_sum + rdh * dsta
          Berm_Segs=[Berm_Segs, kk];
          Berm_Heights=[Berm_Heights, (dep(kk)+dep(kk+1))/2];
       end
       if dep(kk) >= Z2  % jump out of loop if we reached limit of run-up for this iteration
          break
       end
    end
ans =
Berm Factor Calculation: Iteration 0, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
         0.431923830491403
ans =
Berm Factor Calculation: Iteration 0, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
         0.877877093142679
ans =
Berm Factor Calculation: Iteration 0, Profile Segment: 10
dh =
         -2.60941244758778
rdh_sum =
          1.33790298328708
ans =
Berm Factor Calculation: Iteration 0, Profile Segment: 11
dh =
         -2.65889844758778
rdh_sum =
          1.81203344910653
ans =
Berm Factor Calculation: Iteration 0, Profile Segment: 12
dh =
         -2.70838444758778
rdh_sum =
          2.30028916609791
ans =
Berm Factor Calculation: Iteration 0, Profile Segment: 13
dh =
         -2.75787044758778
rdh_sum =
          2.80267952054878
ans =
Berm Factor Calculation: Iteration 0, Profile Segment: 14
dh =
         -2.80735644758778
```

```
rdh_sum =
          3.31920260203565
ans =
Berm Factor Calculation: Iteration 0, Profile Segment: 15
dh =
         -2.85684244758778
rdh_sum =
          3.84984520495081
ans =
Berm Factor Calculation: Iteration 0, Profile Segment: 16
dh =
         -2.90632844758778
rdh_sum =
          4.39458283905656
ans =
Berm Factor Calculation: Iteration 0, Profile Segment: 17
dh =
         -2.95581444758778
rdh_sum =
          4.95337974905832
ans =
Berm Factor Calculation: Iteration 0, Profile Segment: 18
dh =
         -3.00530044758779
rdh_sum =
          5.52618894318109
ans =
Berm Factor Calculation: Iteration 0, Profile Segment: 19
dh =
         -3.01099444758778
rdh_sum =
          6.10060690908255
sprintf ('!------- End Berm Factor Calculation, Iter: %d ---------!',iter)
ans =
!------- End Berm Factor Calculation, Iter: 0 ---------!
    berm_width
berm_width =
    12
    rB=berm_width/Lslope
rB =
        -0.261944481745528
    if (berm_width > 0)
       rdh_mean=rdh_sum/berm_width
    else
       rdh_mean=1
    end
rdh_mean =
         0.508383909090213
    gamma_berm=1- rB * (1-rdh_mean)
gamma_berm =
          1.12877612215113
    if gamma_berm > 1
       gamma_berm=1
    end
gamma_berm =
     1
    if gamma_berm < 0.6
       gamma_berm =0.6
    end
    % Iribarren number
    slope=(Z2-Ztoe)/(Lslope-berm_width)
slope =
        -0.143597120311335
    Irb=(slope/(sqrt(H0/L0)))
Irb =
        -0.581651448469581
    % runup height
    gamma_berm
gamma_berm =
     1
    gamma_perm
gamma_perm =
     1
    gamma_beta
gamma_beta =
     1
    gamma_rough
gamma_rough =
     1
    gamma=gamma_berm*gamma_perm*gamma_beta*gamma_rough
gamma =
     1
% check validity
    TAW_VALID=1;
    if (Irb*gamma_berm < 0.5 | Irb*gamma_berm > 10 )
       sprintf('!!! - - Iribaren number: %6.2f is outside the valid range (0.5-10), TAW NOT VALID - - !!!\n', Irb*gam
       TAW_VALID=0;
    else
       sprintf('!!! - - Iribaren number: %6.2f is in the valid range (0.5-10), TAW RECOMMENDED - - !!!\n', Irb*gamma_
    end
```

```
ans =
!!! - - Iribaren number:  -0.58 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
    islope=1/slope;
    if (slope < 1/8 | slope > 1)
       sprintf('!!! - - slope: 1:%3.1f V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!\n', islope)
       TAW_VALID=0;
    else
       sprintf('!!! - - slope: 1:%3.1f V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!\n', islope)
    end
ans =
!!! - - slope: 1:-7.0 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
    if TAW_VALID == 0
       TAW_ALWAYS_VALID=0;
    end

    if (Irb*gamma_berm < 1.8)
       R2_new=gamma*H0*1.77*Irb
    else
       R2_new=gamma*H0*(4.3-(1.6/sqrt(Irb)))
    end
R2_new =
        -1.88711577592919
berm_width
berm_width =
    12
.25*L0
ans =
         7.51859411032294
    if berm_width > 0.25 * L0;
       disp ('!   Berm_width is greater than 1/4 wave length')
       disp ('!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on ber
       % do the foreshore calculation
       fore_H0=0.78*(SWEL_fore-min(Berm_Heights))
       % get upper slope
       fore_toe_sta=-999;
       fore_toe_dep=-999;
       for kk=length(dep)-1:-1:1
          ddep=dep(kk+1)-dep(kk);
          dsta=sta(kk+1)-sta(kk);
          s=ddep/dsta;
          if s < 1/15
             break
          end
          fore_toe_sta=sta(kk);
          fore_toe_dep=dep(kk);
          upper_slope=(Z2-fore_toe_dep)/(top_sta-fore_toe_sta)
       end
       fore_Irb=upper_slope/(sqrt(fore_H0/L0));
       fore_gamma=gamma_perm*gamma_beta*gamma_rough;
       if (fore_Irb < 1.8)
          fore_R2=fore_gamma*fore_H0*1.77*fore_Irb;
       else
          fore_R2=fore_gamma*fore_H0*(4.3-(1.6/sqrt(fore_Irb)));
       end
       if berm_width >= L0
          R2_new=fore_R2
          disp ('berm is wider than one wavelength, use full shallow foreshore solution');
       else
          w2=(berm_width-0.25*L0)/(0.75*L0)
          w1=1-w2
          R2_new=w2*fore_R2 + w1*R2_new
       end
    end % end berm width check
!   Berm_width is greater than 1/4 wave length
!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on berm
fore_H0 =
            -1.951916616
{ Undefined function or variable 'upper_slope'.
}
% check to see if we need to evaluate a shallow foreshore
    if berm_width > 0.25 * L0;
       disp ('!   Berm_width is greater than 1/4 wave length')
       disp ('!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on ber
       % do the foreshore calculation
       fore_H0=0.78*(SWEL_fore-min(Berm_Heights))
       % get upper slope
       fore_toe_sta=-999;
       fore_toe_dep=-999;
       for kk=length(dep)-1:-1:1
          ddep=dep(kk+1)-dep(kk);
          dsta=sta(kk+1)-sta(kk);
          s=ddep/dsta;
          fore_toe_sta=sta(kk);
          fore_toe_dep=dep(kk);
          upper_slope=(Z2-fore_toe_dep)/(top_sta-fore_toe_sta)
          if s < 1/15
             break
          end
       end
       fore_Irb=upper_slope/(sqrt(fore_H0/L0));
```

```
       fore_gamma=gamma_perm*gamma_beta*gamma_rough;
       if (fore_Irb < 1.8)
          fore_R2=fore_gamma*fore_H0*1.77*fore_Irb;
       else
          fore_R2=fore_gamma*fore_H0*(4.3-(1.6/sqrt(fore_Irb)));
       end
       if berm_width >= L0
          R2_new=fore_R2
          disp ('berm is wider than one wavelength, use full shallow foreshore solution');
       else
          w2=(berm_width-0.25*L0)/(0.75*L0)
          w1=1-w2
          R2_new=w2*fore_R2 + w1*R2_new
       end
    end % end berm width check
!   Berm_width is greater than 1/4 wave length
!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on berm
fore_H0 =
            -1.951916616
upper_slope =
       -0.0380980000000015
w2 =
         0.198681022184016
w1 =
         0.801318977815984
R2_new =
           -1.512181684588 -     0.102650430677669i
clear all
close all
format long g
diary logfiles/CM-144-1-DIARY.txt  % open a diary file to record calculations
diary on         % begin recording
% FEMA appeal for The Town of Harpswell, Cumberland county, Maine
% TRANSECT ID: CM-158-1
% calculation by SJH, Ransom Consulting, Inc. 26-Feb-2020
% 100-year wave runup using TAW methodology
% including berm and weighted average with foreshore if necessary
%
% chk nld 20200220
%
% This script assumes that the incident wave conditions provided
% as input in the configuration section below are the
% appropriate values located at the end of the foreshore
% or toe of the slope on which the run-up is being calculated
% the script does not attempt to apply a depth limit or any other
% transformation to the incident wave conditions other than
% conversion of the peak wave period to the spectral mean wave
% as recommended in the references below
%
% references:
%
% Van der Meer, J.W., 2002. Technical Report Wave Run-up and
% Wave Overtopping at Dikes. TAW Technical Advisory Committee on
% Flood Defence, The Netherlands.
%
% FEMA. 2007,  Atlantic Ocean and Gulf of Mexico Coastal Guidelines Update
%
%
%
%------------------------------------------------------------------------
% CONFIG
%------------------------------------------------------------------------
fname='inpfiles/CM-158-1sta_ele_include.csv';   % file with station, elevation, include
                                     % third colunm is 0 for excluded points
imgname='logfiles/CM-158-1-runup';
SWEL=8.816;  % 100-yr still water level including wave setup.
H0=1.833;    % significant wave height at toe of structure
Tp=2.6668;    % peak period, 1/fma,
T0=Tp/1.1;
gamma_berm=1;   % this may get changed automatically below
gamma_rough=1;
gamma_beta=1;
gamma_perm=1;
setupAtToe=-0.0012402;
maxSetup=0.06101;   % only used in case of berm/shallow foreshore weighted average
plotTitle='Iterative TAW for CM-158-1'
plotTitle =
Iterative TAW for CM-158-1
% END CONFIG
%------------------------------
SWEL=SWEL+setupAtToe
SWEL =
                8.8147598
SWEL_fore=SWEL+maxSetup
SWEL_fore =
                8.8757698
% FIND WAVELENGTH USING DEEPWATER DISPERSION RELATION
% using English units
L0=32.15/(2*pi)*T0^2
L0 =
```

```
          30.0743764412918
% Find Hb (Munk, 1949)
%Hb=H0/(3.3*(H0/L0)^(1/3))
%Db=-Hb/.78+SWEL;  % depth at breaking
% The toe elevation here is only used to determine the average
% structure slope, it is not used to depth limit the wave height.
% Any depth limiting or other modification of the wave height
% to make it consitent with TAW guidance should be performed
% prior to the input of the significant wave height given above.
Ztoe=SWEL-1.5*H0
Ztoe =
                  6.0652598
% read the transect
[sta,dep,inc] = textread(fname,'%n%n%n%*[^\n]','delimiter',',','headerlines',0);
% remove unselected points
k=find(inc==0);
sta(k)=[];
dep(k)=[];
sta_org=sta;  % used for plotting purposes
dep_org=dep;
% initial guess at maximum run-up elevation to estimate slope
Z2=SWEL+1.5*H0
Z2 =
                11.5642598
% determine station at the max runup and -1.5*H0 (i.e. the toe)
top_sta=-999;
toe_sta=-999;
for kk=1:length(sta)-1
    if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
       top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
    end
    if  ((Ztoe > dep(kk)) & (Ztoe <= dep(kk+1)))    % here is the intersection of Ztoe with profile
       toe_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Ztoe)
    end
end
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
% check to make sure we got them, if not extend the end slopes outward
S=diff(dep)./diff(sta);
if toe_sta==-999
   dy=dep(1)-Ztoe;
   toe_sta=sta(1)-dy/S(1)
end
if top_sta==-999
   dy=Z2-dep(end);
   top_sta=sta(end)+dy/S(end)
end
% just so the reader can tell the values aren't -999 anymore
top_sta
top_sta =
          25.2593016206604
toe_sta
toe_sta =
          13.0058230890364
% check for case where the toe of slope is below SWL-1.5*H0
% in this case interpolate setup from the setupAtToe(really setup as first station), and the max setup
% also un-include points seaward of SWL-1.5*H0
if Ztoe > dep(1)
   dd=SWEL_fore-dep;
   k=find(dd<0,1); % k is index of first land point
   staAtSWL=interp1(dep(k-1:k),sta(k-1:k),SWEL_fore);
   dsta=staAtSWL-sta(1);
   dsetup=maxSetup-setupAtToe;
   dsetdsta=dsetup/dsta;
   setup=setupAtToe+dsetdsta*(toe_sta-sta(1));
   sprintf('-!!- Location of SWEL-1.5*H0 is %4.1f ft landward of toe of slope',dsta)
   sprintf('-!!- Setup is interpolated between setup at toe of slope and max setup')
   sprintf('-!!-       setup is adjusted to %4.2f feet',setup)
   SWEL=SWEL-setupAtToe+setup;
   sprintf('-!!-       SWEL is adjusted to %4.2f feet',SWEL)
   k=find(dep < SWEL-1.5*H0)
   sta(k)=[];
   dep(k)=[];
else
   sprintf('-!!- The User has selected a starting point that is %4.2f feet above the elevation of SWEL-1.5H0\n',dep(1
   sprintf('-!!- This may be reasonable for some cases.  However the user may want to consider:\n')
   sprintf('-!!-   1) Selecting a starting point that is at or below %4.2f feet elevation, or\n', Ztoe)
   sprintf('-!!-   2) Reducing the incident wave height to a depth limited condition.\n')
end
ans =
-!!- Location of SWEL-1.5*H0 is 27.0 ft landward of toe of slope
ans =
-!!- Setup is interpolated between setup at toe of slope and max setup
ans =
-!!-       setup is adjusted to 0.05 feet
ans =
-!!-       SWEL is adjusted to 8.87 feet
k =
```

```
% now iterate converge on a runup elevation
tol=0.01;  % convergence criteria
R2del=999;
R2_new=3*H0; %initial guess
R2=R2_new;
iter=0;
R2_all=[];
topStaAll=[];
Berm_Segs=[];
TAW_ALWAYS_VALID=1;
while(abs(R2del) > tol && iter <= 25)
    iter=iter+1;
    sprintf ('!----------------- STARTING ITERATION %d -----------------!',iter)
    % elevation of toe of slope
    Ztoe
    % station of toe slope (relative to 0-NAVD88 shoreline
    toe_sta
    % station of top of slope/extent of 2% run-up
    top_sta
    % elevation of top of slope/extent of 2% run-up
    Z2
    % incident significant wave height
    H0
    % incident spectral peak wave period
    Tp
    % incident spectral mean wave period
    T0

    R2=R2_new
    Z2=R2+SWEL
    % determine slope for this iteration
    top_sta=-999;
    for kk=1:length(sta)-1
       if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
          top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
          break;
       end
    end
    if top_sta==-999
       dy=Z2-dep(end);
       top_sta=sta(end)+dy/S(end)
    end

    % get the length of the slope (not accounting for berm)
    Lslope=top_sta-toe_sta
    % loop over profile segments to determine berm factor
    % re-calculate influence of depth of berm based on this run-up elevation
    % check for berm, berm width, berm height
    berm_width=0;
    rdh_sum=0;
    Berm_Segs=[];
    Berm_Heights=[];
    for kk=1:length(sta)-1
       ddep=dep(kk+1)-dep(kk);
       dsta=sta(kk+1)-sta(kk);
       s=ddep/dsta;
       if  (s < 1/15)     % count it as a berm if slope is flatter than 1:15 (see TAW manual)
          sprintf ('Berm Factor Calculation: Iteration %d, Profile Segment: %d',iter,kk)
          berm_width=berm_width+dsta;   % tally the width of all berm segments
          % compute the rdh for this segment and weight it by the segment length
          dh=SWEL-(dep(kk)+dep(kk+1))/2
          if  dh < 0
              chi=R2;
          else
              chi=2* H0;
          end
          if (dh <= R2 & dh >=-2*H0)
             rdh=(0.5-0.5*cos(3.14159*dh/chi)) ;
          else
             rdh=1;
          end
          rdh_sum=rdh_sum + rdh * dsta
          Berm_Segs=[Berm_Segs, kk];
          Berm_Heights=[Berm_Heights, (dep(kk)+dep(kk+1))/2];
```

```
   end
   if dep(kk) >= Z2  % jump out of loop if we reached limit of run-up for this iteration
      break
   end
end
sprintf ('!------- End Berm Factor Calculation, Iter: %d ---------!',iter)
berm_width
rB=berm_width/Lslope
if (berm_width > 0)
   rdh_mean=rdh_sum/berm_width
else
   rdh_mean=1
end
gamma_berm=1- rB * (1-rdh_mean)
if gamma_berm > 1
   gamma_berm=1
end
if gamma_berm < 0.6
   gamma_berm =0.6
end
% Iribarren number
slope=(Z2-Ztoe)/(Lslope-berm_width)
Irb=(slope/(sqrt(H0/L0)))
% runup height
gamma_berm
gamma_perm
gamma_beta
gamma_rough
gamma=gamma_berm*gamma_perm*gamma_beta*gamma_rough
% check validity
TAW_VALID=1;
if (Irb*gamma_berm < 0.5 | Irb*gamma_berm > 10 )
   sprintf('!!! - - Iribaren number: %6.2f is outside the valid range (0.5-10), TAW NOT VALID - - !!!\n', Irb*gam
   TAW_VALID=0;
else
   sprintf('!!! - - Iribaren number: %6.2f is in the valid range (0.5-10), TAW RECOMMENDED - - !!!\n', Irb*gamma_
end
islope=1/slope;
if (slope < 1/8 | slope > 1)
   sprintf('!!! - - slope: 1:%3.1f V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!\n', islope)
   TAW_VALID=0;
else
   sprintf('!!! - - slope: 1:%3.1f V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!\n', islope)
end
if TAW_VALID == 0
   TAW_ALWAYS_VALID=0;
end

if (Irb*gamma_berm < 1.8)
   R2_new=gamma*H0*1.77*Irb
else
   R2_new=gamma*H0*(4.3-(1.6/sqrt(Irb)))
end
% check to see if we need to evaluate a shallow foreshore
if berm_width > 0.25 * L0;
   disp ('!   Berm_width is greater than 1/4 wave length')
   disp ('!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on ber
   % do the foreshore calculation
   fore_H0=0.78*(SWEL_fore-min(Berm_Heights))
   % get upper slope
   fore_toe_sta=-999;
   fore_toe_dep=-999;
   for kk=length(dep)-1:-1:1
      ddep=dep(kk+1)-dep(kk);
      dsta=sta(kk+1)-sta(kk);
      s=ddep/dsta;
      if s < 1/15
         break
      end
      fore_toe_sta=sta(kk);
      fore_toe_dep=dep(kk);
      upper_slope=(Z2-fore_toe_dep)/(top_sta-fore_toe_sta)
   end
   fore_Irb=upper_slope/(sqrt(fore_H0/L0));
   fore_gamma=gamma_perm*gamma_beta*gamma_rough;
   if (fore_Irb < 1.8)
      fore_R2=fore_gamma*fore_H0*1.77*fore_Irb;
   else
      fore_R2=fore_gamma*fore_H0*(4.3-(1.6/sqrt(fore_Irb)));
   end
   if berm_width >= L0
      R2_new=fore_R2
      disp ('berm is wider than one wavelength, use full shallow foreshore solution');
   else
      w2=(berm_width-0.25*L0)/(0.75*L0)
      w1=1-w2
      R2_new=w2*fore_R2 + w1*R2_new
   end
end % end berm width check
% convergence criterion
```

```
    R2del=abs(R2-R2_new)
    R2_all(iter)=R2_new;
    % get the new top station (for plot purposes)
    Z2=R2_new+SWEL
    top_sta=-999;
    for kk=1:length(sta)-1
       if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
          top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
          break;
       end
    end
    if top_sta==-999
       dy=Z2-dep(end);
       top_sta=sta(end)+dy/S(end);
    end
    topStaAll(iter)=top_sta;
end
ans =
!----------------- STARTING ITERATION 1 -----------------!
Ztoe =
                6.0652598
toe_sta =
         13.0058230890364
top_sta =
         25.2593016206604
Z2 =
               11.5642598
H0 =
                    1.833
Tp =
                   2.6668
T0 =
         2.42436363636364
R2 =
                    5.499
Z2 =
         14.3667865524122
top_sta =
        -32.8054111085127
Lslope =
        -45.8112341975491
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 8
dh =
        -2.51044044758778
rdh_sum =
        0.431923830491403
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 9
dh =
        -2.55992644758778
rdh_sum =
        0.877877093142679
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 10
dh =
        -2.60941244758778
rdh_sum =
         1.33790298328708
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 11
dh =
        -2.65889844758778
rdh_sum =
         1.81203344910653
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 12
dh =
        -2.70838444758778
rdh_sum =
         2.30028916609791
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 13
dh =
        -2.75787044758778
rdh_sum =
         2.80267952054878
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 14
dh =
        -2.80735644758778
rdh_sum =
         3.31920260203565
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 15
dh =
        -2.85684244758778
rdh_sum =
         3.84984520495081
ans =
```

```
Berm Factor Calculation: Iteration 1, Profile Segment: 16
dh =
         -2.90632844758778
rdh_sum =
          4.39458283905656
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 17
dh =
         -2.95581444758778
rdh_sum =
          4.95337974905832
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 18
dh =
         -3.00530044758779
rdh_sum =
          5.52618894318109
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 19
dh =
         -3.01099444758778
rdh_sum =
          6.10060690908255
ans =
!------- End Berm Factor Calculation, Iter: 1 ---------!
berm_width =
    12
rB =
        -0.261944481745528
rdh_mean =
         0.508383909090213
gamma_berm =
          1.12877612215113
gamma_berm =
     1
slope =
        -0.143597120311335
Irb =
        -0.581651448469581
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:  -0.58 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:-7.0 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
         -1.88711577592919
!   Berm_width is greater than 1/4 wave length
!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on berm
fore_H0 =
              -1.951916616
{ Undefined function or variable 'upper_slope'.
}
clear all
close all
format long g
diary logfiles/CM-144-1-DIARY.txt  % open a diary file to record calculations
diary on         % begin recording
% FEMA appeal for The Town of Harpswell, Cumberland county, Maine
% TRANSECT ID: CM-158-1
% calculation by SJH, Ransom Consulting, Inc. 26-Feb-2020
% 100-year wave runup using TAW methodology
% including berm and weighted average with foreshore if necessary
%
% chk nld 20200220
%
% This script assumes that the incident wave conditions provided
% as input in the configuration section below are the
% appropriate values located at the end of the foreshore
% or toe of the slope on which the run-up is being calculated
% the script does not attempt to apply a depth limit or any other
% transformation to the incident wave conditions other than
% conversion of the peak wave period to the spectral mean wave
% as recommended in the references below
%
% references:
%
% Van der Meer, J.W., 2002. Technical Report Wave Run-up and
% Wave Overtopping at Dikes. TAW Technical Advisory Committee on
% Flood Defence, The Netherlands.
%
% FEMA. 2007,  Atlantic Ocean and Gulf of Mexico Coastal Guidelines Update
```

```
%
%
%
%------------------------------------------------------------------------
% CONFIG
%------------------------------------------------------------------------
fname='inpfiles/CM-158-1sta_ele_include.csv';   % file with station, elevation, include
                                    % third colunm is 0 for excluded points
imgname='logfiles/CM-158-1-runup';
SWEL=8.816;  % 100-yr still water level including wave setup.
H0=1.833;    % significant wave height at toe of structure
Tp=2.6668;    % peak period, 1/fma,
T0=Tp/1.1;
gamma_berm=1;   % this may get changed automatically below
gamma_rough=1;
gamma_beta=1;
gamma_perm=1;
setupAtToe=-0.0012402;
maxSetup=0.06101;   % only used in case of berm/shallow foreshore weighted average
plotTitle='Iterative TAW for CM-158-1'
plotTitle =
Iterative TAW for CM-158-1
% END CONFIG
%------------------------------
SWEL=SWEL+setupAtToe
SWEL =
                8.8147598
SWEL_fore=SWEL+maxSetup
SWEL_fore =
                8.8757698
% FIND WAVELENGTH USING DEEPWATER DISPERSION RELATION
% using English units
L0=32.15/(2*pi)*T0^2
L0 =
          30.0743764412918
% Find Hb (Munk, 1949)
%Hb=H0/(3.3*(H0/L0)^(1/3))
%Db=-Hb/.78+SWEL;  % depth at breaking
% The toe elevation here is only used to determine the average
% structure slope, it is not used to depth limit the wave height.
% Any depth limiting or other modification of the wave height
% to make it consitent with TAW guidance should be performed
% prior to the input of the significant wave height given above.
Ztoe=SWEL-1.5*H0
Ztoe =
                6.0652598
% read the transect
[sta,dep,inc] = textread(fname,'%n%n%n%*[^\n]','delimiter',',','headerlines',0);
% remove unselected points
k=find(inc==0);
sta(k)=[];
dep(k)=[];
sta_org=sta;  % used for plotting purposes
dep_org=dep;
% initial guess at maximum run-up elevation to estimate slope
Z2=SWEL+1.5*H0
Z2 =
               11.5642598
% determine station at the max runup and -1.5*H0 (i.e. the toe)
top_sta=-999;
toe_sta=-999;
for kk=1:length(sta)-1
    if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
       top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
    end
    if  ((Ztoe > dep(kk)) & (Ztoe <= dep(kk+1)))    % here is the intersection of Ztoe with profile
       toe_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Ztoe)
    end
end
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
% check to make sure we got them, if not extend the end slopes outward
S=diff(dep)./diff(sta);
if toe_sta==-999
   dy=dep(1)-Ztoe;
   toe_sta=sta(1)-dy/S(1)
end
if top_sta==-999
   dy=Z2-dep(end);
   top_sta=sta(end)+dy/S(end)
end
% just so the reader can tell the values aren't -999 anymore
top_sta
top_sta =
          25.2593016206604
toe_sta
toe_sta =
          13.0058230890364
```

```
% check for case where the toe of slope is below SWL-1.5*H0
% in this case interpolate setup from the setupAtToe(really setup as first station), and the max setup
% also un-include points seaward of SWL-1.5*H0
if Ztoe > dep(1)
   dd=SWEL_fore-dep;
   k=find(dd<0,1); % k is index of first land point
   staAtSWL=interp1(dep(k-1:k),sta(k-1:k),SWEL_fore);
   dsta=staAtSWL-sta(1);
   dsetup=maxSetup-setupAtToe;
   dsetdsta=dsetup/dsta;
   setup=setupAtToe+dsetdsta*(toe_sta-sta(1));
   sprintf('-!!- Location of SWEL-1.5*H0 is %4.1f ft landward of toe of slope',dsta)
   sprintf('-!!- Setup is interpolated between setup at toe of slope and max setup')
   sprintf('-!!-       setup is adjusted to %4.2f feet',setup)
   SWEL=SWEL-setupAtToe+setup;
   sprintf('-!!-       SWEL is adjusted to %4.2f feet',SWEL)
   k=find(dep < SWEL-1.5*H0)
   sta(k)=[];
   dep(k)=[];
else
   sprintf('-!!- The User has selected a starting point that is %4.2f feet above the elevation of SWEL-1.5H0\n',dep(1
   sprintf('-!!- This may be reasonable for some cases.  However the user may want to consider:\n')
   sprintf('-!!-   1) Selecting a starting point that is at or below %4.2f feet elevation, or\n', Ztoe)
   sprintf('-!!-   2) Reducing the incident wave height to a depth limited condition.\n')
end
ans =
-!!- Location of SWEL-1.5*H0 is 27.0 ft landward of toe of slope
ans =
-!!- Setup is interpolated between setup at toe of slope and max setup
ans =
-!!-       setup is adjusted to 0.05 feet
ans =
-!!-       SWEL is adjusted to 8.87 feet
k =
     1
     2
     3
     4
     5
     6
     7
     8
     9
    10
    11
    12
    13
    14
    15
% now iterate converge on a runup elevation
tol=0.01;  % convergence criteria
R2del=999;
R2_new=3*H0; %initial guess
R2=R2_new;
iter=0;
R2_all=[];
topStaAll=[];
Berm_Segs=[];
TAW_ALWAYS_VALID=1;
while(abs(R2del) > tol && iter <= 25)
    iter=iter+1;
    sprintf ('!----------------- STARTING ITERATION %d -----------------!',iter)
    % elevation of toe of slope
    Ztoe
    % station of toe slope (relative to 0-NAVD88 shoreline
    toe_sta
    % station of top of slope/extent of 2% run-up
    top_sta
    % elevation of top of slope/extent of 2% run-up
    Z2
    % incident significant wave height
    H0
    % incident spectral peak wave period
    Tp
    % incident spectral mean wave period
    T0

    R2=R2_new
    Z2=R2+SWEL
    % determine slope for this iteration
    top_sta=-999;
    for kk=1:length(sta)-1
       if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
          top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
          break;
       end
    end
    if top_sta==-999
       dy=Z2-dep(end);
       top_sta=sta(end)+dy/S(end)
```

```
end

% get the length of the slope (not accounting for berm)
Lslope=top_sta-toe_sta
% loop over profile segments to determine berm factor
% re-calculate influence of depth of berm based on this run-up elevation
% check for berm, berm width, berm height
berm_width=0;
rdh_sum=0;
Berm_Segs=[];
Berm_Heights=[];
for kk=1:length(sta)-1
   ddep=dep(kk+1)-dep(kk);
   dsta=sta(kk+1)-sta(kk);
   s=ddep/dsta;
   if  (s < 1/15)     % count it as a berm if slope is flatter than 1:15 (see TAW manual)
      sprintf ('Berm Factor Calculation: Iteration %d, Profile Segment: %d',iter,kk)
      berm_width=berm_width+dsta;   % tally the width of all berm segments
      % compute the rdh for this segment and weight it by the segment length
      dh=SWEL-(dep(kk)+dep(kk+1))/2
      if  dh < 0
          chi=R2;
      else
          chi=2* H0;
      end
      if (dh <= R2 & dh >=-2*H0)
         rdh=(0.5-0.5*cos(3.14159*dh/chi)) ;
      else
         rdh=1;
      end
      rdh_sum=rdh_sum + rdh * dsta
      Berm_Segs=[Berm_Segs, kk];
      Berm_Heights=[Berm_Heights, (dep(kk)+dep(kk+1))/2];
   end
   if dep(kk) >= Z2  % jump out of loop if we reached limit of run-up for this iteration
      break
   end
end
sprintf ('!------- End Berm Factor Calculation, Iter: %d ---------!',iter)
berm_width
rB=berm_width/Lslope
if (berm_width > 0)
   rdh_mean=rdh_sum/berm_width
else
   rdh_mean=1
end
gamma_berm=1- rB * (1-rdh_mean)
if gamma_berm > 1
   gamma_berm=1
end
if gamma_berm < 0.6
   gamma_berm =0.6
end
% Iribarren number
slope=(Z2-Ztoe)/(Lslope-berm_width)
Irb=(slope/(sqrt(H0/L0)))
% runup height
gamma_berm
gamma_perm
gamma_beta
gamma_rough
gamma=gamma_berm*gamma_perm*gamma_beta*gamma_rough
% check validity
TAW_VALID=1;
if (Irb*gamma_berm < 0.5 | Irb*gamma_berm > 10 )
   sprintf('!!! - - Iribaren number: %6.2f is outside the valid range (0.5-10), TAW NOT VALID - - !!!\n', Irb*gam
   TAW_VALID=0;
else
   sprintf('!!! - - Iribaren number: %6.2f is in the valid range (0.5-10), TAW RECOMMENDED - - !!!\n', Irb*gamma_
end
islope=1/slope;
if (slope < 1/8 | slope > 1)
   sprintf('!!! - - slope: 1:%3.1f V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!\n', islope)
   TAW_VALID=0;
else
   sprintf('!!! - - slope: 1:%3.1f V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!\n', islope)
end
if TAW_VALID == 0
   TAW_ALWAYS_VALID=0;
end

if (Irb*gamma_berm < 1.8)
   R2_new=gamma*H0*1.77*Irb
else
   R2_new=gamma*H0*(4.3-(1.6/sqrt(Irb)))
end
% check to see if we need to evaluate a shallow foreshore
if berm_width > 0.25 * L0;
   disp ('!   Berm_width is greater than 1/4 wave length')
   disp ('!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on ber
```

```
       % do the foreshore calculation
       fore_H0=0.78*(SWEL_fore-min(Berm_Heights))
       % get upper slope
       fore_toe_sta=-999;
       fore_toe_dep=-999;
       for kk=length(dep)-1:-1:1
          ddep=dep(kk+1)-dep(kk);
          dsta=sta(kk+1)-sta(kk);
          s=ddep/dsta;
          if s < 1/15
             break
          end
          fore_toe_sta=sta(kk);
          fore_toe_dep=dep(kk);
          upper_slope=(Z2-fore_toe_dep)/(top_sta-fore_toe_sta)
       end
       fore_Irb=upper_slope/(sqrt(fore_H0/L0));
       fore_gamma=gamma_perm*gamma_beta*gamma_rough;
       if (fore_Irb < 1.8)
          fore_R2=fore_gamma*fore_H0*1.77*fore_Irb;
       else
          fore_R2=fore_gamma*fore_H0*(4.3-(1.6/sqrt(fore_Irb)));
       end
       if berm_width >= L0
          R2_new=fore_R2
          disp ('berm is wider than one wavelength, use full shallow foreshore solution');
       else
          w2=(berm_width-0.25*L0)/(0.75*L0)
          w1=1-w2
          R2_new=w2*fore_R2 + w1*R2_new
       end
    end % end berm width check
    % convergence criterion
    R2del=abs(R2-R2_new)
    R2_all(iter)=R2_new;
    % get the new top station (for plot purposes)
    Z2=R2_new+SWEL
    top_sta=-999;
    for kk=1:length(sta)-1
       if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
          top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
          break;
       end
    end
    if top_sta==-999
       dy=Z2-dep(end);
       top_sta=sta(end)+dy/S(end);
    end
    topStaAll(iter)=top_sta;
end
ans =
!----------------- STARTING ITERATION 1 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
Z2 =
               11.5642598
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
R2 =
                    5.499
Z2 =
          14.3667865524122
top_sta =
         -32.8054111085127
Lslope =
         -45.8112341975491
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
          0.431923830491403
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
          0.877877093142679
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 10
dh =
         -2.60941244758778
```

```
rdh_sum =
          1.33790298328708
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 11
dh =
         -2.65889844758778
rdh_sum =
          1.81203344910653
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 12
dh =
         -2.70838444758778
rdh_sum =
          2.30028916609791
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 13
dh =
         -2.75787044758778
rdh_sum =
          2.80267952054878
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 14
dh =
         -2.80735644758778
rdh_sum =
          3.31920260203565
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 15
dh =
         -2.85684244758778
rdh_sum =
          3.84984520495081
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 16
dh =
         -2.90632844758778
rdh_sum =
          4.39458283905656
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 17
dh =
         -2.95581444758778
rdh_sum =
          4.95337974905832
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 18
dh =
         -3.00530044758779
rdh_sum =
          5.52618894318109
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 19
dh =
         -3.01099444758778
rdh_sum =
          6.10060690908255
ans =
!------- End Berm Factor Calculation, Iter: 1 ---------!
berm_width =
    12
rB =
        -0.261944481745528
rdh_mean =
         0.508383909090213
gamma_berm =
          1.12877612215113
gamma_berm =
     1
slope =
        -0.143597120311335
Irb =
        -0.581651448469581
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:  -0.58 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:-7.0 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
         -1.88711577592919
!   Berm_width is greater than 1/4 wave length
!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on berm
```

```
fore_H0 =
           -1.951916616
{ Undefined function or variable 'upper_slope'.
}
TAW_iterative
plotTitle =
Iterative TAW for CM-158-1
SWEL =
                8.8147598
SWEL_fore =
                8.8757698
L0 =
         30.0743764412918
Ztoe =
                6.0652598
Z2 =
               11.5642598
toe_sta =
         13.0058230890364
top_sta =
         25.2593016206604
top_sta =
         25.2593016206604
toe_sta =
         13.0058230890364
ans =
-!!- Location of SWEL-1.5*H0 is 27.0 ft landward of toe of slope
ans =
-!!- Setup is interpolated between setup at toe of slope and max setup
ans =
-!!-       setup is adjusted to 0.05 feet
ans =
-!!-       SWEL is adjusted to 8.87 feet
k =
     1
     2
     3
     4
     5
     6
     7
     8
     9
    10
    11
    12
    13
    14
    15
ans =
!----------------- STARTING ITERATION 1 -----------------!
Ztoe =
                6.0652598
toe_sta =
         13.0058230890364
top_sta =
         25.2593016206604
Z2 =
              11.5642598
H0 =
                   1.833
Tp =
                  2.6668
T0 =
         2.42436363636364
R2 =
                   5.499
Z2 =
         14.3667865524122
top_sta =
        -32.8054111085127
Lslope =
        -45.8112341975491
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 8
dh =
        -2.51044044758778
rdh_sum =
        0.431923830491403
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 9
dh =
        -2.55992644758778
rdh_sum =
        0.877877093142679
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 10
dh =
        -2.60941244758778
rdh_sum =
```

```
          1.33790298328708
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 11
dh =
         -2.65889844758778
rdh_sum =
          1.81203344910653
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 12
dh =
         -2.70838444758778
rdh_sum =
          2.30028916609791
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 13
dh =
         -2.75787044758778
rdh_sum =
          2.80267952054878
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 14
dh =
         -2.80735644758778
rdh_sum =
          3.31920260203565
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 15
dh =
         -2.85684244758778
rdh_sum =
          3.84984520495081
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 16
dh =
         -2.90632844758778
rdh_sum =
          4.39458283905656
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 17
dh =
         -2.95581444758778
rdh_sum =
          4.95337974905832
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 18
dh =
         -3.00530044758779
rdh_sum =
          5.52618894318109
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 19
dh =
         -3.01099444758778
rdh_sum =
          6.10060690908255
ans =
!------- End Berm Factor Calculation, Iter: 1 ---------!
berm_width =
    12
rB =
        -0.261944481745528
rdh_mean =
          0.508383909090213
gamma_berm =
          1.12877612215113
gamma_berm =
     1
slope =
        -0.143597120311335
Irb =
        -0.581651448469581
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:  -0.58 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:-7.0 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
         -1.88711577592919
!   Berm_width is greater than 1/4 wave length
!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on berm
fore_H0 =
```

```
              -1.951916616
{ Undefined function or variable ’upper_slope’.
Error in <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative’, ’C:\FEMA-TransectAn
nt-weight:bold">TAW_iterative</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpsw
       fore_Irb=upper_slope/(sqrt(fore_H0/L0));
}
TAW_iterative_writer
tawfilename =
TAW_iterative.m
fid =
     3
ans =
     0
fid =
     3
ans =
     0
fid =
     3
ans =
     0
fid2 =
     4
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Trans
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
```

```
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Trans
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
```

```
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Trans
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
ans =
     0
clear all
close all
format long g
diary logfiles/CM-144-1-DIARY.txt  % open a diary file to record calculations
diary on         % begin recording
```

```
% FEMA appeal for The Town of Harpswell, Cumberland county, Maine
% TRANSECT ID: CM-158-1
% calculation by SJH, Ransom Consulting, Inc. 26-Feb-2020
% 100-year wave runup using TAW methodology
% including berm and weighted average with foreshore if necessary
%
% chk nld 20200220
%
% This script assumes that the incident wave conditions provided
% as input in the configuration section below are the
% appropriate values located at the end of the foreshore
% or toe of the slope on which the run-up is being calculated
% the script does not attempt to apply a depth limit or any other
% transformation to the incident wave conditions other than
% conversion of the peak wave period to the spectral mean wave
% as recommended in the references below
%
% references:
%
% Van der Meer, J.W., 2002. Technical Report Wave Run-up and
% Wave Overtopping at Dikes. TAW Technical Advisory Committee on
% Flood Defence, The Netherlands.
%
% FEMA. 2007,  Atlantic Ocean and Gulf of Mexico Coastal Guidelines Update
%
%
%
%-------------------------------------------------------------------------
% CONFIG
%-------------------------------------------------------------------------
fname='inpfiles/CM-158-1sta_ele_include.csv';   % file with station, elevation, include
                                        % third colunm is 0 for excluded points
imgname='logfiles/CM-158-1-runup';
SWEL=8.816;  % 100-yr still water level including wave setup.
H0=1.833;    % significant wave height at toe of structure
Tp=2.6668;    % peak period, 1/fma,
T0=Tp/1.1;
gamma_berm=1;   % this may get changed automatically below
gamma_rough=1;
gamma_beta=1;
gamma_perm=1;
setupAtToe=-0.0012402;
maxSetup=0.06101;   % only used in case of berm/shallow foreshore weighted average
plotTitle='Iterative TAW for CM-158-1'
plotTitle =
Iterative TAW for CM-158-1
% END CONFIG
%------------------------------
SWEL=SWEL+setupAtToe
SWEL =
                8.8147598
SWEL_fore=SWEL+maxSetup
SWEL_fore =
                8.8757698
% FIND WAVELENGTH USING DEEPWATER DISPERSION RELATION
% using English units
L0=32.15/(2*pi)*T0^2
L0 =
          30.0743764412918
% Find Hb (Munk, 1949)
%Hb=H0/(3.3*(H0/L0)^(1/3))
%Db=-Hb/.78+SWEL;  % depth at breaking
% The toe elevation here is only used to determine the average
% structure slope, it is not used to depth limit the wave height.
% Any depth limiting or other modification of the wave height
% to make it consitent with TAW guidance should be performed
% prior to the input of the significant wave height given above.
Ztoe=SWEL-1.5*H0
Ztoe =
                6.0652598
% read the transect
[sta,dep,inc] = textread(fname,'%n%n%n%*[^\n]','delimiter',',','headerlines',0);
% remove unselected points
k=find(inc==0);
sta(k)=[];
dep(k)=[];
sta_org=sta;  % used for plotting purposes
dep_org=dep;
% initial guess at maximum run-up elevation to estimate slope
Z2=SWEL+1.5*H0
Z2 =
               11.5642598
% determine station at the max runup and -1.5*H0 (i.e. the toe)
top_sta=-999;
toe_sta=-999;
for kk=1:length(sta)-1
    if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
       top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
    end
    if  ((Ztoe > dep(kk)) & (Ztoe <= dep(kk+1)))    % here is the intersection of Ztoe with profile
```

```
       toe_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Ztoe)
    end
end
toe_sta =
         13.0058230890364
top_sta =
         25.2593016206604
% check to make sure we got them, if not extend the end slopes outward
S=diff(dep)./diff(sta);
if toe_sta==-999
   dy=dep(1)-Ztoe;
   toe_sta=sta(1)-dy/S(1)
end
if top_sta==-999
   dy=Z2-dep(end);
   top_sta=sta(end)+dy/S(end)
end
% just so the reader can tell the values aren't -999 anymore
top_sta
top_sta =
         25.2593016206604
toe_sta
toe_sta =
         13.0058230890364
% check for case where the toe of slope is below SWL-1.5*H0
% in this case interpolate setup from the setupAtToe(really setup as first station), and the max setup
% also un-include points seaward of SWL-1.5*H0
if Ztoe > dep(1)
   dd=SWEL_fore-dep;
   k=find(dd<0,1); % k is index of first land point
   staAtSWL=interp1(dep(k-1:k),sta(k-1:k),SWEL_fore);
   dsta=staAtSWL-sta(1);
   dsetup=maxSetup-setupAtToe;
   dsetdsta=dsetup/dsta;
   setup=setupAtToe+dsetdsta*(toe_sta-sta(1));
   sprintf('-!!- Location of SWEL-1.5*H0 is %4.1f ft landward of toe of slope',dsta)
   sprintf('-!!- Setup is interpolated between setup at toe of slope and max setup')
   sprintf('-!!-       setup is adjusted to %4.2f feet',setup)
   SWEL=SWEL-setupAtToe+setup;
   sprintf('-!!-       SWEL is adjusted to %4.2f feet',SWEL)
   k=find(dep < SWEL-1.5*H0)
   sta(k)=[];
   dep(k)=[];
else
   sprintf('-!!- The User has selected a starting point that is %4.2f feet above the elevation of SWEL-1.5H0\n',dep(1
   sprintf('-!!- This may be reasonable for some cases.  However the user may want to consider:\n')
   sprintf('-!!-   1) Selecting a starting point that is at or below %4.2f feet elevation, or\n', Ztoe)
   sprintf('-!!-   2) Reducing the incident wave height to a depth limited condition.\n')
end
ans =
-!!- Location of SWEL-1.5*H0 is 27.0 ft landward of toe of slope
ans =
-!!- Setup is interpolated between setup at toe of slope and max setup
ans =
-!!-       setup is adjusted to 0.05 feet
ans =
-!!-       SWEL is adjusted to 8.87 feet
k =
     1
     2
     3
     4
     5
     6
     7
     8
     9
    10
    11
    12
    13
    14
    15
% now iterate converge on a runup elevation
tol=0.01;  % convergence criteria
R2del=999;
R2_new=3*H0; %initial guess
R2=R2_new;
iter=0;
R2_all=[];
topStaAll=[];
Berm_Segs=[];
TAW_ALWAYS_VALID=1;
while(abs(R2del) > tol && iter <= 25)
    iter=iter+1;
    sprintf ('!----------------- STARTING ITERATION %d -----------------!',iter)
    % elevation of toe of slope
    Ztoe
    % station of toe slope (relative to 0-NAVD88 shoreline
    toe_sta
```

```
% station of top of slope/extent of 2% run-up
top_sta
% elevation of top of slope/extent of 2% run-up
Z2
% incident significant wave height
H0
% incident spectral peak wave period
Tp
% incident spectral mean wave period
T0

R2=R2_new
Z2=R2+SWEL
% determine slope for this iteration
top_sta=-999;
for kk=1:length(sta)-1
   if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
      top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
      break;
   end
end
if top_sta==-999
   dy=Z2-dep(end);
   top_sta=sta(end)+dy/S(end)
end

% get the length of the slope (not accounting for berm)
Lslope=top_sta-toe_sta
% loop over profile segments to determine berm factor
% re-calculate influence of depth of berm based on this run-up elevation
% check for berm, berm width, berm height
berm_width=0;
rdh_sum=0;
Berm_Segs=[];
Berm_Heights=[];
for kk=1:length(sta)-1
   ddep=dep(kk+1)-dep(kk);
   dsta=sta(kk+1)-sta(kk);
   s=ddep/dsta;
   if  (s < 1/15)     % count it as a berm if slope is flatter than 1:15 (see TAW manual)
      sprintf ('Berm Factor Calculation: Iteration %d, Profile Segment: %d',iter,kk)
      berm_width=berm_width+dsta;   % tally the width of all berm segments
      % compute the rdh for this segment and weight it by the segment length
      dh=SWEL-(dep(kk)+dep(kk+1))/2
      if  dh < 0
          chi=R2;
      else
          chi=2* H0;
      end
      if (dh <= R2 & dh >=-2*H0)
         rdh=(0.5-0.5*cos(3.14159*dh/chi)) ;
      else
         rdh=1;
      end
      rdh_sum=rdh_sum + rdh * dsta
      Berm_Segs=[Berm_Segs, kk];
      Berm_Heights=[Berm_Heights, (dep(kk)+dep(kk+1))/2];
   end
   if dep(kk) >= Z2  % jump out of loop if we reached limit of run-up for this iteration
      break
   end
end
sprintf ('!------- End Berm Factor Calculation, Iter: %d ---------!',iter)
berm_width
rB=berm_width/Lslope
if (berm_width > 0)
   rdh_mean=rdh_sum/berm_width
else
   rdh_mean=1
end
gamma_berm=1- rB * (1-rdh_mean)
if gamma_berm > 1
   gamma_berm=1
end
if gamma_berm < 0.6
   gamma_berm =0.6
end
% Iribarren number
slope=(Z2-Ztoe)/(Lslope-berm_width)
Irb=(slope/(sqrt(H0/L0)))
% runup height
gamma_berm
gamma_perm
gamma_beta
gamma_rough
gamma=gamma_berm*gamma_perm*gamma_beta*gamma_rough
% check validity
TAW_VALID=1;
if (Irb*gamma_berm < 0.5 | Irb*gamma_berm > 10 )
   sprintf('!!! - - Iribaren number: %6.2f is outside the valid range (0.5-10), TAW NOT VALID - - !!!\n', Irb*gam
```

```
      TAW_VALID=0;
   else
      sprintf('!!! - - Iribaren number: %6.2f is in the valid range (0.5-10), TAW RECOMMENDED - - !!!\n', Irb*gamma_
   end
   islope=1/slope;
   if (slope < 1/8 | slope > 1)
      sprintf('!!! - - slope: 1:%3.1f V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!\n', islope)
      TAW_VALID=0;
   else
      sprintf('!!! - - slope: 1:%3.1f V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!\n', islope)
   end
   if TAW_VALID == 0
      TAW_ALWAYS_VALID=0;
   end

   if (Irb*gamma_berm < 1.8)
      R2_new=gamma*H0*1.77*Irb
   else
      R2_new=gamma*H0*(4.3-(1.6/sqrt(Irb)))
   end
   % check to see if we need to evaluate a shallow foreshore
   if berm_width > 0.25 * L0;
      disp ('!   Berm_width is greater than 1/4 wave length')
      disp ('!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on ber
      % do the foreshore calculation
      fore_H0=0.78*(SWEL_fore-min(Berm_Heights))
      % get upper slope
      fore_toe_sta=-999;
      fore_toe_dep=-999;
      for kk=length(dep)-1:-1:1
         ddep=dep(kk+1)-dep(kk);
         dsta=sta(kk+1)-sta(kk);
         s=ddep/dsta;
         if s < 1/15
            break
         end
         fore_toe_sta=sta(kk);
         fore_toe_dep=dep(kk);
         upper_slope=(Z2-fore_toe_dep)/(top_sta-fore_toe_sta)
      end
      fore_Irb=upper_slope/(sqrt(fore_H0/L0));
      fore_gamma=gamma_perm*gamma_beta*gamma_rough;
      if (fore_Irb < 1.8)
         fore_R2=fore_gamma*fore_H0*1.77*fore_Irb;
      else
         fore_R2=fore_gamma*fore_H0*(4.3-(1.6/sqrt(fore_Irb)));
      end
      if berm_width >= L0
         R2_new=fore_R2
         disp ('berm is wider than one wavelength, use full shallow foreshore solution');
      else
         w2=(berm_width-0.25*L0)/(0.75*L0)
         w1=1-w2
         R2_new=w2*fore_R2 + w1*R2_new
      end
   end % end berm width check
   % convergence criterion
   R2del=abs(R2-R2_new)
   R2_all(iter)=R2_new;
   % get the new top station (for plot purposes)
   Z2=R2_new+SWEL
   top_sta=-999;
   for kk=1:length(sta)-1
      if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
         top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
         break;
      end
   end
   if top_sta==-999
      dy=Z2-dep(end);
      top_sta=sta(end)+dy/S(end);
   end
   topStaAll(iter)=top_sta;
end
ans =
!----------------- STARTING ITERATION 1 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
Z2 =
               11.5642598
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
```

```
R2 =
                    5.499
Z2 =
         14.3667865524122
top_sta =
         81.8920210243749
Lslope =
         68.8861979353385
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 8
dh =
        -2.51044044758778
rdh_sum =
         0.431923830491403
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 9
dh =
        -2.55992644758778
rdh_sum =
         0.877877093142679
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 10
dh =
        -2.60941244758778
rdh_sum =
         1.33790298328708
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 11
dh =
        -2.65889844758778
rdh_sum =
         1.81203344910653
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 12
dh =
        -2.70838444758778
rdh_sum =
         2.30028916609791
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 13
dh =
        -2.75787044758778
rdh_sum =
         2.80267952054878
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 14
dh =
        -2.80735644758778
rdh_sum =
         3.31920260203565
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 15
dh =
        -2.85684244758778
rdh_sum =
         3.84984520495081
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 16
dh =
        -2.90632844758778
rdh_sum =
         4.39458283905656
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 17
dh =
        -2.95581444758778
rdh_sum =
         4.95337974905832
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 18
dh =
        -3.00530044758779
rdh_sum =
         5.52618894318109
ans =
!-------- End Berm Factor Calculation, Iter: 1 ---------!
berm_width =
    11
rB =
        0.159683656954407
rdh_mean =
        0.502380813016463
gamma_berm =
         0.92053834845179
slope =
        0.143411159283347
Irb =
        0.580898198675621
gamma_berm =
         0.92053834845179
```

```
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
          0.92053834845179
ans =
!!! - - Iribaren number:   0.53 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:7.0 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          1.73491278099679
!   Berm_width is greater than 1/4 wave length
!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on berm
fore_H0 =
             -1.951916616
{ Undefined function or variable 'upper_slope'.
}
TAW_iterative
plotTitle =
Iterative TAW for CM-158-1
SWEL =
                8.8147598
SWEL_fore =
                8.8757698
L0 =
          30.0743764412918
Ztoe =
                6.0652598
Z2 =
               11.5642598
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
top_sta =
          25.2593016206604
toe_sta =
          13.0058230890364
ans =
-!!- Location of SWEL-1.5*H0 is 27.0 ft landward of toe of slope
ans =
-!!- Setup is interpolated between setup at toe of slope and max setup
ans =
-!!-       setup is adjusted to 0.05 feet
ans =
-!!-       SWEL is adjusted to 8.87 feet
k =
     1
     2
     3
     4
     5
     6
     7
     8
     9
    10
    11
    12
    13
    14
    15
ans =
!----------------- STARTING ITERATION 1 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
Z2 =
               11.5642598
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
R2 =
                    5.499
Z2 =
          14.3667865524122
top_sta =
          81.8920210243749
Lslope =
          68.8861979353385
ans =
```

```
Berm Factor Calculation: Iteration 1, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
         0.431923830491403
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
         0.877877093142679
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 10
dh =
         -2.60941244758778
rdh_sum =
          1.33790298328708
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 11
dh =
         -2.65889844758778
rdh_sum =
          1.81203344910653
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 12
dh =
         -2.70838444758778
rdh_sum =
          2.30028916609791
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 13
dh =
         -2.75787044758778
rdh_sum =
          2.80267952054878
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 14
dh =
         -2.80735644758778
rdh_sum =
          3.31920260203565
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 15
dh =
         -2.85684244758778
rdh_sum =
          3.84984520495081
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 16
dh =
         -2.90632844758778
rdh_sum =
          4.39458283905656
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 17
dh =
         -2.95581444758778
rdh_sum =
          4.95337974905832
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 18
dh =
         -3.00530044758779
rdh_sum =
          5.52618894318109
ans =
!------- End Berm Factor Calculation, Iter: 1 ---------!
berm_width =
    11
rB =
         0.159683656954407
rdh_mean =
         0.502380813016463
gamma_berm =
          0.92053834845179
slope =
         0.143411159283347
Irb =
         0.580898198675621
gamma_berm =
          0.92053834845179
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
          0.92053834845179
ans =
```

```
!!! - - Iribaren number:   0.53 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:7.0 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          1.73491278099679
!   Berm_width is greater than 1/4 wave length
!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on berm
fore_H0 =
             -1.951916616
{ Undefined function or variable 'upper_slope'.
Error in <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative', 'C:\FEMA-TransectAn
nt-weight:bold">TAW_iterative</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpsw
       fore_Irb=upper_slope/(sqrt(fore_H0/L0));
}
opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswell\Additional_Transects\4_taw\TAW_iterative.m',706,
clear all
close all
format long g
diary logfiles/CM-144-1-DIARY.txt  % open a diary file to record calculations
diary on         % begin recording
% FEMA appeal for The Town of Harpswell, Cumberland county, Maine
% TRANSECT ID: CM-158-1
% calculation by SJH, Ransom Consulting, Inc. 26-Feb-2020
% 100-year wave runup using TAW methodology
% including berm and weighted average with foreshore if necessary
%
% chk nld 20200220
%
% This script assumes that the incident wave conditions provided
% as input in the configuration section below are the
% appropriate values located at the end of the foreshore
% or toe of the slope on which the run-up is being calculated
% the script does not attempt to apply a depth limit or any other
% transformation to the incident wave conditions other than
% conversion of the peak wave period to the spectral mean wave
% as recommended in the references below
%
% references:
%
% Van der Meer, J.W., 2002. Technical Report Wave Run-up and
% Wave Overtopping at Dikes. TAW Technical Advisory Committee on
% Flood Defence, The Netherlands.
%
% FEMA. 2007,  Atlantic Ocean and Gulf of Mexico Coastal Guidelines Update
%
%
%
%------------------------------------------------------------------------
% CONFIG
%------------------------------------------------------------------------
fname='inpfiles/CM-158-1sta_ele_include.csv';   % file with station, elevation, include
                                    % third colunm is 0 for excluded points
imgname='logfiles/CM-158-1-runup';
SWEL=8.816;  % 100-yr still water level including wave setup.
H0=1.833;    % significant wave height at toe of structure
Tp=2.6668;    % peak period, 1/fma,
T0=Tp/1.1;
gamma_berm=1;   % this may get changed automatically below
gamma_rough=1;
gamma_beta=1;
gamma_perm=1;
setupAtToe=-0.0012402;
maxSetup=0.06101;   % only used in case of berm/shallow foreshore weighted average
plotTitle='Iterative TAW for CM-158-1'
plotTitle =
Iterative TAW for CM-158-1
% END CONFIG
%------------------------------
SWEL=SWEL+setupAtToe
SWEL =
                8.8147598
SWEL_fore=SWEL+maxSetup
SWEL_fore =
                8.8757698
% FIND WAVELENGTH USING DEEPWATER DISPERSION RELATION
% using English units
L0=32.15/(2*pi)*T0^2
L0 =
          30.0743764412918
% Find Hb (Munk, 1949)
%Hb=H0/(3.3*(H0/L0)^(1/3))
%Db=-Hb/.78+SWEL;  % depth at breaking
% The toe elevation here is only used to determine the average
% structure slope, it is not used to depth limit the wave height.
% Any depth limiting or other modification of the wave height
% to make it consitent with TAW guidance should be performed
% prior to the input of the significant wave height given above.
Ztoe=SWEL-1.5*H0
Ztoe =
                6.0652598
```

```
% read the transect
[sta,dep,inc] = textread(fname,'%n%n%n%*[^\n]','delimiter',',','headerlines',0);
% remove unselected points
k=find(inc==0);
sta(k)=[];
dep(k)=[];
sta_org=sta;  % used for plotting purposes
dep_org=dep;
% initial guess at maximum run-up elevation to estimate slope
Z2=SWEL+1.5*H0
Z2 =
               11.5642598
% determine station at the max runup and -1.5*H0 (i.e. the toe)
top_sta=-999;
toe_sta=-999;
for kk=1:length(sta)-1
    if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
       top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
    end
    if  ((Ztoe > dep(kk)) & (Ztoe <= dep(kk+1)))    % here is the intersection of Ztoe with profile
       toe_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Ztoe)
    end
end
toe_sta =
         13.0058230890364
top_sta =
         25.2593016206604
% check to make sure we got them, if not extend the end slopes outward
S=diff(dep)./diff(sta);
if toe_sta==-999
   dy=dep(1)-Ztoe;
   toe_sta=sta(1)-dy/S(1)
end
if top_sta==-999
   dy=Z2-dep(end);
   top_sta=sta(end)+dy/S(end)
end
% just so the reader can tell the values aren't -999 anymore
top_sta
top_sta =
         25.2593016206604
toe_sta
toe_sta =
         13.0058230890364
% check for case where the toe of slope is below SWL-1.5*H0
% in this case interpolate setup from the setupAtToe(really setup as first station), and the max setup
% also un-include points seaward of SWL-1.5*H0
if Ztoe > dep(1)
   dd=SWEL_fore-dep;
   k=find(dd<0,1); % k is index of first land point
   staAtSWL=interp1(dep(k-1:k),sta(k-1:k),SWEL_fore);
   dsta=staAtSWL-sta(1);
   dsetup=maxSetup-setupAtToe;
   dsetdsta=dsetup/dsta;
   setup=setupAtToe+dsetdsta*(toe_sta-sta(1));
   sprintf('-!!- Location of SWEL-1.5*H0 is %4.1f ft landward of toe of slope',dsta)
   sprintf('-!!- Setup is interpolated between setup at toe of slope and max setup')
   sprintf('-!!-       setup is adjusted to %4.2f feet',setup)
   SWEL=SWEL-setupAtToe+setup;
   sprintf('-!!-       SWEL is adjusted to %4.2f feet',SWEL)
   k=find(dep < SWEL-1.5*H0)
   sta(k)=[];
   dep(k)=[];
else
   sprintf('-!!- The User has selected a starting point that is %4.2f feet above the elevation of SWEL-1.5H0\n',dep(1
   sprintf('-!!- This may be reasonable for some cases.  However the user may want to consider:\n')
   sprintf('-!!-   1) Selecting a starting point that is at or below %4.2f feet elevation, or\n', Ztoe)
   sprintf('-!!-   2) Reducing the incident wave height to a depth limited condition.\n')
end
ans =
-!!- Location of SWEL-1.5*H0 is 27.0 ft landward of toe of slope
ans =
-!!- Setup is interpolated between setup at toe of slope and max setup
ans =
-!!-       setup is adjusted to 0.05 feet
ans =
-!!-       SWEL is adjusted to 8.87 feet
k =
     1
     2
     3
     4
     5
     6
     7
     8
     9
    10
    11
    12
```

```
    13
    14
    15
% now iterate converge on a runup elevation
tol=0.01;  % convergence criteria
R2del=999;
R2_new=3*H0; %initial guess
R2=R2_new;
iter=0;
R2_all=[];
topStaAll=[];
Berm_Segs=[];
TAW_ALWAYS_VALID=1;
while(abs(R2del) > tol && iter <= 25)
    iter=iter+1;
    sprintf ('!----------------- STARTING ITERATION %d -----------------!',iter)
    % elevation of toe of slope
    Ztoe
    % station of toe slope (relative to 0-NAVD88 shoreline
    toe_sta
    % station of top of slope/extent of 2% run-up
    top_sta
    % elevation of top of slope/extent of 2% run-up
    Z2
    % incident significant wave height
    H0
    % incident spectral peak wave period
    Tp
    % incident spectral mean wave period
    T0

    R2=R2_new
    Z2=R2+SWEL
    % determine slope for this iteration
    top_sta=-999;
    for kk=1:length(sta)-1
       if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
          top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
          break;
       end
    end
    if top_sta==-999
       dy=Z2-dep(end);
       top_sta=sta(end)+dy/S(end)
    end

    % get the length of the slope (not accounting for berm)
    Lslope=top_sta-toe_sta
    % loop over profile segments to determine berm factor
    % re-calculate influence of depth of berm based on this run-up elevation
    % check for berm, berm width, berm height
    berm_width=0;
    rdh_sum=0;
    Berm_Segs=[];
    Berm_Heights=[];
    for kk=1:length(sta)-1
       ddep=dep(kk+1)-dep(kk);
       dsta=sta(kk+1)-sta(kk);
       s=ddep/dsta;
       if  (s < 1/15)     % count it as a berm if slope is flatter than 1:15 (see TAW manual)
          sprintf ('Berm Factor Calculation: Iteration %d, Profile Segment: %d',iter,kk)
          berm_width=berm_width+dsta;   % tally the width of all berm segments
          % compute the rdh for this segment and weight it by the segment length
          dh=SWEL-(dep(kk)+dep(kk+1))/2
          if  dh < 0
              chi=R2;
          else
              chi=2* H0;
          end
          if (dh <= R2 & dh >=-2*H0)
             rdh=(0.5-0.5*cos(3.14159*dh/chi)) ;
          else
             rdh=1;
          end
          rdh_sum=rdh_sum + rdh * dsta
          Berm_Segs=[Berm_Segs, kk];
          Berm_Heights=[Berm_Heights, (dep(kk)+dep(kk+1))/2];
       end
       if dep(kk) >= Z2  % jump out of loop if we reached limit of run-up for this iteration
          break
       end
    end
    sprintf ('!------- End Berm Factor Calculation, Iter: %d ---------!',iter)
    berm_width
    rB=berm_width/Lslope
    if (berm_width > 0)
       rdh_mean=rdh_sum/berm_width
    else
       rdh_mean=1
    end
```

```
gamma_berm=1- rB * (1-rdh_mean)
if gamma_berm > 1
   gamma_berm=1
end
if gamma_berm < 0.6
   gamma_berm =0.6
end
% Iribarren number
slope=(Z2-Ztoe)/(Lslope-berm_width)
Irb=(slope/(sqrt(H0/L0)))
% runup height
gamma_berm
gamma_perm
gamma_beta
gamma_rough
gamma=gamma_berm*gamma_perm*gamma_beta*gamma_rough
% check validity
TAW_VALID=1;
if (Irb*gamma_berm < 0.5 | Irb*gamma_berm > 10 )
   sprintf('!!! - - Iribaren number: %6.2f is outside the valid range (0.5-10), TAW NOT VALID - - !!!\n', Irb*gam
   TAW_VALID=0;
else
   sprintf('!!! - - Iribaren number: %6.2f is in the valid range (0.5-10), TAW RECOMMENDED - - !!!\n', Irb*gamma_
end
islope=1/slope;
if (slope < 1/8 | slope > 1)
   sprintf('!!! - - slope: 1:%3.1f V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!\n', islope)
   TAW_VALID=0;
else
   sprintf('!!! - - slope: 1:%3.1f V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!\n', islope)
end
if TAW_VALID == 0
   TAW_ALWAYS_VALID=0;
end

if (Irb*gamma_berm < 1.8)
   R2_new=gamma*H0*1.77*Irb
else
   R2_new=gamma*H0*(4.3-(1.6/sqrt(Irb)))
end
% check to see if we need to evaluate a shallow foreshore
if berm_width > 0.25 * L0;
   disp ('!   Berm_width is greater than 1/4 wave length')
   disp ('!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on ber
   % do the foreshore calculation
   fore_H0=0.78*(SWEL_fore-min(Berm_Heights))
   % get upper slope
   fore_toe_sta=-999;
   fore_toe_dep=-999;
   for kk=length(dep)-1:-1:1
      ddep=dep(kk+1)-dep(kk);
      dsta=sta(kk+1)-sta(kk);
      s=ddep/dsta;
                fore_toe_sta=sta(kk);
      fore_toe_dep=dep(kk);
      upper_slope=(Z2-fore_toe_dep)/(top_sta-fore_toe_sta)
      if s < 1/15
         break
      end
   end
   fore_Irb=upper_slope/(sqrt(fore_H0/L0));
   fore_gamma=gamma_perm*gamma_beta*gamma_rough;
   if (fore_Irb < 1.8)
      fore_R2=fore_gamma*fore_H0*1.77*fore_Irb;
   else
      fore_R2=fore_gamma*fore_H0*(4.3-(1.6/sqrt(fore_Irb)));
   end
   if berm_width >= L0
      R2_new=fore_R2
      disp ('berm is wider than one wavelength, use full shallow foreshore solution');
   else
      w2=(berm_width-0.25*L0)/(0.75*L0)
      w1=1-w2
      R2_new=w2*fore_R2 + w1*R2_new
   end
end % end berm width check
% convergence criterion
R2del=abs(R2-R2_new)
R2_all(iter)=R2_new;
% get the new top station (for plot purposes)
Z2=R2_new+SWEL
top_sta=-999;
for kk=1:length(sta)-1
   if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
      top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
      break;
   end
end
if top_sta==-999
   dy=Z2-dep(end);
```

```
       top_sta=sta(end)+dy/S(end);
    end
    topStaAll(iter)=top_sta;
end
ans =
!----------------- STARTING ITERATION 1 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
Z2 =
               11.5642598
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
R2 =
                    5.499
Z2 =
          14.3667865524122
top_sta =
          81.8920210243749
Lslope =
          68.8861979353385
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
          0.431923830491403
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
          0.877877093142679
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 10
dh =
         -2.60941244758778
rdh_sum =
          1.33790298328708
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 11
dh =
         -2.65889844758778
rdh_sum =
          1.81203344910653
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 12
dh =
         -2.70838444758778
rdh_sum =
          2.30028916609791
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 13
dh =
         -2.75787044758778
rdh_sum =
          2.80267952054878
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 14
dh =
         -2.80735644758778
rdh_sum =
          3.31920260203565
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 15
dh =
         -2.85684244758778
rdh_sum =
          3.84984520495081
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 16
dh =
         -2.90632844758778
rdh_sum =
          4.39458283905656
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 17
dh =
         -2.95581444758778
rdh_sum =
          4.95337974905832
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 18
```

```
dh =
        -3.00530044758779
rdh_sum =
         5.52618894318109
ans =
!------- End Berm Factor Calculation, Iter: 1 ---------!
berm_width =
    11
rB =
         0.159683656954407
rdh_mean =
         0.502380813016463
gamma_berm =
          0.92053834845179
slope =
         0.143411159283347
Irb =
         0.580898198675621
gamma_berm =
          0.92053834845179
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
          0.92053834845179
ans =
!!! - - Iribaren number:   0.53 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:7.0 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
         1.73491278099679
!   Berm_width is greater than 1/4 wave length
!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on berm
fore_H0 =
              -1.951916616
upper_slope =
       0.0494859999999999
w2 =
         0.15434649255757
w1 =
         0.84565350744243
R2_new =
          1.46713507835663 +     0.103581293065431i
R2del =
         4.03319523835028
Z2 =
          10.3349216307688 +     0.103581293065431i
{ Error using <a href="matlab:matlab.internal.language.introspective.errorDocCallback('griddedInterpolant/subsref')"
Data points in complex number format are not supported.
Use REAL and IMAG to extract the real and imaginary components.
Error in <a href="matlab:matlab.internal.language.introspective.errorDocCallback('interp1', 'C:\Program Files\MATLAB\
ab: opentoline('C:\Program Files\MATLAB\R2016b\toolbox\matlab\polyfun\interp1.p',162,0)">line 162</a>)
        VqLite = F(Xqcol);
}
opentoline('C:\Program Files\MATLAB\R2016b\toolbox\matlab\polyfun\interp1.p',162,0)
TAW_iterative
plotTitle =
Iterative TAW for CM-158-1
SWEL =
                8.8147598
SWEL_fore =
                8.8757698
L0 =
         30.0743764412918
Ztoe =
                6.0652598
Z2 =
               11.5642598
toe_sta =
         13.0058230890364
top_sta =
         25.2593016206604
top_sta =
         25.2593016206604
toe_sta =
         13.0058230890364
ans =
-!!- Location of SWEL-1.5*H0 is 27.0 ft landward of toe of slope
ans =
-!!- Setup is interpolated between setup at toe of slope and max setup
ans =
-!!-       setup is adjusted to 0.05 feet
ans =
-!!-       SWEL is adjusted to 8.87 feet
k =
     1
     2
```

```
     3
     4
     5
     6
     7
     8
     9
    10
    11
    12
    13
    14
    15
ans =
!----------------- STARTING ITERATION 1 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
Z2 =
               11.5642598
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
R2 =
                    5.499
Z2 =
          14.3667865524122
top_sta =
          81.8920210243749
Lslope =
          68.8861979353385
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
          0.431923830491403
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
          0.877877093142679
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 10
dh =
         -2.60941244758778
rdh_sum =
           1.33790298328708
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 11
dh =
         -2.65889844758778
rdh_sum =
           1.81203344910653
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 12
dh =
         -2.70838444758778
rdh_sum =
           2.30028916609791
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 13
dh =
         -2.75787044758778
rdh_sum =
           2.80267952054878
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 14
dh =
         -2.80735644758778
rdh_sum =
           3.31920260203565
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 15
dh =
         -2.85684244758778
rdh_sum =
           3.84984520495081
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 16
dh =
         -2.90632844758778
rdh_sum =
```

```
          4.39458283905656
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 17
dh =
         -2.95581444758778
rdh_sum =
          4.95337974905832
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 18
dh =
         -3.00530044758779
rdh_sum =
          5.52618894318109
ans =
!------- End Berm Factor Calculation, Iter: 1 ---------!
berm_width =
    11
rB =
         0.159683656954407
rdh_mean =
         0.502380813016463
gamma_berm =
          0.92053834845179
slope =
         0.143411159283347
Irb =
         0.580898198675621
gamma_berm =
          0.92053834845179
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
          0.92053834845179
ans =
!!! - - Iribaren number:   0.53 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:7.0 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          1.73491278099679
!   Berm_width is greater than 1/4 wave length
!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on berm
fore_H0 =
              -1.951916616
upper_slope =
       0.0494859999999999
w2 =
         0.15434649255757
w1 =
         0.84565350744243
R2_new =
           1.46713507835663 +     0.1035812930654311i
R2del =
          4.03319523835028
Z2 =
           10.3349216307688 +     0.1035812930654311i
{ Error using <a href="matlab:matlab.internal.language.introspective.errorDocCallback('griddedInterpolant/subsref')"
Data points in complex number format are not supported.
Use REAL and IMAG to extract the real and imaginary components.
Error in <a href="matlab:matlab.internal.language.introspective.errorDocCallback('interp1', 'C:\Program Files\MATLAB\
ab: opentoline('C:\Program Files\MATLAB\R2016b\toolbox\matlab\polyfun\interp1.p',162,0)">line 162</a>)
       VqLite = F(Xqcol);
Error in <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative', 'C:\FEMA-TransectAn
nt-weight:bold">TAW_iterative</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpsw
          top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
}
opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswell\Additional_Transects\4_taw\TAW_iterative.m',733,
opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswell\Additional_Transects\4_taw\TAW_iterative.m',733,
TAW_iterative_writer
tawfilename =
TAW_iterative.m
fid =
     3
ans =
     0
fid =
     3
ans =
     0
fid =
     3
ans =
     0
fid2 =
     4
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Trans
```

```
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Trans
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
```

```
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Trans
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
```

```
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ Warning: Error: <a href="matlab: opentoline(’C:\Program Files\MATLAB\R2016b\toolbox\matlab\scribe\+matlab\+graphics
</a>One or more output arguments not assigned during call to "regexp".]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Error: <a href="matlab: opentoline(’C:\Program Files\MATLAB\R2016b\toolbox\matlab\specgraph\+matlab\+graph
</a>One or more output arguments not assigned during call to "regexp".]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
ans =
     0
TAW_iterative_writer
tawfilename =
TAW_iterative.m
fid =
     3
ans =
     0
fid =
     3
ans =
     0
fid =
     3
ans =
     0
fid2 =
     4
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Trans
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
```

```
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Trans
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
```

```
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Trans
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
```

```
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
)" style="font-weight:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-Tran
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
ans =
     0
clear all
close all
format long g
diary logfiles/CM-144-1-DIARY.txt  % open a diary file to record calculations
diary on         % begin recording
% FEMA appeal for The Town of Harpswell, Cumberland county, Maine
% TRANSECT ID: CM-158-1
% calculation by SJH, Ransom Consulting, Inc. 26-Feb-2020
% 100-year wave runup using TAW methodology
% including berm and weighted average with foreshore if necessary
%
% chk nld 20200220
%
% This script assumes that the incident wave conditions provided
% as input in the configuration section below are the
% appropriate values located at the end of the foreshore
% or toe of the slope on which the run-up is being calculated
% the script does not attempt to apply a depth limit or any other
% transformation to the incident wave conditions other than
% conversion of the peak wave period to the spectral mean wave
% as recommended in the references below
%
% references:
%
% Van der Meer, J.W., 2002. Technical Report Wave Run-up and
% Wave Overtopping at Dikes. TAW Technical Advisory Committee on
% Flood Defence, The Netherlands.
%
% FEMA. 2007,  Atlantic Ocean and Gulf of Mexico Coastal Guidelines Update
%
%
%
%--------------------------------------------------------------------------
```

```
% CONFIG
%-------------------------------------------------------------------------
fname='inpfiles/CM-158-1sta_ele_include.csv';   % file with station, elevation, include
                                        % third colunm is 0 for excluded points
imgname='logfiles/CM-158-1-runup';
SWEL=8.816;  % 100-yr still water level including wave setup.
H0=1.833;    % significant wave height at toe of structure
Tp=2.6668;    % peak period, 1/fma,
T0=Tp/1.1;
gamma_berm=1;   % this may get changed automatically below
gamma_rough=1;
gamma_beta=1;
gamma_perm=1;
setupAtToe=-0.0012402;
maxSetup=0.06101;   % only used in case of berm/shallow foreshore weighted average
plotTitle='Iterative TAW for CM-158-1'
plotTitle =
Iterative TAW for CM-158-1
% END CONFIG
%------------------------------
SWEL=SWEL+setupAtToe
SWEL =
                8.8147598
SWEL_fore=SWEL+maxSetup
SWEL_fore =
                8.8757698
% FIND WAVELENGTH USING DEEPWATER DISPERSION RELATION
% using English units
L0=32.15/(2*pi)*T0^2
L0 =
          30.0743764412918
% Find Hb (Munk, 1949)
%Hb=H0/(3.3*(H0/L0)^(1/3))
%Db=-Hb/.78+SWEL;  % depth at breaking
% The toe elevation here is only used to determine the average
% structure slope, it is not used to depth limit the wave height.
% Any depth limiting or other modification of the wave height
% to make it consitent with TAW guidance should be performed
% prior to the input of the significant wave height given above.
Ztoe=SWEL-1.5*H0
Ztoe =
                6.0652598
% read the transect
[sta,dep,inc] = textread(fname,'%n%n%n%*[^\n]','delimiter',',','headerlines',0);
% remove unselected points
k=find(inc==0);
sta(k)=[];
dep(k)=[];
sta_org=sta;  % used for plotting purposes
dep_org=dep;
% initial guess at maximum run-up elevation to estimate slope
Z2=SWEL+1.5*H0
Z2 =
               11.5642598
% determine station at the max runup and -1.5*H0 (i.e. the toe)
top_sta=-999;
toe_sta=-999;
for kk=1:length(sta)-1
    if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
       top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
    end
    if  ((Ztoe > dep(kk)) & (Ztoe <= dep(kk+1)))    % here is the intersection of Ztoe with profile
       toe_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Ztoe)
    end
end
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
% check to make sure we got them, if not extend the end slopes outward
S=diff(dep)./diff(sta);
if toe_sta==-999
   dy=dep(1)-Ztoe;
   toe_sta=sta(1)-dy/S(1)
end
if top_sta==-999
   dy=Z2-dep(end);
   top_sta=sta(end)+dy/S(end)
end
% just so the reader can tell the values aren't -999 anymore
top_sta
top_sta =
          25.2593016206604
toe_sta
toe_sta =
          13.0058230890364
% check for case where the toe of slope is below SWL-1.5*H0
% in this case interpolate setup from the setupAtToe(really setup as first station), and the max setup
% also un-include points seaward of SWL-1.5*H0
if Ztoe > dep(1)
```

```
   dd=SWEL_fore-dep;
   k=find(dd<0,1); % k is index of first land point
   staAtSWL=interp1(dep(k-1:k),sta(k-1:k),SWEL_fore);
   dsta=staAtSWL-sta(1);
   dsetup=maxSetup-setupAtToe;
   dsetdsta=dsetup/dsta;
   setup=setupAtToe+dsetdsta*(toe_sta-sta(1));
   sprintf('-!!- Location of SWEL-1.5*H0 is %4.1f ft landward of toe of slope',dsta)
   sprintf('-!!- Setup is interpolated between setup at toe of slope and max setup')
   sprintf('-!!-       setup is adjusted to %4.2f feet',setup)
   SWEL=SWEL-setupAtToe+setup;
   sprintf('-!!-       SWEL is adjusted to %4.2f feet',SWEL)
   k=find(dep < SWEL-1.5*H0)
   sta(k)=[];
   dep(k)=[];
else
   sprintf('-!!- The User has selected a starting point that is %4.2f feet above the elevation of SWEL-1.5H0\n',dep(1
   sprintf('-!!- This may be reasonable for some cases.  However the user may want to consider:\n')
   sprintf('-!!-   1) Selecting a starting point that is at or below %4.2f feet elevation, or\n', Ztoe)
   sprintf('-!!-   2) Reducing the incident wave height to a depth limited condition.\n')
end
ans =
-!!- Location of SWEL-1.5*H0 is  8.0 ft landward of toe of slope
ans =
-!!- Setup is interpolated between setup at toe of slope and max setup
ans =
-!!-       setup is adjusted to 0.03 feet
ans =
-!!-       SWEL is adjusted to 8.85 feet
k =
     1
     2
     3
     4
     5
% now iterate converge on a runup elevation
tol=0.01;  % convergence criteria
R2del=999;
R2_new=3*H0; %initial guess
R2=R2_new;
iter=0;
R2_all=[];
topStaAll=[];
Berm_Segs=[];
TAW_ALWAYS_VALID=1;
while(abs(R2del) > tol && iter <= 25)
    iter=iter+1;
    sprintf ('!----------------- STARTING ITERATION %d -----------------!',iter)
    % elevation of toe of slope
    Ztoe
    % station of toe slope (relative to 0-NAVD88 shoreline
    toe_sta
    % station of top of slope/extent of 2% run-up
    top_sta
    % elevation of top of slope/extent of 2% run-up
    Z2
    % incident significant wave height
    H0
    % incident spectral peak wave period
    Tp
    % incident spectral mean wave period
    T0

    R2=R2_new
    Z2=R2+SWEL
    % determine slope for this iteration
    top_sta=-999;
    for kk=1:length(sta)-1
       if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
          top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
          break;
       end
    end
    if top_sta==-999
       dy=Z2-dep(end);
       top_sta=sta(end)+dy/S(end)
    end

    % get the length of the slope (not accounting for berm)
    Lslope=top_sta-toe_sta
    % loop over profile segments to determine berm factor
    % re-calculate influence of depth of berm based on this run-up elevation
    % check for berm, berm width, berm height
    berm_width=0;
    rdh_sum=0;
    Berm_Segs=[];
    Berm_Heights=[];
    for kk=1:length(sta)-1
       ddep=dep(kk+1)-dep(kk);
       dsta=sta(kk+1)-sta(kk);
```

```
   s=ddep/dsta;
   if  (s < 1/15)     % count it as a berm if slope is flatter than 1:15 (see TAW manual)
      sprintf ('Berm Factor Calculation: Iteration %d, Profile Segment: %d',iter,kk)
      berm_width=berm_width+dsta;   % tally the width of all berm segments
      % compute the rdh for this segment and weight it by the segment length
      dh=SWEL-(dep(kk)+dep(kk+1))/2
      if  dh < 0
          chi=R2;
      else
          chi=2* H0;
      end
      if (dh <= R2 & dh >=-2*H0)
         rdh=(0.5-0.5*cos(3.14159*dh/chi)) ;
      else
         rdh=1;
      end
      rdh_sum=rdh_sum + rdh * dsta
      Berm_Segs=[Berm_Segs, kk];
      Berm_Heights=[Berm_Heights, (dep(kk)+dep(kk+1))/2];
   end
   if dep(kk) >= Z2  % jump out of loop if we reached limit of run-up for this iteration
      break
   end
end
sprintf ('!------- End Berm Factor Calculation, Iter: %d ---------!',iter)
berm_width
rB=berm_width/Lslope
if (berm_width > 0)
   rdh_mean=rdh_sum/berm_width
else
   rdh_mean=1
end
gamma_berm=1- rB * (1-rdh_mean)
if gamma_berm > 1
   gamma_berm=1
end
if gamma_berm < 0.6
   gamma_berm =0.6
end
% Iribarren number
slope=(Z2-Ztoe)/(Lslope-berm_width)
Irb=(slope/(sqrt(H0/L0)))
% runup height
gamma_berm
gamma_perm
gamma_beta
gamma_rough
gamma=gamma_berm*gamma_perm*gamma_beta*gamma_rough
% check validity
TAW_VALID=1;
if (Irb*gamma_berm < 0.5 | Irb*gamma_berm > 10 )
   sprintf('!!! - - Iribaren number: %6.2f is outside the valid range (0.5-10), TAW NOT VALID - - !!!\n', Irb*gam
   TAW_VALID=0;
else
   sprintf('!!! - - Iribaren number: %6.2f is in the valid range (0.5-10), TAW RECOMMENDED - - !!!\n', Irb*gamma_
end
islope=1/slope;
if (slope < 1/8 | slope > 1)
   sprintf('!!! - - slope: 1:%3.1f V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!\n', islope)
   TAW_VALID=0;
else
   sprintf('!!! - - slope: 1:%3.1f V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!\n', islope)
end
if TAW_VALID == 0
   TAW_ALWAYS_VALID=0;
end

if (Irb*gamma_berm < 1.8)
   R2_new=gamma*H0*1.77*Irb
else
   R2_new=gamma*H0*(4.3-(1.6/sqrt(Irb)))
end
% check to see if we need to evaluate a shallow foreshore
if berm_width > 0.25 * L0;
   disp ('!   Berm_width is greater than 1/4 wave length')
   disp ('!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on ber
   % do the foreshore calculation
   fore_H0=0.78*(SWEL_fore-min(Berm_Heights))
   % get upper slope
   fore_toe_sta=-999;
   fore_toe_dep=-999;
   for kk=length(dep)-1:-1:1
      ddep=dep(kk+1)-dep(kk);
      dsta=sta(kk+1)-sta(kk);
      s=ddep/dsta;
      if s < 1/15
         break
      end
      fore_toe_sta=sta(kk);
      fore_toe_dep=dep(kk);
```

```
         upper_slope=(Z2-fore_toe_dep)/(top_sta-fore_toe_sta)
      end
      fore_Irb=upper_slope/(sqrt(fore_H0/L0));
      fore_gamma=gamma_perm*gamma_beta*gamma_rough;
      if (fore_Irb < 1.8)
         fore_R2=fore_gamma*fore_H0*1.77*fore_Irb;
      else
         fore_R2=fore_gamma*fore_H0*(4.3-(1.6/sqrt(fore_Irb)));
      end
      if berm_width >= L0
         R2_new=fore_R2
         disp ('berm is wider than one wavelength, use full shallow foreshore solution');
      else
         w2=(berm_width-0.25*L0)/(0.75*L0)
         w1=1-w2
         R2_new=w2*fore_R2 + w1*R2_new
      end
   end % end berm width check
   % convergence criterion
   R2del=abs(R2-R2_new)
   R2_all(iter)=R2_new;
   % get the new top station (for plot purposes)
   Z2=R2_new+SWEL
   top_sta=-999;
   for kk=1:length(sta)-1
      if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
         top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
         break;
      end
   end
   if top_sta==-999
      dy=Z2-dep(end);
      top_sta=sta(end)+dy/S(end);
   end
   topStaAll(iter)=top_sta;
end
ans =
!----------------- STARTING ITERATION 1 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
Z2 =
               11.5642598
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
R2 =
                    5.499
Z2 =
          14.3449012308591
top_sta =
         -32.2309630652274
Lslope =
         -45.2367861542638
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 8
dh =
         -2.53232576914086
rdh_sum =
          0.438122339644676
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 9
dh =
         -2.58181176914087
rdh_sum =
          0.89029459984304
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 10
dh =
         -2.63129776914086
rdh_sum =
          1.35655500556964
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 11
dh =
         -2.68078376914087
rdh_sum =
          1.83693052224533
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 12
dh =
         -2.73026976914086
rdh_sum =
          2.33143683418592
```

```
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 13
dh =
         -2.77975576914087
rdh_sum =
          2.84007833206694
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 14
dh =
         -2.82924176914086
rdh_sum =
          3.36284810941447
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 15
dh =
         -2.87872776914087
rdh_sum =
          3.89972796812502
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 16
dh =
         -2.92821376914086
rdh_sum =
          4.45068843300979
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 17
dh =
         -2.97769976914087
rdh_sum =
          5.01568877535189
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 18
dh =
         -3.02718576914087
rdh_sum =
          5.59467704545765
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 19
dh =
         -3.03287976914087
rdh_sum =
          6.17527096761498
ans =
!------- End Berm Factor Calculation, Iter: 1 ---------!
berm_width =
    12
rB =
        -0.265270834207326
rdh_mean =
          0.514605913967915
gamma_berm =
          1.12876089412103
gamma_berm =
     1
slope =
         -0.14465594571547
Irb =
        -0.585940304183791
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:  -0.59 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:-6.9 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
         -1.90103058229693
!   Berm_width is greater than 1/4 wave length
!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on berm
fore_H0 =
              -1.951916616
{ Undefined function or variable 'upper_slope'.
}
TAW_iterative
plotTitle =
Iterative TAW for CM-158-1
SWEL =
                 8.8147598
SWEL_fore =
                 8.8757698
L0 =
          30.0743764412918
Ztoe =
```

```
                6.0652598
Z2 =
               11.5642598
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
top_sta =
          25.2593016206604
toe_sta =
          13.0058230890364
ans =
-!!- Location of SWEL-1.5*H0 is  8.0 ft landward of toe of slope
ans =
-!!- Setup is interpolated between setup at toe of slope and max setup
ans =
-!!-       setup is adjusted to 0.03 feet
ans =
-!!-       SWEL is adjusted to 8.85 feet
k =
     1
     2
     3
     4
     5
ans =
!----------------- STARTING ITERATION 1 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
Z2 =
               11.5642598
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
R2 =
                    5.499
Z2 =
          14.3449012308591
top_sta =
          81.4497682346348
Lslope =
          68.4439451455984
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 8
dh =
         -2.53232576914086
rdh_sum =
         0.438122339644676
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 9
dh =
         -2.58181176914087
rdh_sum =
          0.89029459984304
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 10
dh =
         -2.63129776914086
rdh_sum =
          1.35655500556964
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 11
dh =
         -2.68078376914087
rdh_sum =
          1.83693052224533
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 12
dh =
         -2.73026976914086
rdh_sum =
          2.33143683418592
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 13
dh =
         -2.77975576914087
rdh_sum =
          2.84007833206694
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 14
dh =
         -2.82924176914086
rdh_sum =
```

```
          3.36284810941447
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 15
dh =
         -2.87872776914087
rdh_sum =
          3.89972796812502
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 16
dh =
         -2.92821376914086
rdh_sum =
          4.45068843300979
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 17
dh =
         -2.97769976914087
rdh_sum =
          5.01568877535189
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 18
dh =
         -3.02718576914087
rdh_sum =
          5.59467704545765
ans =
!------- End Berm Factor Calculation, Iter: 1 ---------!
berm_width =
    11
rB =
         0.160715458125625
rdh_mean =
         0.508607004132514
gamma_berm =
         0.921025549549434
slope =
         0.144134275768724
Irb =
         0.583827238967095
gamma_berm =
         0.921025549549434
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
         0.921025549549434
ans =
!!! - - Iribaren number:   0.54 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:6.9 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
           1.7445835080355
!   Berm_width is greater than 1/4 wave length
!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on berm
fore_H0 =
              -1.951916616
{ Undefined function or variable 'upper_slope'.
Error in <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative', 'C:\FEMA-TransectAn
nt-weight:bold">TAW_iterative</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpsw
       fore_Irb=upper_slope/(sqrt(fore_H0/L0));
}
opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswell\Additional_Transects\4_taw\TAW_iterative.m',706,
plot(dep)
TAW_iterative
plotTitle =
Iterative TAW for CM-158-1
SWEL =
                 8.8147598
SWEL_fore =
                 8.8757698
L0 =
         30.0743764412918
Ztoe =
                 6.0652598
Z2 =
                11.5642598
toe_sta =
         13.0058230890364
top_sta =
         21.5607006903156
top_sta =
         21.5607006903156
toe_sta =
         13.0058230890364
ans =
-!!- Location of SWEL-1.5*H0 is  8.0 ft landward of toe of slope
ans =
```

```
-!!- Setup is interpolated between setup at toe of slope and max setup
ans =
-!!-       setup is adjusted to 0.03 feet
ans =
-!!-       SWEL is adjusted to 8.85 feet
k =
     1
     2
     3
     4
     5
ans =
!----------------- STARTING ITERATION 1 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          21.5607006903156
Z2 =
               11.5642598
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
R2 =
                    5.499
Z2 =
          14.3449012308591
top_sta =
          28.9576958377802
Lslope =
          15.9518727487438
ans =
!------- End Berm Factor Calculation, Iter: 1 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
         0.519038833952029
Irb =
          2.10240768704559
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:   2.10 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.9 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          5.85923490600709
R2del =
          0.36023490600709
Z2 =
          14.7051361368662
ans =
!----------------- STARTING ITERATION 2 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          29.9159840306086
Z2 =
          14.7051361368662
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
R2 =
          5.85923490600709
Z2 =
          14.7051361368662
top_sta =
```

```
          29.9159840306086
Lslope =
          16.9101609415722
ans =
!------- End Berm Factor Calculation, Iter: 2 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
          0.51092809623271
Irb =
          2.06955450494576
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:   2.07 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:2.0 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          5.84324370166387
R2del =
        0.0159912043432158
Z2 =
           14.689144932523
ans =
!----------------- STARTING ITERATION 3 -----------------!
Ztoe =
                  6.0652598
toe_sta =
          13.0058230890364
top_sta =
          29.8734446152003
Z2 =
           14.689144932523
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          5.84324370166387
Z2 =
           14.689144932523
top_sta =
          29.8734446152003
Lslope =
          16.8676215261639
ans =
!------- End Berm Factor Calculation, Iter: 3 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
          0.511268593449659
Irb =
          2.07093371574754
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:   2.07 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:2.0 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
```

```
           5.8439226719916
R2del =
        0.0006789703277299
Z2 =
          14.6898239028507
Z2 =
          14.6898239028507
diary on         % begin recording
% FEMA appeal for The Town of Harpswell, Cumberland county, Maine
% TRANSECT ID: CM-158-1
% calculation by SJH, Ransom Consulting, Inc. 26-Feb-2020
% 100-year wave runup using TAW methodology
% including berm and weighted average with foreshore if necessary
%
% chk nld 20200220
%
% This script assumes that the incident wave conditions provided
% as input in the configuration section below are the
% appropriate values located at the end of the foreshore
% or toe of the slope on which the run-up is being calculated
% the script does not attempt to apply a depth limit or any other
% transformation to the incident wave conditions other than
% conversion of the peak wave period to the spectral mean wave
% as recommended in the references below
%
% references:
%
% Van der Meer, J.W., 2002. Technical Report Wave Run-up and
% Wave Overtopping at Dikes. TAW Technical Advisory Committee on
% Flood Defence, The Netherlands.
%
% FEMA. 2007,  Atlantic Ocean and Gulf of Mexico Coastal Guidelines Update
%
%
%
%------------------------------------------------------------------------
% CONFIG
%------------------------------------------------------------------------
fname='inpfiles/CM-158-1sta_ele_include.csv';   % file with station, elevation, include
                                    % third colunm is 0 for excluded points
imgname='logfiles/CM-158-1-runup';
SWEL=8.816;  % 100-yr still water level including wave setup.
H0=1.833;    % significant wave height at toe of structure
Tp=2.6668;    % peak period, 1/fma,
T0=Tp/1.1;
gamma_berm=1;   % this may get changed automatically below
gamma_rough=1;
gamma_beta=1;
gamma_perm=1;
setupAtToe=-0.0012402;
maxSetup=0.06101;   % only used in case of berm/shallow foreshore weighted average
plotTitle='Iterative TAW for CM-158-1'
plotTitle =
Iterative TAW for CM-158-1
% END CONFIG
%------------------------------
SWEL=SWEL+setupAtToe
SWEL =
                8.8147598
SWEL_fore=SWEL+maxSetup
SWEL_fore =
                8.8757698
% FIND WAVELENGTH USING DEEPWATER DISPERSION RELATION
% using English units
L0=32.15/(2*pi)*T0^2
L0 =
          30.0743764412918
% Find Hb (Munk, 1949)
%Hb=H0/(3.3*(H0/L0)^(1/3))
%Db=-Hb/.78+SWEL;  % depth at breaking
% The toe elevation here is only used to determine the average
% structure slope, it is not used to depth limit the wave height.
% Any depth limiting or other modification of the wave height
% to make it consitent with TAW guidance should be performed
% prior to the input of the significant wave height given above.
Ztoe=SWEL-1.5*H0
Ztoe =
                6.0652598
% read the transect
[sta,dep,inc] = textread(fname,'%n%n%n%*[^\n]','delimiter',',','headerlines',0);
% remove unselected points
k=find(inc==0);
sta(k)=[];
dep(k)=[];
sta_org=sta;  % used for plotting purposes
dep_org=dep;
% initial guess at maximum run-up elevation to estimate slope
Z2=SWEL+1.5*H0
Z2 =
               11.5642598
```

```
% determine station at the max runup and -1.5*H0 (i.e. the toe)
top_sta=-999;
toe_sta=-999;
for kk=1:length(sta)-1
    if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
       top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
    end
    if  ((Ztoe > dep(kk)) & (Ztoe <= dep(kk+1)))    % here is the intersection of Ztoe with profile
       toe_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Ztoe)
    end
end
toe_sta =
          13.0058230890364
% check to make sure we got them, if not extend the end slopes outward
S=diff(dep)./diff(sta);
if toe_sta==-999
   dy=dep(1)-Ztoe;
   toe_sta=sta(1)-dy/S(1)
end
if top_sta==-999
   dy=Z2-dep(end);
   top_sta=sta(end)+dy/S(end)
end
top_sta =
          25.2593016206604
% just so the reader can tell the values aren't -999 anymore
top_sta
top_sta =
          25.2593016206604
toe_sta
toe_sta =
          13.0058230890364
% check for case where the toe of slope is below SWL-1.5*H0
% in this case interpolate setup from the setupAtToe(really setup as first station), and the max setup
% also un-include points seaward of SWL-1.5*H0
if Ztoe > dep(1)
   dd=SWEL_fore-dep;
   k=find(dd<0,1); % k is index of first land point
   staAtSWL=interp1(dep(k-1:k),sta(k-1:k),SWEL_fore);
   dsta=staAtSWL-sta(1);
   dsetup=maxSetup-setupAtToe;
   dsetdsta=dsetup/dsta;
   setup=setupAtToe+dsetdsta*(toe_sta-sta(1));
   sprintf('-!!- Location of SWEL-1.5*H0 is %4.1f ft landward of toe of slope',dsta)
   sprintf('-!!- Setup is interpolated between setup at toe of slope and max setup')
   sprintf('-!!-       setup is adjusted to %4.2f feet',setup)
   SWEL=SWEL-setupAtToe+setup;
   sprintf('-!!-       SWEL is adjusted to %4.2f feet',SWEL)
   k=find(dep < SWEL-1.5*H0)
   sta(k)=[];
   dep(k)=[];
else
   sprintf('-!!- The User has selected a starting point that is %4.2f feet above the elevation of SWEL-1.5H0\n',dep(1
   sprintf('-!!- This may be reasonable for some cases.  However the user may want to consider:\n')
   sprintf('-!!-   1) Selecting a starting point that is at or below %4.2f feet elevation, or\n', Ztoe)
   sprintf('-!!-   2) Reducing the incident wave height to a depth limited condition.\n')
end
ans =
-!!- Location of SWEL-1.5*H0 is 27.0 ft landward of toe of slope
ans =
-!!- Setup is interpolated between setup at toe of slope and max setup
ans =
-!!-       setup is adjusted to 0.05 feet
ans =
-!!-       SWEL is adjusted to 8.87 feet
k =
     1
     2
     3
     4
     5
     6
     7
     8
     9
    10
    11
    12
    13
    14
    15
% now iterate converge on a runup elevation
tol=0.01;  % convergence criteria
R2del=999;
R2_new=3*H0; %initial guess
R2=R2_new;
iter=0;
R2_all=[];
topStaAll=[];
Berm_Segs=[];
```

```
TAW_ALWAYS_VALID=1;
while(abs(R2del) > tol && iter <= 25)
    iter=iter+1;
    sprintf ('!----------------- STARTING ITERATION %d -----------------!',iter)
    % elevation of toe of slope
    Ztoe
    % station of toe slope (relative to 0-NAVD88 shoreline
    toe_sta
    % station of top of slope/extent of 2% run-up
    top_sta
    % elevation of top of slope/extent of 2% run-up
    Z2
    % incident significant wave height
    H0
    % incident spectral peak wave period
    Tp
    % incident spectral mean wave period
    T0

    R2=R2_new
    Z2=R2+SWEL
    % determine slope for this iteration
    top_sta=-999;
    for kk=1:length(sta)-1
       if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
          top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
          break;
       end
    end
    if top_sta==-999
       dy=Z2-dep(end);
       top_sta=sta(end)+dy/S(end)
    end

    % get the length of the slope (not accounting for berm)
    Lslope=top_sta-toe_sta
    % loop over profile segments to determine berm factor
    % re-calculate influence of depth of berm based on this run-up elevation
    % check for berm, berm width, berm height
    berm_width=0;
    rdh_sum=0;
    Berm_Segs=[];
    Berm_Heights=[];
    for kk=1:length(sta)-1
       ddep=dep(kk+1)-dep(kk);
       dsta=sta(kk+1)-sta(kk);
       s=ddep/dsta;
       if  (s < 1/15)     % count it as a berm if slope is flatter than 1:15 (see TAW manual)
          sprintf ('Berm Factor Calculation: Iteration %d, Profile Segment: %d',iter,kk)
          berm_width=berm_width+dsta;   % tally the width of all berm segments
          % compute the rdh for this segment and weight it by the segment length
          dh=SWEL-(dep(kk)+dep(kk+1))/2
          if  dh < 0
              chi=R2;
          else
              chi=2* H0;
          end
          if (dh <= R2 & dh >=-2*H0)
             rdh=(0.5-0.5*cos(3.14159*dh/chi)) ;
          else
             rdh=1;
          end
          rdh_sum=rdh_sum + rdh * dsta
          Berm_Segs=[Berm_Segs, kk];
          Berm_Heights=[Berm_Heights, (dep(kk)+dep(kk+1))/2];
       end
       if dep(kk) >= Z2  % jump out of loop if we reached limit of run-up for this iteration
          break
       end
    end
    sprintf ('!------- End Berm Factor Calculation, Iter: %d ---------!',iter)
    berm_width
    rB=berm_width/Lslope
    if (berm_width > 0)
       rdh_mean=rdh_sum/berm_width
    else
       rdh_mean=1
    end
    gamma_berm=1- rB * (1-rdh_mean)
    if gamma_berm > 1
       gamma_berm=1
    end
    if gamma_berm < 0.6
       gamma_berm =0.6
    end
    % Iribarren number
    slope=(Z2-Ztoe)/(Lslope-berm_width)
    Irb=(slope/(sqrt(H0/L0)))
    % runup height
    gamma_berm
```

```
    gamma_perm
    gamma_beta
    gamma_rough
    gamma=gamma_berm*gamma_perm*gamma_beta*gamma_rough
    % check validity
    TAW_VALID=1;
    if (Irb*gamma_berm < 0.5 | Irb*gamma_berm > 10 )
       sprintf('!!! - - Iribaren number: %6.2f is outside the valid range (0.5-10), TAW NOT VALID - - !!!\n', Irb*gam
       TAW_VALID=0;
    else
       sprintf('!!! - - Iribaren number: %6.2f is in the valid range (0.5-10), TAW RECOMMENDED - - !!!\n', Irb*gamma_
    end
    islope=1/slope;
    if (slope < 1/8 | slope > 1)
       sprintf('!!! - - slope: 1:%3.1f V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!\n', islope)
       TAW_VALID=0;
    else
       sprintf('!!! - - slope: 1:%3.1f V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!\n', islope)
    end
    if TAW_VALID == 0
       TAW_ALWAYS_VALID=0;
    end

    if (Irb*gamma_berm < 1.8)
       R2_new=gamma*H0*1.77*Irb
    else
       R2_new=gamma*H0*(4.3-(1.6/sqrt(Irb)))
    end
    % check to see if we need to evaluate a shallow foreshore
    if berm_width > 0.25 * L0;
       disp ('!   Berm_width is greater than 1/4 wave length')
       disp ('!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on ber
       % do the foreshore calculation
       fore_H0=0.78*(SWEL_fore-min(Berm_Heights))
       % get upper slope
       fore_toe_sta=-999;
       fore_toe_dep=-999;
       for kk=length(dep)-1:-1:1
          ddep=dep(kk+1)-dep(kk);
          dsta=sta(kk+1)-sta(kk);
          s=ddep/dsta;
          if s < 1/15
             break
          end
          fore_toe_sta=sta(kk);
          fore_toe_dep=dep(kk);
          upper_slope=(Z2-fore_toe_dep)/(top_sta-fore_toe_sta)
       end
       fore_Irb=upper_slope/(sqrt(fore_H0/L0));
       fore_gamma=gamma_perm*gamma_beta*gamma_rough;
       if (fore_Irb < 1.8)
          fore_R2=fore_gamma*fore_H0*1.77*fore_Irb;
       else
          fore_R2=fore_gamma*fore_H0*(4.3-(1.6/sqrt(fore_Irb)));
       end
       if berm_width >= L0
          R2_new=fore_R2
          disp ('berm is wider than one wavelength, use full shallow foreshore solution');
       else
          w2=(berm_width-0.25*L0)/(0.75*L0)
          w1=1-w2
          R2_new=w2*fore_R2 + w1*R2_new
       end
    end % end berm width check
    % convergence criterion
    R2del=abs(R2-R2_new)
    R2_all(iter)=R2_new;
    % get the new top station (for plot purposes)
    Z2=R2_new+SWEL
    top_sta=-999;
    for kk=1:length(sta)-1
       if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
          top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
          break;
       end
    end
    if top_sta==-999
       dy=Z2-dep(end);
       top_sta=sta(end)+dy/S(end);
    end
    topStaAll(iter)=top_sta;
end
ans =
!----------------- STARTING ITERATION 1 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
```

```
Z2 =
               11.5642598
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
R2 =
                    5.499
Z2 =
          14.3667865524122
top_sta =
           81.892021024375
Lslope =
          68.8861979353386
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
          0.431923830491403
ans =
!------- End Berm Factor Calculation, Iter: 1 ---------!
berm_width =
     1
rB =
        0.0145166960867643
rdh_mean =
          0.431923830491403
gamma_berm =
          0.991753410893111
slope =
          0.122285928581822
Irb =
          0.495328787463503
gamma_berm =
          0.991753410893111
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
          0.991753410893111
ans =
!!! - - Iribaren number:   0.49 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:8.2 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
          1.59379699302061
R2del =
          3.90520300697939
Z2 =
          10.4615835454328
top_sta =
          19.2653369442464
ans =
!----------------- STARTING ITERATION 2 -----------------!
Ztoe =
                 6.0652598
toe_sta =
          13.0058230890364
top_sta =
          19.2653369442464
Z2 =
          10.4615835454328
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
R2 =
          1.59379699302061
Z2 =
          10.4615835454328
top_sta =
          19.2653369442464
Lslope =
          6.25951385520996
ans =
!------- End Berm Factor Calculation, Iter: 2 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
```

```
gamma_berm =
     1
slope =
          0.702342681416647
Irb =
           2.84489436196414
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          6.14310102574172
R2del =
          4.54930403272112
Z2 =
          15.0108875781539
ans =
!----------------- STARTING ITERATION 3 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          94.9078442014701
Z2 =
          15.0108875781539
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          6.14310102574172
Z2 =
          15.0108875781539
top_sta =
          94.9078442014701
Lslope =
          81.9020211124337
ans =
Berm Factor Calculation: Iteration 3, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
          0.358484070289488
ans =
!------- End Berm Factor Calculation, Iter: 3 ---------!
berm_width =
     1
rB =
          0.012209710901117
rdh_mean =
          0.358484070289488
gamma_berm =
          0.992167275959773
slope =
          0.110573600698081
Irb =
          0.447887080667724
gamma_berm =
          0.992167275959773
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
          0.992167275959773
ans =
!!! - - Iribaren number:   0.44 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:9.0 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
           1.4417473624043
R2del =
          4.70135366333742
Z2 =
          10.3095339148165
```

```
top_sta =
         19.0488478062948
ans =
!----------------- STARTING ITERATION 4 -----------------!
Ztoe =
                6.0652598
toe_sta =
         13.0058230890364
top_sta =
         19.0488478062948
Z2 =
         10.3095339148165
H0 =
                   1.833
Tp =
                  2.6668
T0 =
         2.42436363636364
R2 =
          1.4417473624043
Z2 =
         10.3095339148165
top_sta =
         19.0488478062948
Lslope =
         6.04302471725842
ans =
!------- End Berm Factor Calculation, Iter: 4 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
        0.702342670003516
Irb =
         2.84489431573435
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          6.1431010116139
R2del =
          4.7013536492096
Z2 =
         15.0108875640261
ans =
!----------------- STARTING ITERATION 5 -----------------!
Ztoe =
                6.0652598
toe_sta =
         13.0058230890364
top_sta =
         94.9078439159788
Z2 =
         15.0108875640261
H0 =
                   1.833
Tp =
                  2.6668
T0 =
         2.42436363636364
R2 =
          6.1431010116139
Z2 =
         15.0108875640261
top_sta =
         94.9078439159788
Lslope =
         81.9020208269424
ans =
Berm Factor Calculation: Iteration 5, Profile Segment: 8
dh =
        -2.51044044758778
rdh_sum =
         0.358484071705405
```

```
ans =
!------- End Berm Factor Calculation, Iter: 5 ---------!
berm_width =
     1
rB =
        0.0122097109436772
rdh_mean =
         0.358484071705405
gamma_berm =
         0.992167275949758
slope =
          0.11057360091365
Irb =
         0.447887081540904
gamma_berm =
         0.992167275949758
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
         0.992167275949758
ans =
!!! - - Iribaren number:   0.44 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:9.0 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
          1.44174736520051
R2del =
          4.70135364641339
Z2 =
          10.3095339176127
top_sta =
          19.0488478102761
ans =
!----------------- STARTING ITERATION 6 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          19.0488478102761
Z2 =
          10.3095339176127
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          1.44174736520051
Z2 =
          10.3095339176127
top_sta =
          19.0488478102761
Lslope =
          6.04302472123968
ans =
!------- End Berm Factor Calculation, Iter: 6 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
         0.702342670003516
Irb =
          2.84489431573435
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
           6.1431010116139
```

```
R2del =
          4.70135364641339
Z2 =
          15.0108875640261
ans =
!----------------- STARTING ITERATION 7 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          94.9078439159788
Z2 =
          15.0108875640261
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
           6.1431010116139
Z2 =
          15.0108875640261
top_sta =
          94.9078439159788
Lslope =
          81.9020208269424
ans =
Berm Factor Calculation: Iteration 7, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
          0.358484071705405
ans =
!------- End Berm Factor Calculation, Iter: 7 ---------!
berm_width =
     1
rB =
        0.0122097109436772
rdh_mean =
          0.358484071705405
gamma_berm =
          0.992167275949758
slope =
           0.11057360091365
Irb =
          0.447887081540904
gamma_berm =
          0.992167275949758
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
          0.992167275949758
ans =
!!! - - Iribaren number:   0.44 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:9.0 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
           1.44174736520051
R2del =
          4.70135364641339
Z2 =
          10.3095339176127
top_sta =
          19.0488478102761
ans =
!----------------- STARTING ITERATION 8 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          19.0488478102761
Z2 =
          10.3095339176127
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          1.44174736520051
Z2 =
          10.3095339176127
```

```
top_sta =
         19.0488478102761
Lslope =
          6.04302472123968
ans =
!------- End Berm Factor Calculation, Iter: 8 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
         0.702342670003516
Irb =
          2.84489431573435
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          6.1431010116139
R2del =
          4.70135364641339
Z2 =
          15.0108875640261
ans =
!----------------- STARTING ITERATION 9 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          94.9078439159788
Z2 =
          15.0108875640261
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          6.1431010116139
Z2 =
          15.0108875640261
top_sta =
          94.9078439159788
Lslope =
          81.9020208269424
ans =
Berm Factor Calculation: Iteration 9, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
         0.358484071705405
ans =
!------- End Berm Factor Calculation, Iter: 9 ---------!
berm_width =
     1
rB =
        0.0122097109436772
rdh_mean =
          0.358484071705405
gamma_berm =
          0.992167275949758
slope =
          0.11057360091365
Irb =
          0.447887081540904
gamma_berm =
          0.992167275949758
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
```

```
gamma =
         0.992167275949758
ans =
!!! - - Iribaren number:   0.44 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:9.0 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
          1.44174736520051
R2del =
          4.70135364641339
Z2 =
          10.3095339176127
top_sta =
          19.0488478102761
ans =
!----------------- STARTING ITERATION 10 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          19.0488478102761
Z2 =
          10.3095339176127
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          1.44174736520051
Z2 =
          10.3095339176127
top_sta =
          19.0488478102761
Lslope =
          6.04302472123968
ans =
!------- End Berm Factor Calculation, Iter: 10 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
          0.702342670003516
Irb =
          2.84489431573435
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
           6.1431010116139
R2del =
          4.70135364641339
Z2 =
          15.0108875640261
ans =
!----------------- STARTING ITERATION 11 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          94.9078439159788
Z2 =
          15.0108875640261
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
           6.1431010116139
```

```
Z2 =
         15.0108875640261
top_sta =
         94.9078439159788
Lslope =
         81.9020208269424
ans =
Berm Factor Calculation: Iteration 11, Profile Segment: 8
dh =
        -2.51044044758778
rdh_sum =
         0.358484071705405
ans =
!------- End Berm Factor Calculation, Iter: 11 ---------!
berm_width =
     1
rB =
        0.0122097109436772
rdh_mean =
         0.358484071705405
gamma_berm =
         0.992167275949758
slope =
         0.11057360091365
Irb =
         0.447887081540904
gamma_berm =
         0.992167275949758
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
         0.992167275949758
ans =
!!! - - Iribaren number:   0.44 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:9.0 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
         1.44174736520051
R2del =
         4.70135364641339
Z2 =
         10.3095339176127
top_sta =
         19.0488478102761
ans =
!----------------- STARTING ITERATION 12 -----------------!
Ztoe =
                6.0652598
toe_sta =
         13.0058230890364
top_sta =
         19.0488478102761
Z2 =
         10.3095339176127
H0 =
                   1.833
Tp =
                  2.6668
T0 =
         2.42436363636364
R2 =
         1.44174736520051
Z2 =
         10.3095339176127
top_sta =
         19.0488478102761
Lslope =
         6.04302472123968
ans =
!------- End Berm Factor Calculation, Iter: 12 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
         0.702342670003516
Irb =
         2.84489431573435
gamma_berm =
     1
gamma_perm =
     1
```

```
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
           6.1431010116139
R2del =
          4.70135364641339
Z2 =
          15.0108875640261
ans =
!----------------- STARTING ITERATION 13 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          94.9078439159788
Z2 =
          15.0108875640261
H0 =
                   1.833
Tp =
                  2.6668
T0 =
          2.42436363636364
R2 =
           6.1431010116139
Z2 =
          15.0108875640261
top_sta =
          94.9078439159788
Lslope =
          81.9020208269424
ans =
Berm Factor Calculation: Iteration 13, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
          0.358484071705405
ans =
!------- End Berm Factor Calculation, Iter: 13 ---------!
berm_width =
     1
rB =
        0.0122097109436772
rdh_mean =
          0.358484071705405
gamma_berm =
          0.992167275949758
slope =
           0.11057360091365
Irb =
          0.447887081540904
gamma_berm =
          0.992167275949758
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
          0.992167275949758
ans =
!!! - - Iribaren number:   0.44 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:9.0 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
          1.44174736520051
R2del =
          4.70135364641339
Z2 =
          10.3095339176127
top_sta =
          19.0488478102761
ans =
!----------------- STARTING ITERATION 14 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          19.0488478102761
```

```
Z2 =
          10.3095339176127
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          1.44174736520051
Z2 =
          10.3095339176127
top_sta =
          19.0488478102761
Lslope =
          6.04302472123968
ans =
!------- End Berm Factor Calculation, Iter: 14 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
         0.702342670003516
Irb =
          2.84489431573435
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
           6.1431010116139
R2del =
          4.70135364641339
Z2 =
          15.0108875640261
ans =
!------------------ STARTING ITERATION 15 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          94.9078439159788
Z2 =
          15.0108875640261
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
           6.1431010116139
Z2 =
          15.0108875640261
top_sta =
          94.9078439159788
Lslope =
          81.9020208269424
ans =
Berm Factor Calculation: Iteration 15, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
         0.358484071705405
ans =
!------- End Berm Factor Calculation, Iter: 15 ---------!
berm_width =
     1
rB =
        0.0122097109436772
rdh_mean =
          0.358484071705405
gamma_berm =
          0.992167275949758
```

```
slope =
          0.11057360091365
Irb =
          0.447887081540904
gamma_berm =
          0.992167275949758
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
          0.992167275949758
ans =
!!! - - Iribaren number:   0.44 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:9.0 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
          1.44174736520051
R2del =
          4.70135364641339
Z2 =
          10.3095339176127
top_sta =
          19.0488478102761
ans =
!----------------- STARTING ITERATION 16 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          19.0488478102761
Z2 =
          10.3095339176127
H0 =
                   1.833
Tp =
                  2.6668
T0 =
          2.42436363636364
R2 =
          1.44174736520051
Z2 =
          10.3095339176127
top_sta =
          19.0488478102761
Lslope =
          6.04302472123968
ans =
!------- End Berm Factor Calculation, Iter: 16 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
          0.702342670003516
Irb =
          2.84489431573435
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
           6.1431010116139
R2del =
          4.70135364641339
Z2 =
          15.0108875640261
ans =
!----------------- STARTING ITERATION 17 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
```

```
top_sta =
          94.9078439159788
Z2 =
          15.0108875640261
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
           6.1431010116139
Z2 =
          15.0108875640261
top_sta =
          94.9078439159788
Lslope =
          81.9020208269424
ans =
Berm Factor Calculation: Iteration 17, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
          0.358484071705405
ans =
!------- End Berm Factor Calculation, Iter: 17 ---------!
berm_width =
     1
rB =
        0.0122097109436772
rdh_mean =
          0.358484071705405
gamma_berm =
          0.992167275949758
slope =
           0.11057360091365
Irb =
          0.447887081540904
gamma_berm =
          0.992167275949758
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
          0.992167275949758
ans =
!!! - - Iribaren number:   0.44 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:9.0 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
          1.44174736520051
R2del =
          4.70135364641339
Z2 =
          10.3095339176127
top_sta =
          19.0488478102761
ans =
!----------------- STARTING ITERATION 18 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          19.0488478102761
Z2 =
          10.3095339176127
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          1.44174736520051
Z2 =
          10.3095339176127
top_sta =
          19.0488478102761
Lslope =
          6.04302472123968
ans =
!------- End Berm Factor Calculation, Iter: 18 ---------!
berm_width =
     0
rB =
     0
```

```
rdh_mean =
     1
gamma_berm =
     1
slope =
         0.702342670003516
Irb =
          2.84489431573435
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          6.1431010116139
R2del =
          4.70135364641339
Z2 =
          15.0108875640261
ans =
!----------------- STARTING ITERATION 19 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          94.9078439159788
Z2 =
          15.0108875640261
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          6.1431010116139
Z2 =
          15.0108875640261
top_sta =
          94.9078439159788
Lslope =
          81.9020208269424
ans =
Berm Factor Calculation: Iteration 19, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
          0.358484071705405
ans =
!------- End Berm Factor Calculation, Iter: 19 ---------!
berm_width =
     1
rB =
        0.0122097109436772
rdh_mean =
          0.358484071705405
gamma_berm =
          0.992167275949758
slope =
          0.11057360091365
Irb =
          0.447887081540904
gamma_berm =
          0.992167275949758
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
          0.992167275949758
ans =
!!! - - Iribaren number:   0.44 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:9.0 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
          1.44174736520051
R2del =
          4.70135364641339
```

```
Z2 =
         10.3095339176127
top_sta =
         19.0488478102761
ans =
!----------------- STARTING ITERATION 20 -----------------!
Ztoe =
                6.0652598
toe_sta =
         13.0058230890364
top_sta =
         19.0488478102761
Z2 =
         10.3095339176127
H0 =
                   1.833
Tp =
                  2.6668
T0 =
         2.42436363636364
R2 =
         1.44174736520051
Z2 =
         10.3095339176127
top_sta =
         19.0488478102761
Lslope =
         6.04302472123968
ans =
!------- End Berm Factor Calculation, Iter: 20 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
        0.702342670003516
Irb =
         2.84489431573435
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          6.1431010116139
R2del =
         4.70135364641339
Z2 =
         15.0108875640261
ans =
!----------------- STARTING ITERATION 21 -----------------!
Ztoe =
                6.0652598
toe_sta =
         13.0058230890364
top_sta =
         94.9078439159788
Z2 =
         15.0108875640261
H0 =
                   1.833
Tp =
                  2.6668
T0 =
         2.42436363636364
R2 =
          6.1431010116139
Z2 =
         15.0108875640261
top_sta =
         94.9078439159788
Lslope =
         81.9020208269424
ans =
Berm Factor Calculation: Iteration 21, Profile Segment: 8
dh =
        -2.51044044758778
```

```
rdh_sum =
         0.358484071705405
ans =
!------- End Berm Factor Calculation, Iter: 21 ---------!
berm_width =
     1
rB =
        0.0122097109436772
rdh_mean =
         0.358484071705405
gamma_berm =
         0.992167275949758
slope =
          0.11057360091365
Irb =
         0.447887081540904
gamma_berm =
         0.992167275949758
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
         0.992167275949758
ans =
!!! - - Iribaren number:   0.44 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:9.0 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
          1.44174736520051
R2del =
          4.70135364641339
Z2 =
          10.3095339176127
top_sta =
          19.0488478102761
ans =
!----------------- STARTING ITERATION 22 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          19.0488478102761
Z2 =
          10.3095339176127
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          1.44174736520051
Z2 =
          10.3095339176127
top_sta =
          19.0488478102761
Lslope =
          6.04302472123968
ans =
!------- End Berm Factor Calculation, Iter: 22 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
         0.702342670003516
Irb =
          2.84489431573435
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
```

```
R2_new =
          6.1431010116139
R2del =
          4.70135364641339
Z2 =
          15.0108875640261
ans =
!----------------- STARTING ITERATION 23 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          94.9078439159788
Z2 =
          15.0108875640261
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
R2 =
          6.1431010116139
Z2 =
          15.0108875640261
top_sta =
          94.9078439159788
Lslope =
          81.9020208269424
ans =
Berm Factor Calculation: Iteration 23, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
          0.358484071705405
ans =
!------- End Berm Factor Calculation, Iter: 23 ---------!
berm_width =
     1
rB =
        0.0122097109436772
rdh_mean =
          0.358484071705405
gamma_berm =
          0.992167275949758
slope =
          0.11057360091365
Irb =
          0.447887081540904
gamma_berm =
          0.992167275949758
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
          0.992167275949758
ans =
!!! - - Iribaren number:   0.44 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:9.0 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
          1.44174736520051
R2del =
          4.70135364641339
Z2 =
          10.3095339176127
top_sta =
          19.0488478102761
ans =
!----------------- STARTING ITERATION 24 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          19.0488478102761
Z2 =
          10.3095339176127
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
R2 =
          1.44174736520051
```

```
Z2 =
         10.3095339176127
top_sta =
         19.0488478102761
Lslope =
         6.04302472123968
ans =
!------- End Berm Factor Calculation, Iter: 24 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
        0.702342670003516
Irb =
         2.84489431573435
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          6.1431010116139
R2del =
         4.70135364641339
Z2 =
         15.0108875640261
ans =
!----------------- STARTING ITERATION 25 -----------------!
Ztoe =
                6.0652598
toe_sta =
         13.0058230890364
top_sta =
         94.9078439159788
Z2 =
         15.0108875640261
H0 =
                    1.833
Tp =
                   2.6668
T0 =
         2.42436363636364
R2 =
          6.1431010116139
Z2 =
         15.0108875640261
top_sta =
         94.9078439159788
Lslope =
         81.9020208269424
ans =
Berm Factor Calculation: Iteration 25, Profile Segment: 8
dh =
        -2.51044044758778
rdh_sum =
         0.358484071705405
ans =
!------- End Berm Factor Calculation, Iter: 25 ---------!
berm_width =
     1
rB =
       0.0122097109436772
rdh_mean =
         0.358484071705405
gamma_berm =
         0.992167275949758
slope =
          0.11057360091365
Irb =
         0.447887081540904
gamma_berm =
         0.992167275949758
gamma_perm =
     1
gamma_beta =
     1
```

```
gamma_rough =
     1
gamma =
         0.992167275949758
ans =
!!! - - Iribaren number:   0.44 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:9.0 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
          1.44174736520051
R2del =
          4.70135364641339
Z2 =
          10.3095339176127
top_sta =
          19.0488478102761
ans =
!----------------- STARTING ITERATION 26 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          19.0488478102761
Z2 =
          10.3095339176127
H0 =
                   1.833
Tp =
                  2.6668
T0 =
          2.42436363636364
R2 =
          1.44174736520051
Z2 =
          10.3095339176127
top_sta =
          19.0488478102761
Lslope =
          6.04302472123968
ans =
!------- End Berm Factor Calculation, Iter: 26 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
         0.702342670003516
Irb =
          2.84489431573435
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
           6.1431010116139
R2del =
          4.70135364641339
Z2 =
          15.0108875640261
% final 2% runup elevation
Z2=R2_new+SWEL
Z2 =
          15.0108875640261
diary off
diary on         % begin recording
% FEMA appeal for The Town of Harpswell, Cumberland county, Maine
% TRANSECT ID: CM-158-1
% calculation by SJH, Ransom Consulting, Inc. 26-Feb-2020
% 100-year wave runup using TAW methodology
% including berm and weighted average with foreshore if necessary
%
% chk nld 20200220
%
% This script assumes that the incident wave conditions provided
% as input in the configuration section below are the
```

```
% appropriate values located at the end of the foreshore
% or toe of the slope on which the run-up is being calculated
% the script does not attempt to apply a depth limit or any other
% transformation to the incident wave conditions other than
% conversion of the peak wave period to the spectral mean wave
% as recommended in the references below
%
% references:
%
% Van der Meer, J.W., 2002. Technical Report Wave Run-up and
% Wave Overtopping at Dikes. TAW Technical Advisory Committee on
% Flood Defence, The Netherlands.
%
% FEMA. 2007,  Atlantic Ocean and Gulf of Mexico Coastal Guidelines Update
%
%
%
%--------------------------------------------------------------------------
% CONFIG
%--------------------------------------------------------------------------
fname='inpfiles/CM-158-1sta_ele_include.csv';   % file with station, elevation, include
                                        % third colunm is 0 for excluded points
imgname='logfiles/CM-158-1-runup';
SWEL=8.816;  % 100-yr still water level including wave setup.
H0=1.833;    % significant wave height at toe of structure
Tp=2.6668;    % peak period, 1/fma,
T0=Tp/1.1;
gamma_berm=1;   % this may get changed automatically below
gamma_rough=1;
gamma_beta=1;
gamma_perm=1;
setupAtToe=-0.0012402;
maxSetup=0.06101;   % only used in case of berm/shallow foreshore weighted average
plotTitle='Iterative TAW for CM-158-1'
plotTitle =
Iterative TAW for CM-158-1
% END CONFIG
%------------------------------
SWEL=SWEL+setupAtToe
SWEL =
                8.8147598
SWEL_fore=SWEL+maxSetup
SWEL_fore =
                8.8757698
% FIND WAVELENGTH USING DEEPWATER DISPERSION RELATION
% using English units
L0=32.15/(2*pi)*T0^2
L0 =
          30.0743764412918
% Find Hb (Munk, 1949)
%Hb=H0/(3.3*(H0/L0)^(1/3))
%Db=-Hb/.78+SWEL;  % depth at breaking
% The toe elevation here is only used to determine the average
% structure slope, it is not used to depth limit the wave height.
% Any depth limiting or other modification of the wave height
% to make it consitent with TAW guidance should be performed
% prior to the input of the significant wave height given above.
Ztoe=SWEL-1.5*H0
Ztoe =
                6.0652598
% read the transect
[sta,dep,inc] = textread(fname,'%n%n%n%*[^\n]','delimiter',',','headerlines',0);
% remove unselected points
k=find(inc==0);
sta(k)=[];
dep(k)=[];
sta_org=sta;  % used for plotting purposes
dep_org=dep;
% initial guess at maximum run-up elevation to estimate slope
Z2=SWEL+1.5*H0
Z2 =
               11.5642598
% determine station at the max runup and -1.5*H0 (i.e. the toe)
top_sta=-999;
toe_sta=-999;
for kk=1:length(sta)-1
    if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
       top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
    end
    if  ((Ztoe > dep(kk)) & (Ztoe <= dep(kk+1)))    % here is the intersection of Ztoe with profile
       toe_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Ztoe)
    end
end
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
% check to make sure we got them, if not extend the end slopes outward
S=diff(dep)./diff(sta);
if toe_sta==-999
```

```
   dy=dep(1)-Ztoe;
   toe_sta=sta(1)-dy/S(1)
end
if top_sta==-999
   dy=Z2-dep(end);
   top_sta=sta(end)+dy/S(end)
end
% just so the reader can tell the values aren't -999 anymore
top_sta
top_sta =
         25.2593016206604
toe_sta
toe_sta =
         13.0058230890364
% check for case where the toe of slope is below SWL-1.5*H0
% in this case interpolate setup from the setupAtToe(really setup as first station), and the max setup
% also un-include points seaward of SWL-1.5*H0
if Ztoe > dep(1)
   dd=SWEL_fore-dep;
   k=find(dd<0,1); % k is index of first land point
   staAtSWL=interp1(dep(k-1:k),sta(k-1:k),SWEL_fore);
   dsta=staAtSWL-sta(1);
   dsetup=maxSetup-setupAtToe;
   dsetdsta=dsetup/dsta;
   setup=setupAtToe+dsetdsta*(toe_sta-sta(1));
   sprintf('-!!- Location of SWEL-1.5*H0 is %4.1f ft landward of toe of slope',dsta)
   sprintf('-!!- Setup is interpolated between setup at toe of slope and max setup')
   sprintf('-!!-       setup is adjusted to %4.2f feet',setup)
   SWEL=SWEL-setupAtToe+setup;
   sprintf('-!!-       SWEL is adjusted to %4.2f feet',SWEL)
   k=find(dep < SWEL-1.5*H0)
   sta(k)=[];
   dep(k)=[];
else
   sprintf('-!!- The User has selected a starting point that is %4.2f feet above the elevation of SWEL-1.5H0\n',dep(1
   sprintf('-!!- This may be reasonable for some cases.  However the user may want to consider:\n')
   sprintf('-!!-   1) Selecting a starting point that is at or below %4.2f feet elevation, or\n', Ztoe)
   sprintf('-!!-   2) Reducing the incident wave height to a depth limited condition.\n')
end
ans =
-!!- Location of SWEL-1.5*H0 is 27.0 ft landward of toe of slope
ans =
-!!- Setup is interpolated between setup at toe of slope and max setup
ans =
-!!-       setup is adjusted to 0.05 feet
ans =
-!!-       SWEL is adjusted to 8.87 feet
k =
     1
     2
     3
     4
     5
     6
     7
     8
     9
    10
    11
    12
    13
    14
    15
% now iterate converge on a runup elevation
tol=0.01;  % convergence criteria
R2del=999;
R2_new=3*H0; %initial guess
R2=R2_new;
iter=0;
R2_all=[];
topStaAll=[];
Berm_Segs=[];
TAW_ALWAYS_VALID=1;
while(abs(R2del) > tol && iter <= 25)
    iter=iter+1;
    sprintf ('!----------------- STARTING ITERATION %d -----------------!',iter)
    % elevation of toe of slope
    Ztoe
    % station of toe slope (relative to 0-NAVD88 shoreline
    toe_sta
    % station of top of slope/extent of 2% run-up
    top_sta
    % elevation of top of slope/extent of 2% run-up
    Z2
    % incident significant wave height
    H0
    % incident spectral peak wave period
    Tp
    % incident spectral mean wave period
    T0
```

```
R2=R2_new
Z2=R2+SWEL
% determine slope for this iteration
top_sta=-999;
for kk=1:length(sta)-1
   if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
      top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
      break;
   end
end
if top_sta==-999
   dy=Z2-dep(end);
   top_sta=sta(end)+dy/S(end)
end

% get the length of the slope (not accounting for berm)
Lslope=top_sta-toe_sta
% loop over profile segments to determine berm factor
% re-calculate influence of depth of berm based on this run-up elevation
% check for berm, berm width, berm height
berm_width=0;
rdh_sum=0;
Berm_Segs=[];
Berm_Heights=[];
for kk=1:length(sta)-1
   ddep=dep(kk+1)-dep(kk);
   dsta=sta(kk+1)-sta(kk);
   s=ddep/dsta;
   if  (s < 1/15)     % count it as a berm if slope is flatter than 1:15 (see TAW manual)
      sprintf ('Berm Factor Calculation: Iteration %d, Profile Segment: %d',iter,kk)
      berm_width=berm_width+dsta;   % tally the width of all berm segments
      % compute the rdh for this segment and weight it by the segment length
      dh=SWEL-(dep(kk)+dep(kk+1))/2
      if  dh < 0
          chi=R2;
      else
          chi=2* H0;
      end
      if (dh <= R2 & dh >=-2*H0)
         rdh=(0.5-0.5*cos(3.14159*dh/chi)) ;
      else
         rdh=1;
      end
      rdh_sum=rdh_sum + rdh * dsta
      Berm_Segs=[Berm_Segs, kk];
      Berm_Heights=[Berm_Heights, (dep(kk)+dep(kk+1))/2];
   end
   if dep(kk) >= Z2  % jump out of loop if we reached limit of run-up for this iteration
      break
   end
end
sprintf ('!------- End Berm Factor Calculation, Iter: %d ---------!',iter)
berm_width
rB=berm_width/Lslope
if (berm_width > 0)
   rdh_mean=rdh_sum/berm_width
else
   rdh_mean=1
end
gamma_berm=1- rB * (1-rdh_mean)
if gamma_berm > 1
   gamma_berm=1
end
if gamma_berm < 0.6
   gamma_berm =0.6
end
% Iribarren number
slope=(Z2-Ztoe)/(Lslope-berm_width)
Irb=(slope/(sqrt(H0/L0)))
% runup height
gamma_berm
gamma_perm
gamma_beta
gamma_rough
gamma=gamma_berm*gamma_perm*gamma_beta*gamma_rough
% check validity
TAW_VALID=1;
if (Irb*gamma_berm < 0.5 | Irb*gamma_berm > 10 )
   sprintf('!!! - - Iribaren number: %6.2f is outside the valid range (0.5-10), TAW NOT VALID - - !!!\n', Irb*gam
   TAW_VALID=0;
else
   sprintf('!!! - - Iribaren number: %6.2f is in the valid range (0.5-10), TAW RECOMMENDED - - !!!\n', Irb*gamma_
end
islope=1/slope;
if (slope < 1/8 | slope > 1)
   sprintf('!!! - - slope: 1:%3.1f V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!\n', islope)
   TAW_VALID=0;
else
   sprintf('!!! - - slope: 1:%3.1f V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!\n', islope)
```

```
    end
    if TAW_VALID == 0
       TAW_ALWAYS_VALID=0;
    end

    if (Irb*gamma_berm < 1.8)
       R2_new=gamma*H0*1.77*Irb
    else
       R2_new=gamma*H0*(4.3-(1.6/sqrt(Irb)))
    end
    % check to see if we need to evaluate a shallow foreshore
    if berm_width > 0.25 * L0;
       disp ('!   Berm_width is greater than 1/4 wave length')
       disp ('!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on ber
       % do the foreshore calculation
       fore_H0=0.78*(SWEL_fore-min(Berm_Heights))
       % get upper slope
       fore_toe_sta=-999;
       fore_toe_dep=-999;
       for kk=length(dep)-1:-1:1
          ddep=dep(kk+1)-dep(kk);
          dsta=sta(kk+1)-sta(kk);
          s=ddep/dsta;
          if s < 1/15
             break
          end
          fore_toe_sta=sta(kk);
          fore_toe_dep=dep(kk);
          upper_slope=(Z2-fore_toe_dep)/(top_sta-fore_toe_sta)
       end
       fore_Irb=upper_slope/(sqrt(fore_H0/L0));
       fore_gamma=gamma_perm*gamma_beta*gamma_rough;
       if (fore_Irb < 1.8)
          fore_R2=fore_gamma*fore_H0*1.77*fore_Irb;
       else
          fore_R2=fore_gamma*fore_H0*(4.3-(1.6/sqrt(fore_Irb)));
       end
       if berm_width >= L0
          R2_new=fore_R2
          disp ('berm is wider than one wavelength, use full shallow foreshore solution');
       else
          w2=(berm_width-0.25*L0)/(0.75*L0)
          w1=1-w2
          R2_new=w2*fore_R2 + w1*R2_new
       end
    end % end berm width check
    % convergence criterion
    R2del=abs(R2-R2_new)
    R2_all(iter)=R2_new;
    % get the new top station (for plot purposes)
    Z2=R2_new+SWEL
    top_sta=-999;
    for kk=1:length(sta)-1
       if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
          top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
          break;
       end
    end
    if top_sta==-999
       dy=Z2-dep(end);
       top_sta=sta(end)+dy/S(end);
    end
    topStaAll(iter)=top_sta;
end
ans =
!----------------- STARTING ITERATION 1 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
Z2 =
               11.5642598
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
R2 =
                    5.499
Z2 =
          14.3667865524122
top_sta =
           81.892021024375
Lslope =
          68.8861979353386
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 8
```

```
dh =
         -2.51044044758778
rdh_sum =
         0.431923830491403
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
         0.877877093142679
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 10
dh =
         -2.60941244758778
rdh_sum =
          1.33790298328708
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 11
dh =
         -2.65889844758778
rdh_sum =
          1.81203344910653
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 12
dh =
         -2.70838444758778
rdh_sum =
          2.30028916609791
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 13
dh =
         -2.75787044758778
rdh_sum =
          2.80267952054878
ans =
!------- End Berm Factor Calculation, Iter: 1 ---------!
berm_width =
     6
rB =
        0.0871001765205858
rdh_mean =
         0.467113253424797
gamma_berm =
         0.953585470307819
slope =
         0.132008724091542
Irb =
         0.534711736642019
gamma_berm =
         0.953585470307819
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
         0.953585470307819
ans =
!!! - - Iribaren number:   0.51 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:7.6 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          1.65430306052428
R2del =
          3.84469693947572
Z2 =
          10.5220896129365
top_sta =
          19.3514858309067
ans =
!----------------- STARTING ITERATION 2 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          19.3514858309067
Z2 =
          10.5220896129365
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          1.65430306052428
Z2 =
          10.5220896129365
```

```
top_sta =
          19.3514858309067
Lslope =
          6.34566274187029
ans =
!------- End Berm Factor Calculation, Iter: 2 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
         0.702342685741744
Irb =
          2.84489437948328
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          6.14310103109557
R2del =
           4.4887979705713
Z2 =
          15.0108875835078
ans =
!----------------- STARTING ITERATION 3 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          94.9078443096592
Z2 =
          15.0108875835078
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          6.14310103109557
Z2 =
          15.0108875835078
top_sta =
          94.9078443096592
Lslope =
          81.9020212206228
ans =
Berm Factor Calculation: Iteration 3, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
         0.358484069752914
ans =
Berm Factor Calculation: Iteration 3, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
         0.729148362505709
ans =
Berm Factor Calculation: Iteration 3, Profile Segment: 10
dh =
         -2.60941244758778
rdh_sum =
          1.1120757074954
ans =
Berm Factor Calculation: Iteration 3, Profile Segment: 11
dh =
         -2.65889844758778
rdh_sum =
          1.50734108044906
ans =
Berm Factor Calculation: Iteration 3, Profile Segment: 12
dh =
         -2.70838444758778
```

```
rdh_sum =
         1.91501155556789
ans =
Berm Factor Calculation: Iteration 3, Profile Segment: 13
dh =
        -2.75787044758778
rdh_sum =
         2.33514626257144
ans =
!------- End Berm Factor Calculation, Iter: 3 ---------!
berm_width =
     6
rB =
        0.0732582653099312
rdh_mean =
         0.389191043761906
gamma_berm =
         0.955253195430228
slope =
          0.1178575700574
Irb =
         0.47739137239217
gamma_berm =
         0.955253195430228
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
         0.955253195430228
ans =
!!! - - Iribaren number:   0.46 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:8.5 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
         1.47954710467866
R2del =
         4.66355392641691
Z2 =
         10.3473336570909
top_sta =
         19.1026672965928
ans =
!----------------- STARTING ITERATION 4 -----------------!
Ztoe =
                6.0652598
toe_sta =
         13.0058230890364
top_sta =
         19.1026672965928
Z2 =
         10.3473336570909
H0 =
                   1.833
Tp =
                  2.6668
T0 =
         2.42436363636364
R2 =
         1.47954710467866
Z2 =
         10.3473336570909
top_sta =
         19.1026672965928
Lslope =
          6.0968442075564
ans =
!------- End Berm Factor Calculation, Iter: 4 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
         0.702342672916538
Irb =
         2.84489432753378
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
```

```
gamma =
     1
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          6.14310101521981
R2del =
          4.66355391054114
Z2 =
           15.010887567632
ans =
!----------------- STARTING ITERATION 5 -----------------!
Ztoe =
                 6.0652598
toe_sta =
          13.0058230890364
top_sta =
           94.907843988846
Z2 =
           15.010887567632
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          6.14310101521981
Z2 =
           15.010887567632
top_sta =
           94.907843988846
Lslope =
          81.9020208998096
ans =
Berm Factor Calculation: Iteration 5, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
          0.358484071344014
ans =
Berm Factor Calculation: Iteration 5, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
          0.729148365730869
ans =
Berm Factor Calculation: Iteration 5, Profile Segment: 10
dh =
         -2.60941244758778
rdh_sum =
           1.11207571239696
ans =
Berm Factor Calculation: Iteration 5, Profile Segment: 11
dh =
         -2.65889844758778
rdh_sum =
           1.50734108706868
ans =
Berm Factor Calculation: Iteration 5, Profile Segment: 12
dh =
         -2.70838444758778
rdh_sum =
           1.91501156394646
ans =
Berm Factor Calculation: Iteration 5, Profile Segment: 13
dh =
         -2.75787044758778
rdh_sum =
           2.33514627274905
ans =
!------- End Berm Factor Calculation, Iter: 5 ---------!
berm_width =
     6
rB =
        0.0732582655968865
rdh_mean =
          0.389191045458175
gamma_berm =
          0.955253195379218
slope =
          0.117857570346384
Irb =
          0.477391373562724
gamma_berm =
          0.955253195379218
gamma_perm =
     1
```

```
gamma_beta =
     1
gamma_rough =
     1
gamma =
         0.955253195379218
ans =
!!! - - Iribaren number:   0.46 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:8.5 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
          1.47954710822748
R2del =
          4.66355390699233
Z2 =
          10.3473336606397
top_sta =
          19.1026673016456
ans =
!----------------- STARTING ITERATION 6 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          19.1026673016456
Z2 =
          10.3473336606397
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          1.47954710822748
Z2 =
          10.3473336606397
top_sta =
          19.1026673016456
Lslope =
          6.09684421260923
ans =
!------- End Berm Factor Calculation, Iter: 6 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
         0.702342672916538
Irb =
          2.84489432753377
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          6.14310101521981
R2del =
          4.66355390699233
Z2 =
           15.010887567632
ans =
!----------------- STARTING ITERATION 7 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          94.9078439888459
Z2 =
           15.010887567632
H0 =
                     1.833
Tp =
                    2.6668
```

```
T0 =
          2.42436363636364
R2 =
          6.14310101521981
Z2 =
           15.010887567632
top_sta =
          94.9078439888459
Lslope =
          81.9020208998095
ans =
Berm Factor Calculation: Iteration 7, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
          0.358484071344014
ans =
Berm Factor Calculation: Iteration 7, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
          0.729148365730869
ans =
Berm Factor Calculation: Iteration 7, Profile Segment: 10
dh =
         -2.60941244758778
rdh_sum =
           1.11207571239696
ans =
Berm Factor Calculation: Iteration 7, Profile Segment: 11
dh =
         -2.65889844758778
rdh_sum =
           1.50734108706868
ans =
Berm Factor Calculation: Iteration 7, Profile Segment: 12
dh =
         -2.70838444758778
rdh_sum =
           1.91501156394646
ans =
Berm Factor Calculation: Iteration 7, Profile Segment: 13
dh =
         -2.75787044758778
rdh_sum =
           2.33514627274905
ans =
!------- End Berm Factor Calculation, Iter: 7 ---------!
berm_width =
     6
rB =
        0.0732582655968866
rdh_mean =
          0.389191045458175
gamma_berm =
          0.955253195379218
slope =
          0.117857570346384
Irb =
          0.477391373562724
gamma_berm =
          0.955253195379218
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
          0.955253195379218
ans =
!!! - - Iribaren number:   0.46 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:8.5 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
           1.47954710822748
R2del =
           4.66355390699233
Z2 =
           10.3473336606397
top_sta =
           19.1026673016456
ans =
!----------------- STARTING ITERATION 8 -----------------!
Ztoe =
                 6.0652598
toe_sta =
           13.0058230890364
top_sta =
           19.1026673016456
```

```
Z2 =
         10.3473336606397
H0 =
                    1.833
Tp =
                   2.6668
T0 =
         2.42436363636364
R2 =
         1.47954710822748
Z2 =
         10.3473336606397
top_sta =
         19.1026673016456
Lslope =
         6.09684421260923
ans =
!------- End Berm Factor Calculation, Iter: 8 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
        0.702342672916538
Irb =
         2.84489432753377
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
         6.14310101521981
R2del =
         4.66355390699233
Z2 =
          15.010887567632
ans =
!----------------- STARTING ITERATION 9 -----------------!
Ztoe =
                6.0652598
toe_sta =
         13.0058230890364
top_sta =
         94.9078439888459
Z2 =
          15.010887567632
H0 =
                    1.833
Tp =
                   2.6668
T0 =
         2.42436363636364
R2 =
         6.14310101521981
Z2 =
          15.010887567632
top_sta =
         94.9078439888459
Lslope =
         81.9020208998095
ans =
Berm Factor Calculation: Iteration 9, Profile Segment: 8
dh =
        -2.51044044758778
rdh_sum =
        0.358484071344014
ans =
Berm Factor Calculation: Iteration 9, Profile Segment: 9
dh =
        -2.55992644758778
rdh_sum =
        0.729148365730869
ans =
Berm Factor Calculation: Iteration 9, Profile Segment: 10
dh =
        -2.60941244758778
```

```
rdh_sum =
          1.11207571239696
ans =
Berm Factor Calculation: Iteration 9, Profile Segment: 11
dh =
         -2.65889844758778
rdh_sum =
          1.50734108706868
ans =
Berm Factor Calculation: Iteration 9, Profile Segment: 12
dh =
         -2.70838444758778
rdh_sum =
          1.91501156394646
ans =
Berm Factor Calculation: Iteration 9, Profile Segment: 13
dh =
         -2.75787044758778
rdh_sum =
          2.33514627274905
ans =
!------- End Berm Factor Calculation, Iter: 9 ---------!
berm_width =
     6
rB =
        0.0732582655968866
rdh_mean =
          0.389191045458175
gamma_berm =
          0.955253195379218
slope =
          0.117857570346384
Irb =
          0.477391373562724
gamma_berm =
          0.955253195379218
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
          0.955253195379218
ans =
!!! - - Iribaren number:   0.46 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:8.5 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
          1.47954710822748
R2del =
          4.66355390699233
Z2 =
          10.3473336606397
top_sta =
          19.1026673016456
ans =
!----------------- STARTING ITERATION 10 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          19.1026673016456
Z2 =
          10.3473336606397
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
R2 =
          1.47954710822748
Z2 =
          10.3473336606397
top_sta =
          19.1026673016456
Lslope =
          6.09684421260923
ans =
!------- End Berm Factor Calculation, Iter: 10 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
```

```
slope =
        0.702342672916538
Irb =
         2.84489432753377
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
         6.14310101521981
R2del =
         4.66355390699233
Z2 =
          15.010887567632
ans =
!----------------- STARTING ITERATION 11 -----------------!
Ztoe =
                6.0652598
toe_sta =
         13.0058230890364
top_sta =
         94.9078439888459
Z2 =
          15.010887567632
H0 =
                   1.833
Tp =
                  2.6668
T0 =
         2.42436363636364
R2 =
         6.14310101521981
Z2 =
          15.010887567632
top_sta =
         94.9078439888459
Lslope =
         81.9020208998095
ans =
Berm Factor Calculation: Iteration 11, Profile Segment: 8
dh =
        -2.51044044758778
rdh_sum =
        0.358484071344014
ans =
Berm Factor Calculation: Iteration 11, Profile Segment: 9
dh =
        -2.55992644758778
rdh_sum =
        0.729148365730869
ans =
Berm Factor Calculation: Iteration 11, Profile Segment: 10
dh =
        -2.60941244758778
rdh_sum =
         1.11207571239696
ans =
Berm Factor Calculation: Iteration 11, Profile Segment: 11
dh =
        -2.65889844758778
rdh_sum =
         1.50734108706868
ans =
Berm Factor Calculation: Iteration 11, Profile Segment: 12
dh =
        -2.70838444758778
rdh_sum =
         1.91501156394646
ans =
Berm Factor Calculation: Iteration 11, Profile Segment: 13
dh =
        -2.75787044758778
rdh_sum =
         2.33514627274905
ans =
!------- End Berm Factor Calculation, Iter: 11 ---------!
berm_width =
     6
rB =
        0.0732582655968866
```

```
rdh_mean =
         0.389191045458175
gamma_berm =
         0.955253195379218
slope =
         0.117857570346384
Irb =
         0.477391373562724
gamma_berm =
         0.955253195379218
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
         0.955253195379218
ans =
!!! - - Iribaren number:   0.46 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:8.5 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
          1.47954710822748
R2del =
          4.66355390699233
Z2 =
          10.3473336606397
top_sta =
          19.1026673016456
ans =
!----------------- STARTING ITERATION 12 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          19.1026673016456
Z2 =
          10.3473336606397
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          1.47954710822748
Z2 =
          10.3473336606397
top_sta =
          19.1026673016456
Lslope =
          6.09684421260923
ans =
!------- End Berm Factor Calculation, Iter: 12 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
         0.702342672916538
Irb =
          2.84489432753377
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          6.14310101521981
R2del =
          4.66355390699233
Z2 =
           15.010887567632
ans =
!----------------- STARTING ITERATION 13 -----------------!
```

```
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          94.9078439888459
Z2 =
           15.010887567632
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          6.14310101521981
Z2 =
           15.010887567632
top_sta =
          94.9078439888459
Lslope =
          81.9020208998095
ans =
Berm Factor Calculation: Iteration 13, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
          0.358484071344014
ans =
Berm Factor Calculation: Iteration 13, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
          0.729148365730869
ans =
Berm Factor Calculation: Iteration 13, Profile Segment: 10
dh =
         -2.60941244758778
rdh_sum =
           1.11207571239696
ans =
Berm Factor Calculation: Iteration 13, Profile Segment: 11
dh =
         -2.65889844758778
rdh_sum =
           1.50734108706868
ans =
Berm Factor Calculation: Iteration 13, Profile Segment: 12
dh =
         -2.70838444758778
rdh_sum =
           1.91501156394646
ans =
Berm Factor Calculation: Iteration 13, Profile Segment: 13
dh =
         -2.75787044758778
rdh_sum =
           2.33514627274905
ans =
!------- End Berm Factor Calculation, Iter: 13 ---------!
berm_width =
     6
rB =
        0.0732582655968866
rdh_mean =
          0.389191045458175
gamma_berm =
          0.955253195379218
slope =
          0.117857570346384
Irb =
          0.477391373562724
gamma_berm =
          0.955253195379218
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
          0.955253195379218
ans =
!!! - - Iribaren number:   0.46 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:8.5 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
           1.47954710822748
R2del =
           4.66355390699233
```

```
Z2 =
         10.3473336606397
top_sta =
         19.1026673016456
ans =
!----------------- STARTING ITERATION 14 -----------------!
Ztoe =
                6.0652598
toe_sta =
         13.0058230890364
top_sta =
         19.1026673016456
Z2 =
         10.3473336606397
H0 =
                   1.833
Tp =
                  2.6668
T0 =
         2.42436363636364
R2 =
         1.47954710822748
Z2 =
         10.3473336606397
top_sta =
         19.1026673016456
Lslope =
         6.09684421260923
ans =
!------- End Berm Factor Calculation, Iter: 14 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
        0.702342672916538
Irb =
         2.84489432753377
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
         6.14310101521981
R2del =
         4.66355390699233
Z2 =
          15.010887567632
ans =
!----------------- STARTING ITERATION 15 -----------------!
Ztoe =
                6.0652598
toe_sta =
         13.0058230890364
top_sta =
         94.9078439888459
Z2 =
          15.010887567632
H0 =
                   1.833
Tp =
                  2.6668
T0 =
         2.42436363636364
R2 =
         6.14310101521981
Z2 =
          15.010887567632
top_sta =
         94.9078439888459
Lslope =
         81.9020208998095
ans =
Berm Factor Calculation: Iteration 15, Profile Segment: 8
dh =
        -2.51044044758778
```

```
rdh_sum =
         0.358484071344014
ans =
Berm Factor Calculation: Iteration 15, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
         0.729148365730869
ans =
Berm Factor Calculation: Iteration 15, Profile Segment: 10
dh =
         -2.60941244758778
rdh_sum =
          1.11207571239696
ans =
Berm Factor Calculation: Iteration 15, Profile Segment: 11
dh =
         -2.65889844758778
rdh_sum =
          1.50734108706868
ans =
Berm Factor Calculation: Iteration 15, Profile Segment: 12
dh =
         -2.70838444758778
rdh_sum =
          1.91501156394646
ans =
Berm Factor Calculation: Iteration 15, Profile Segment: 13
dh =
         -2.75787044758778
rdh_sum =
          2.33514627274905
ans =
!------- End Berm Factor Calculation, Iter: 15 ---------!
berm_width =
     6
rB =
        0.0732582655968866
rdh_mean =
         0.389191045458175
gamma_berm =
         0.955253195379218
slope =
         0.117857570346384
Irb =
         0.477391373562724
gamma_berm =
         0.955253195379218
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
         0.955253195379218
ans =
!!! - - Iribaren number:   0.46 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:8.5 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
          1.47954710822748
R2del =
          4.66355390699233
Z2 =
          10.3473336606397
top_sta =
          19.1026673016456
ans =
!----------------- STARTING ITERATION 16 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          19.1026673016456
Z2 =
          10.3473336606397
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
R2 =
          1.47954710822748
Z2 =
          10.3473336606397
top_sta =
          19.1026673016456
```

```
Lslope =
          6.09684421260923
ans =
!------- End Berm Factor Calculation, Iter: 16 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
         0.702342672916538
Irb =
          2.84489432753377
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          6.14310101521981
R2del =
          4.66355390699233
Z2 =
           15.010887567632
ans =
!----------------- STARTING ITERATION 17 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          94.9078439888459
Z2 =
           15.010887567632
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          6.14310101521981
Z2 =
           15.010887567632
top_sta =
          94.9078439888459
Lslope =
          81.9020208998095
ans =
Berm Factor Calculation: Iteration 17, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
         0.358484071344014
ans =
Berm Factor Calculation: Iteration 17, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
         0.729148365730869
ans =
Berm Factor Calculation: Iteration 17, Profile Segment: 10
dh =
         -2.60941244758778
rdh_sum =
          1.11207571239696
ans =
Berm Factor Calculation: Iteration 17, Profile Segment: 11
dh =
         -2.65889844758778
rdh_sum =
          1.50734108706868
ans =
Berm Factor Calculation: Iteration 17, Profile Segment: 12
dh =
         -2.70838444758778
rdh_sum =
          1.91501156394646
```

```
ans =
Berm Factor Calculation: Iteration 17, Profile Segment: 13
dh =
         -2.75787044758778
rdh_sum =
          2.33514627274905
ans =
!------- End Berm Factor Calculation, Iter: 17 ---------!
berm_width =
     6
rB =
        0.0732582655968866
rdh_mean =
          0.389191045458175
gamma_berm =
          0.955253195379218
slope =
          0.117857570346384
Irb =
          0.477391373562724
gamma_berm =
          0.955253195379218
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
          0.955253195379218
ans =
!!! - - Iribaren number:   0.46 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:8.5 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
          1.47954710822748
R2del =
          4.66355390699233
Z2 =
          10.3473336606397
top_sta =
          19.1026673016456
ans =
!----------------- STARTING ITERATION 18 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          19.1026673016456
Z2 =
          10.3473336606397
H0 =
                      1.833
Tp =
                     2.6668
T0 =
          2.42436363636364
R2 =
          1.47954710822748
Z2 =
          10.3473336606397
top_sta =
          19.1026673016456
Lslope =
          6.09684421260923
ans =
!------- End Berm Factor Calculation, Iter: 18 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
          0.702342672916538
Irb =
          2.84489432753377
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
```

```
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          6.14310101521981
R2del =
          4.66355390699233
Z2 =
           15.010887567632
ans =
!----------------- STARTING ITERATION 19 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          94.9078439888459
Z2 =
           15.010887567632
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          6.14310101521981
Z2 =
           15.010887567632
top_sta =
          94.9078439888459
Lslope =
          81.9020208998095
ans =
Berm Factor Calculation: Iteration 19, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
          0.358484071344014
ans =
Berm Factor Calculation: Iteration 19, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
          0.729148365730869
ans =
Berm Factor Calculation: Iteration 19, Profile Segment: 10
dh =
         -2.60941244758778
rdh_sum =
           1.11207571239696
ans =
Berm Factor Calculation: Iteration 19, Profile Segment: 11
dh =
         -2.65889844758778
rdh_sum =
           1.50734108706868
ans =
Berm Factor Calculation: Iteration 19, Profile Segment: 12
dh =
         -2.70838444758778
rdh_sum =
           1.91501156394646
ans =
Berm Factor Calculation: Iteration 19, Profile Segment: 13
dh =
         -2.75787044758778
rdh_sum =
           2.33514627274905
ans =
!------- End Berm Factor Calculation, Iter: 19 ---------!
berm_width =
     6
rB =
        0.0732582655968866
rdh_mean =
          0.389191045458175
gamma_berm =
          0.955253195379218
slope =
          0.117857570346384
Irb =
          0.477391373562724
gamma_berm =
          0.955253195379218
gamma_perm =
     1
gamma_beta =
     1
```

```
gamma_rough =
     1
gamma =
          0.955253195379218
ans =
!!! - - Iribaren number:   0.46 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:8.5 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
          1.47954710822748
R2del =
          4.66355390699233
Z2 =
          10.3473336606397
top_sta =
          19.1026673016456
ans =
!----------------- STARTING ITERATION 20 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          19.1026673016456
Z2 =
          10.3473336606397
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          1.47954710822748
Z2 =
          10.3473336606397
top_sta =
          19.1026673016456
Lslope =
          6.09684421260923
ans =
!------- End Berm Factor Calculation, Iter: 20 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
          0.702342672916538
Irb =
          2.84489432753377
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          6.14310101521981
R2del =
          4.66355390699233
Z2 =
           15.010887567632
ans =
!----------------- STARTING ITERATION 21 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          94.9078439888459
Z2 =
           15.010887567632
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
```

```
R2 =
         6.14310101521981
Z2 =
          15.010887567632
top_sta =
         94.9078439888459
Lslope =
         81.9020208998095
ans =
Berm Factor Calculation: Iteration 21, Profile Segment: 8
dh =
        -2.51044044758778
rdh_sum =
         0.358484071344014
ans =
Berm Factor Calculation: Iteration 21, Profile Segment: 9
dh =
        -2.55992644758778
rdh_sum =
         0.729148365730869
ans =
Berm Factor Calculation: Iteration 21, Profile Segment: 10
dh =
        -2.60941244758778
rdh_sum =
          1.11207571239696
ans =
Berm Factor Calculation: Iteration 21, Profile Segment: 11
dh =
        -2.65889844758778
rdh_sum =
          1.50734108706868
ans =
Berm Factor Calculation: Iteration 21, Profile Segment: 12
dh =
        -2.70838444758778
rdh_sum =
          1.91501156394646
ans =
Berm Factor Calculation: Iteration 21, Profile Segment: 13
dh =
        -2.75787044758778
rdh_sum =
          2.33514627274905
ans =
!------- End Berm Factor Calculation, Iter: 21 ---------!
berm_width =
     6
rB =
        0.0732582655968866
rdh_mean =
         0.389191045458175
gamma_berm =
         0.955253195379218
slope =
         0.117857570346384
Irb =
         0.477391373562724
gamma_berm =
         0.955253195379218
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
         0.955253195379218
ans =
!!! - - Iribaren number:   0.46 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:8.5 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
          1.47954710822748
R2del =
          4.66355390699233
Z2 =
          10.3473336606397
top_sta =
          19.1026673016456
ans =
!----------------- STARTING ITERATION 22 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          19.1026673016456
Z2 =
          10.3473336606397
```

```
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
R2 =
          1.47954710822748
Z2 =
          10.3473336606397
top_sta =
          19.1026673016456
Lslope =
          6.09684421260923
ans =
!------- End Berm Factor Calculation, Iter: 22 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
         0.702342672916538
Irb =
          2.84489432753377
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          6.14310101521981
R2del =
          4.66355390699233
Z2 =
           15.010887567632
ans =
!----------------- STARTING ITERATION 23 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          94.9078439888459
Z2 =
           15.010887567632
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
R2 =
          6.14310101521981
Z2 =
           15.010887567632
top_sta =
          94.9078439888459
Lslope =
          81.9020208998095
ans =
Berm Factor Calculation: Iteration 23, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
         0.358484071344014
ans =
Berm Factor Calculation: Iteration 23, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
         0.729148365730869
ans =
Berm Factor Calculation: Iteration 23, Profile Segment: 10
dh =
         -2.60941244758778
rdh_sum =
          1.11207571239696
```

```
ans =
Berm Factor Calculation: Iteration 23, Profile Segment: 11
dh =
         -2.65889844758778
rdh_sum =
          1.50734108706868
ans =
Berm Factor Calculation: Iteration 23, Profile Segment: 12
dh =
         -2.70838444758778
rdh_sum =
          1.91501156394646
ans =
Berm Factor Calculation: Iteration 23, Profile Segment: 13
dh =
         -2.75787044758778
rdh_sum =
          2.33514627274905
ans =
!------- End Berm Factor Calculation, Iter: 23 ---------!
berm_width =
     6
rB =
        0.0732582655968866
rdh_mean =
         0.389191045458175
gamma_berm =
         0.955253195379218
slope =
         0.117857570346384
Irb =
         0.477391373562724
gamma_berm =
         0.955253195379218
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
         0.955253195379218
ans =
!!! - - Iribaren number:   0.46 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:8.5 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
          1.47954710822748
R2del =
          4.66355390699233
Z2 =
          10.3473336606397
top_sta =
          19.1026673016456
ans =
!----------------- STARTING ITERATION 24 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          19.1026673016456
Z2 =
          10.3473336606397
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
R2 =
          1.47954710822748
Z2 =
          10.3473336606397
top_sta =
          19.1026673016456
Lslope =
          6.09684421260923
ans =
!------- End Berm Factor Calculation, Iter: 24 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
         0.702342672916538
```

```
Irb =
          2.84489432753377
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          6.14310101521981
R2del =
          4.66355390699233
Z2 =
           15.010887567632
ans =
!----------------- STARTING ITERATION 25 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          94.9078439888459
Z2 =
           15.010887567632
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          6.14310101521981
Z2 =
           15.010887567632
top_sta =
          94.9078439888459
Lslope =
          81.9020208998095
ans =
Berm Factor Calculation: Iteration 25, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
          0.358484071344014
ans =
Berm Factor Calculation: Iteration 25, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
          0.729148365730869
ans =
Berm Factor Calculation: Iteration 25, Profile Segment: 10
dh =
         -2.60941244758778
rdh_sum =
           1.11207571239696
ans =
Berm Factor Calculation: Iteration 25, Profile Segment: 11
dh =
         -2.65889844758778
rdh_sum =
           1.50734108706868
ans =
Berm Factor Calculation: Iteration 25, Profile Segment: 12
dh =
         -2.70838444758778
rdh_sum =
           1.91501156394646
ans =
Berm Factor Calculation: Iteration 25, Profile Segment: 13
dh =
         -2.75787044758778
rdh_sum =
           2.33514627274905
ans =
!------- End Berm Factor Calculation, Iter: 25 ---------!
berm_width =
     6
rB =
        0.0732582655968866
rdh_mean =
          0.389191045458175
```

```
gamma_berm =
         0.955253195379218
slope =
         0.117857570346384
Irb =
         0.477391373562724
gamma_berm =
         0.955253195379218
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
         0.955253195379218
ans =
!!! - - Iribaren number:   0.46 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:8.5 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
          1.47954710822748
R2del =
          4.66355390699233
Z2 =
          10.3473336606397
top_sta =
          19.1026673016456
ans =
!----------------- STARTING ITERATION 26 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          19.1026673016456
Z2 =
          10.3473336606397
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          1.47954710822748
Z2 =
          10.3473336606397
top_sta =
          19.1026673016456
Lslope =
          6.09684421260923
ans =
!------- End Berm Factor Calculation, Iter: 26 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
         0.702342672916538
Irb =
          2.84489432753377
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
     1
gamma =
     1
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          6.14310101521981
R2del =
          4.66355390699233
Z2 =
           15.010887567632
% final 2% runup elevation
Z2=R2_new+SWEL
Z2 =
           15.010887567632
```

```
diary off
diary on         % begin recording
% FEMA appeal for The Town of Harpswell, Cumberland county, Maine
% TRANSECT ID: CM-158-1
% calculation by SJH, Ransom Consulting, Inc. 26-Feb-2020
% 100-year wave runup using TAW methodology
% including berm and weighted average with foreshore if necessary
%
% chk nld 20200220
%
% This script assumes that the incident wave conditions provided
% as input in the configuration section below are the
% appropriate values located at the end of the foreshore
% or toe of the slope on which the run-up is being calculated
% the script does not attempt to apply a depth limit or any other
% transformation to the incident wave conditions other than
% conversion of the peak wave period to the spectral mean wave
% as recommended in the references below
%
% references:
%
% Van der Meer, J.W., 2002. Technical Report Wave Run-up and
% Wave Overtopping at Dikes. TAW Technical Advisory Committee on
% Flood Defence, The Netherlands.
%
% FEMA. 2007,  Atlantic Ocean and Gulf of Mexico Coastal Guidelines Update
%
%
%
%------------------------------------------------------------------------
% CONFIG
%------------------------------------------------------------------------
fname='inpfiles/CM-158-1sta_ele_include.csv';   % file with station, elevation, include
                                        % third colunm is 0 for excluded points
imgname='logfiles/CM-158-1-runup';
SWEL=8.816;  % 100-yr still water level including wave setup.
H0=1.833;    % significant wave height at toe of structure
Tp=2.6668;    % peak period, 1/fma,
T0=Tp/1.1;
gamma_berm=1;   % this may get changed automatically below
gamma_rough=0.8;
gamma_beta=1;
gamma_perm=1;
setupAtToe=-0.0012402;
maxSetup=0.06101;   % only used in case of berm/shallow foreshore weighted average
plotTitle='Iterative TAW for CM-158-1'
plotTitle =
Iterative TAW for CM-158-1
% END CONFIG
%------------------------------
SWEL=SWEL+setupAtToe
SWEL =
                8.8147598
SWEL_fore=SWEL+maxSetup
SWEL_fore =
                8.8757698
% FIND WAVELENGTH USING DEEPWATER DISPERSION RELATION
% using English units
L0=32.15/(2*pi)*T0^2
L0 =
          30.0743764412918
% Find Hb (Munk, 1949)
%Hb=H0/(3.3*(H0/L0)^(1/3))
%Db=-Hb/.78+SWEL;  % depth at breaking
% The toe elevation here is only used to determine the average
% structure slope, it is not used to depth limit the wave height.
% Any depth limiting or other modification of the wave height
% to make it consitent with TAW guidance should be performed
% prior to the input of the significant wave height given above.
Ztoe=SWEL-1.5*H0
Ztoe =
                6.0652598
% read the transect
[sta,dep,inc] = textread(fname,'%n%n%n%*[^\n]','delimiter',',','headerlines',0);
% remove unselected points
k=find(inc==0);
sta(k)=[];
dep(k)=[];
sta_org=sta;  % used for plotting purposes
dep_org=dep;
% initial guess at maximum run-up elevation to estimate slope
Z2=SWEL+1.5*H0
Z2 =
               11.5642598
% determine station at the max runup and -1.5*H0 (i.e. the toe)
top_sta=-999;
toe_sta=-999;
for kk=1:length(sta)-1
    if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
       top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
```

```
    end
    if  ((Ztoe > dep(kk)) & (Ztoe <= dep(kk+1)))    % here is the intersection of Ztoe with profile
       toe_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Ztoe)
    end
end
toe_sta =
         13.0058230890364
top_sta =
         25.2593016206604
% check to make sure we got them, if not extend the end slopes outward
S=diff(dep)./diff(sta);
if toe_sta==-999
   dy=dep(1)-Ztoe;
   toe_sta=sta(1)-dy/S(1)
end
if top_sta==-999
   dy=Z2-dep(end);
   top_sta=sta(end)+dy/S(end)
end
% just so the reader can tell the values aren't -999 anymore
top_sta
top_sta =
         25.2593016206604
toe_sta
toe_sta =
         13.0058230890364
% check for case where the toe of slope is below SWL-1.5*H0
% in this case interpolate setup from the setupAtToe(really setup as first station), and the max setup
% also un-include points seaward of SWL-1.5*H0
if Ztoe > dep(1)
   dd=SWEL_fore-dep;
   k=find(dd<0,1); % k is index of first land point
   staAtSWL=interp1(dep(k-1:k),sta(k-1:k),SWEL_fore);
   dsta=staAtSWL-sta(1);
   dsetup=maxSetup-setupAtToe;
   dsetdsta=dsetup/dsta;
   setup=setupAtToe+dsetdsta*(toe_sta-sta(1));
   sprintf('-!!- Location of SWEL-1.5*H0 is %4.1f ft landward of toe of slope',dsta)
   sprintf('-!!- Setup is interpolated between setup at toe of slope and max setup')
   sprintf('-!!-       setup is adjusted to %4.2f feet',setup)
   SWEL=SWEL-setupAtToe+setup;
   sprintf('-!!-       SWEL is adjusted to %4.2f feet',SWEL)
   k=find(dep < SWEL-1.5*H0)
   sta(k)=[];
   dep(k)=[];
else
   sprintf('-!!- The User has selected a starting point that is %4.2f feet above the elevation of SWEL-1.5H0\n',dep(1
   sprintf('-!!- This may be reasonable for some cases.  However the user may want to consider:\n')
   sprintf('-!!-   1) Selecting a starting point that is at or below %4.2f feet elevation, or\n', Ztoe)
   sprintf('-!!-   2) Reducing the incident wave height to a depth limited condition.\n')
end
ans =
-!!- Location of SWEL-1.5*H0 is 27.0 ft landward of toe of slope
ans =
-!!- Setup is interpolated between setup at toe of slope and max setup
ans =
-!!-       setup is adjusted to 0.05 feet
ans =
-!!-       SWEL is adjusted to 8.87 feet
k =
     1
     2
     3
     4
     5
     6
     7
     8
     9
    10
    11
    12
    13
    14
    15
    16
    17
    18
    19
    20
    21
    22
    23
    24
% now iterate converge on a runup elevation
tol=0.01;  % convergence criteria
R2del=999;
R2_new=3*H0; %initial guess
R2=R2_new;
iter=0;
```

```
R2_all=[];
topStaAll=[];
Berm_Segs=[];
TAW_ALWAYS_VALID=1;
while(abs(R2del) > tol && iter <= 25)
    iter=iter+1;
    sprintf ('!----------------- STARTING ITERATION %d -----------------!',iter)
    % elevation of toe of slope
    Ztoe
    % station of toe slope (relative to 0-NAVD88 shoreline
    toe_sta
    % station of top of slope/extent of 2% run-up
    top_sta
    % elevation of top of slope/extent of 2% run-up
    Z2
    % incident significant wave height
    H0
    % incident spectral peak wave period
    Tp
    % incident spectral mean wave period
    T0

    R2=R2_new
    Z2=R2+SWEL
    % determine slope for this iteration
    top_sta=-999;
    for kk=1:length(sta)-1
       if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
          top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
          break;
       end
    end
    if top_sta==-999
       dy=Z2-dep(end);
       top_sta=sta(end)+dy/S(end)
    end

    % get the length of the slope (not accounting for berm)
    Lslope=top_sta-toe_sta
    % loop over profile segments to determine berm factor
    % re-calculate influence of depth of berm based on this run-up elevation
    % check for berm, berm width, berm height
    berm_width=0;
    rdh_sum=0;
    Berm_Segs=[];
    Berm_Heights=[];
    for kk=1:length(sta)-1
       ddep=dep(kk+1)-dep(kk);
       dsta=sta(kk+1)-sta(kk);
       s=ddep/dsta;
       if  (s < 1/15)     % count it as a berm if slope is flatter than 1:15 (see TAW manual)
          sprintf ('Berm Factor Calculation: Iteration %d, Profile Segment: %d',iter,kk)
          berm_width=berm_width+dsta;   % tally the width of all berm segments
          % compute the rdh for this segment and weight it by the segment length
          dh=SWEL-(dep(kk)+dep(kk+1))/2
          if  dh < 0
              chi=R2;
          else
              chi=2* H0;
          end
          if (dh <= R2 & dh >=-2*H0)
             rdh=(0.5-0.5*cos(3.14159*dh/chi)) ;
          else
             rdh=1;
          end
          rdh_sum=rdh_sum + rdh * dsta
          Berm_Segs=[Berm_Segs, kk];
          Berm_Heights=[Berm_Heights, (dep(kk)+dep(kk+1))/2];
       end
       if dep(kk) >= Z2  % jump out of loop if we reached limit of run-up for this iteration
          break
       end
    end
    sprintf ('!------- End Berm Factor Calculation, Iter: %d ---------!',iter)
    berm_width
    rB=berm_width/Lslope
    if (berm_width > 0)
       rdh_mean=rdh_sum/berm_width
    else
       rdh_mean=1
    end
    gamma_berm=1- rB * (1-rdh_mean)
    if gamma_berm > 1
       gamma_berm=1
    end
    if gamma_berm < 0.6
       gamma_berm =0.6
    end
    % Iribarren number
    slope=(Z2-Ztoe)/(Lslope-berm_width)
```

```
    Irb=(slope/(sqrt(H0/L0)))
    % runup height
    gamma_berm
    gamma_perm
    gamma_beta
    gamma_rough
    gamma=gamma_berm*gamma_perm*gamma_beta*gamma_rough
    % check validity
    TAW_VALID=1;
    if (Irb*gamma_berm < 0.5 | Irb*gamma_berm > 10 )
       sprintf('!!! - - Iribaren number: %6.2f is outside the valid range (0.5-10), TAW NOT VALID - - !!!\n', Irb*gam
       TAW_VALID=0;
    else
       sprintf('!!! - - Iribaren number: %6.2f is in the valid range (0.5-10), TAW RECOMMENDED - - !!!\n', Irb*gamma_
    end
    islope=1/slope;
    if (slope < 1/8 | slope > 1)
       sprintf('!!! - - slope: 1:%3.1f V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!\n', islope)
       TAW_VALID=0;
    else
       sprintf('!!! - - slope: 1:%3.1f V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!\n', islope)
    end
    if TAW_VALID == 0
       TAW_ALWAYS_VALID=0;
    end

    if (Irb*gamma_berm < 1.8)
       R2_new=gamma*H0*1.77*Irb
    else
       R2_new=gamma*H0*(4.3-(1.6/sqrt(Irb)))
    end
    % check to see if we need to evaluate a shallow foreshore
    if berm_width > 0.25 * L0;
       disp ('!   Berm_width is greater than 1/4 wave length')
       disp ('!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on ber
       % do the foreshore calculation
       fore_H0=0.78*(SWEL_fore-min(Berm_Heights))
       % get upper slope
       fore_toe_sta=-999;
       fore_toe_dep=-999;
       for kk=length(dep)-1:-1:1
          ddep=dep(kk+1)-dep(kk);
          dsta=sta(kk+1)-sta(kk);
          s=ddep/dsta;
          if s < 1/15
             break
          end
          fore_toe_sta=sta(kk);
          fore_toe_dep=dep(kk);
          upper_slope=(Z2-fore_toe_dep)/(top_sta-fore_toe_sta)
       end
       fore_Irb=upper_slope/(sqrt(fore_H0/L0));
       fore_gamma=gamma_perm*gamma_beta*gamma_rough;
       if (fore_Irb < 1.8)
          fore_R2=fore_gamma*fore_H0*1.77*fore_Irb;
       else
          fore_R2=fore_gamma*fore_H0*(4.3-(1.6/sqrt(fore_Irb)));
       end
       if berm_width >= L0
          R2_new=fore_R2
          disp ('berm is wider than one wavelength, use full shallow foreshore solution');
       else
          w2=(berm_width-0.25*L0)/(0.75*L0)
          w1=1-w2
          R2_new=w2*fore_R2 + w1*R2_new
       end
    end % end berm width check
    % convergence criterion
    R2del=abs(R2-R2_new)
    R2_all(iter)=R2_new;
    % get the new top station (for plot purposes)
    Z2=R2_new+SWEL
    top_sta=-999;
    for kk=1:length(sta)-1
       if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
          top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
          break;
       end
    end
    if top_sta==-999
       dy=Z2-dep(end);
       top_sta=sta(end)+dy/S(end);
    end
    topStaAll(iter)=top_sta;
end
ans =
!----------------- STARTING ITERATION 1 -----------------!
Ztoe =
                6.0652598
toe_sta =
```

```
          13.0058230890364
top_sta =
          25.2593016206604
Z2 =
               11.5642598
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
R2 =
                    5.499
Z2 =
          14.3667865524122
top_sta =
         -32.8054111085127
Lslope =
         -45.8112341975491
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
         0.431923830491403
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
         0.877877093142679
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 10
dh =
         -2.60941244758778
rdh_sum =
          1.33790298328708
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 11
dh =
         -2.65889844758778
rdh_sum =
          1.81203344910653
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 12
dh =
         -2.70838444758778
rdh_sum =
          2.30028916609791
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 13
dh =
         -2.75787044758778
rdh_sum =
          2.80267952054878
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 14
dh =
         -2.80735644758778
rdh_sum =
          3.31920260203565
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 15
dh =
         -2.85684244758778
rdh_sum =
          3.84984520495081
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 16
dh =
         -2.90632844758778
rdh_sum =
          4.39458283905656
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 17
dh =
         -2.95581444758778
rdh_sum =
          4.95337974905832
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 18
dh =
         -3.00530044758779
rdh_sum =
          5.52618894318109
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 19
dh =
         -3.01099444758778
rdh_sum =
```

```
          6.10060690908255
ans =
!------- End Berm Factor Calculation, Iter: 1 ---------!
berm_width =
    12
rB =
        -0.261944481745528
rdh_mean =
         0.508383909090213
gamma_berm =
          1.12877612215113
gamma_berm =
     1
slope =
        -0.143597120311335
Irb =
        -0.581651448469581
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
                       0.8
gamma =
                       0.8
ans =
!!! - - Iribaren number:  -0.58 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:-7.0 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
         -1.50969262074336
!   Berm_width is greater than 1/4 wave length
!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on berm
fore_H0 =
              -1.951916616
{ Undefined function or variable ’upper_slope’.
}
%chk nld 20200220
clc;clear all;close all
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% config %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
datafile=’../data/transectdata.xls’;
tDIR=’../ADCIRC_returns/’; %location of transects
imgfile=’tpng’; runupname=’CM-runup’;
L_append=2;
csvoutpre=’inpfiles/’;
templatefile=’TAW_template.txt’;
templatelines=401;
%config
CITYNAME=’The Town of Harpswell’;
COUNTY=’Cumberland’;
ENGINEER=’SJH’;
DATE=date;
tawfilename=’TAW_iterative.m’
tawfilename =
TAW_iterative.m
[num,txt,raw]=xlsread(datafile);
for i=2:size(raw,1)
fnames{i-1}=raw{i,1};
end
uiopen(’C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswell\Additional_Transects\1_input\User_input.m’,1)
TAW_iterative_writer
tawfilename =
TAW_iterative.m
fid =
     3
ans =
     0
fid2 =
     4
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswell
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
```

```
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
ans =
     0
TAW_iterative_writer
tawfilename =
TAW_iterative.m
fid =
     3
ans =
     0
fid2 =
     4
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswell
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback(’TAW_iterative_writer’, ’C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline(’C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
```

```
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
[ > In <a href="matlab:matlab.internal.language.introspective.errorDocCallback('TAW_iterative_writer', 'C:\FEMA-Trans
:bold">TAW_iterative_writer</a> (<a href="matlab: opentoline('C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswel
[ Warning: Inputs must be character vectors, cell arrays of character vectors, or string
arrays.]
ans =
     0
clear all
close all
format long g
diary logfiles/CM-145-DIARY.txt  % open a diary file to record calculations
diary on         % begin recording
% FEMA appeal for The Town of Harpswell, Cumberland county, Maine
% TRANSECT ID: CM-158-1
% calculation by SJH, Ransom Consulting, Inc. 26-Feb-2020
% 100-year wave runup using TAW methodology
% including berm and weighted average with foreshore if necessary
%
% chk nld 20200220
%
% This script assumes that the incident wave conditions provided
% as input in the configuration section below are the
% appropriate values located at the end of the foreshore
% or toe of the slope on which the run-up is being calculated
% the script does not attempt to apply a depth limit or any other
% transformation to the incident wave conditions other than
% conversion of the peak wave period to the spectral mean wave
% as recommended in the references below
%
% references:
%
% Van der Meer, J.W., 2002. Technical Report Wave Run-up and
% Wave Overtopping at Dikes. TAW Technical Advisory Committee on
```

```
% Flood Defence, The Netherlands.
%
% FEMA. 2007,  Atlantic Ocean and Gulf of Mexico Coastal Guidelines Update
%
%
%
%------------------------------------------------------------------------
% CONFIG
%------------------------------------------------------------------------
fname='inpfiles/CM-158-1sta_ele_include.csv';   % file with station, elevation, include
                                    % third colunm is 0 for excluded points
imgname='logfiles/CM-158-1-runup';
SWEL=8.816;  % 100-yr still water level including wave setup.
H0=1.833;    % significant wave height at toe of structure
Tp=2.6668;    % peak period, 1/fma,
T0=Tp/1.1;
gamma_berm=1;   % this may get changed automatically below
gamma_rough=0.6;
gamma_beta=1;
gamma_perm=1;
setupAtToe=-0.0012402;
maxSetup=0.06101;   % only used in case of berm/shallow foreshore weighted average
plotTitle='Iterative TAW for CM-158-1'
plotTitle =
Iterative TAW for CM-158-1
% END CONFIG
%------------------------------
SWEL=SWEL+setupAtToe
SWEL =
                8.8147598
SWEL_fore=SWEL+maxSetup
SWEL_fore =
                8.8757698
% FIND WAVELENGTH USING DEEPWATER DISPERSION RELATION
% using English units
L0=32.15/(2*pi)*T0^2
L0 =
          30.0743764412918
% Find Hb (Munk, 1949)
%Hb=H0/(3.3*(H0/L0)^(1/3))
%Db=-Hb/.78+SWEL;  % depth at breaking
% The toe elevation here is only used to determine the average
% structure slope, it is not used to depth limit the wave height.
% Any depth limiting or other modification of the wave height
% to make it consitent with TAW guidance should be performed
% prior to the input of the significant wave height given above.
Ztoe=SWEL-1.5*H0
Ztoe =
                6.0652598
% read the transect
[sta,dep,inc] = textread(fname,'%n%n%n%*[^\n]','delimiter',',','headerlines',0);
% remove unselected points
k=find(inc==0);
sta(k)=[];
dep(k)=[];
sta_org=sta;  % used for plotting purposes
dep_org=dep;
% initial guess at maximum run-up elevation to estimate slope
Z2=SWEL+1.5*H0
Z2 =
               11.5642598
% determine station at the max runup and -1.5*H0 (i.e. the toe)
top_sta=-999;
toe_sta=-999;
for kk=1:length(sta)-1
    if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
       top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
    end
    if  ((Ztoe > dep(kk)) & (Ztoe <= dep(kk+1)))    % here is the intersection of Ztoe with profile
       toe_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Ztoe)
    end
end
toe_sta =
          13.0058230890364
% check to make sure we got them, if not extend the end slopes outward
S=diff(dep)./diff(sta);
if toe_sta==-999
   dy=dep(1)-Ztoe;
   toe_sta=sta(1)-dy/S(1)
end
if top_sta==-999
   dy=Z2-dep(end);
   top_sta=sta(end)+dy/S(end)
end
top_sta =
          25.2593016206604
% just so the reader can tell the values aren't -999 anymore
top_sta
top_sta =
          25.2593016206604
```

```
toe_sta
toe_sta =
         13.0058230890364
% check for case where the toe of slope is below SWL-1.5*H0
% in this case interpolate setup from the setupAtToe(really setup as first station), and the max setup
% also un-include points seaward of SWL-1.5*H0
if Ztoe > dep(1)
   dd=SWEL_fore-dep;
   k=find(dd<0,1); % k is index of first land point
   staAtSWL=interp1(dep(k-1:k),sta(k-1:k),SWEL_fore);
   dsta=staAtSWL-sta(1);
   dsetup=maxSetup-setupAtToe;
   dsetdsta=dsetup/dsta;
   setup=setupAtToe+dsetdsta*(toe_sta-sta(1));
   sprintf('-!!- Location of SWEL-1.5*H0 is %4.1f ft landward of toe of slope',dsta)
   sprintf('-!!- Setup is interpolated between setup at toe of slope and max setup')
   sprintf('-!!-       setup is adjusted to %4.2f feet',setup)
   SWEL=SWEL-setupAtToe+setup;
   sprintf('-!!-       SWEL is adjusted to %4.2f feet',SWEL)
   k=find(dep < SWEL-1.5*H0)
   sta(k)=[];
   dep(k)=[];
else
   sprintf('-!!- The User has selected a starting point that is %4.2f feet above the elevation of SWEL-1.5H0\n',dep(1
   sprintf('-!!- This may be reasonable for some cases.  However the user may want to consider:\n')
   sprintf('-!!-   1) Selecting a starting point that is at or below %4.2f feet elevation, or\n', Ztoe)
   sprintf('-!!-   2) Reducing the incident wave height to a depth limited condition.\n')
end
ans =
-!!- Location of SWEL-1.5*H0 is 27.0 ft landward of toe of slope
ans =
-!!- Setup is interpolated between setup at toe of slope and max setup
ans =
-!!-       setup is adjusted to 0.05 feet
ans =
-!!-       SWEL is adjusted to 8.87 feet
k =
     1
     2
     3
     4
     5
     6
     7
     8
     9
    10
    11
    12
    13
    14
    15
    16
    17
    18
    19
    20
    21
    22
    23
    24
% now iterate converge on a runup elevation
tol=0.01;  % convergence criteria
R2del=999;
R2_new=3*H0; %initial guess
R2=R2_new;
iter=0;
R2_all=[];
topStaAll=[];
Berm_Segs=[];
TAW_ALWAYS_VALID=1;
while(abs(R2del) > tol && iter <= 25)
    iter=iter+1;
    sprintf ('!----------------- STARTING ITERATION %d -----------------!',iter)
    % elevation of toe of slope
    Ztoe
    % station of toe slope (relative to 0-NAVD88 shoreline
    toe_sta
    % station of top of slope/extent of 2% run-up
    top_sta
    % elevation of top of slope/extent of 2% run-up
    Z2
    % incident significant wave height
    H0
    % incident spectral peak wave period
    Tp
    % incident spectral mean wave period
    T0

    R2=R2_new
```

```
Z2=R2+SWEL
% determine slope for this iteration
top_sta=-999;
for kk=1:length(sta)-1
   if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
      top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
      break;
   end
end
if top_sta==-999
   dy=Z2-dep(end);
   top_sta=sta(end)+dy/S(end)
end

% get the length of the slope (not accounting for berm)
Lslope=top_sta-toe_sta
% loop over profile segments to determine berm factor
% re-calculate influence of depth of berm based on this run-up elevation
% check for berm, berm width, berm height
berm_width=0;
rdh_sum=0;
Berm_Segs=[];
Berm_Heights=[];
for kk=1:length(sta)-1
   ddep=dep(kk+1)-dep(kk);
   dsta=sta(kk+1)-sta(kk);
   s=ddep/dsta;
   if  (s < 1/15)     % count it as a berm if slope is flatter than 1:15 (see TAW manual)
      sprintf ('Berm Factor Calculation: Iteration %d, Profile Segment: %d',iter,kk)
      berm_width=berm_width+dsta;   % tally the width of all berm segments
      % compute the rdh for this segment and weight it by the segment length
      dh=SWEL-(dep(kk)+dep(kk+1))/2
      if  dh < 0
          chi=R2;
      else
          chi=2* H0;
      end
      if (dh <= R2 & dh >=-2*H0)
         rdh=(0.5-0.5*cos(3.14159*dh/chi)) ;
      else
         rdh=1;
      end
      rdh_sum=rdh_sum + rdh * dsta
      Berm_Segs=[Berm_Segs, kk];
      Berm_Heights=[Berm_Heights, (dep(kk)+dep(kk+1))/2];
   end
   if dep(kk) >= Z2  % jump out of loop if we reached limit of run-up for this iteration
      break
   end
end
sprintf ('!------- End Berm Factor Calculation, Iter: %d ---------!',iter)
berm_width
rB=berm_width/Lslope
if (berm_width > 0)
   rdh_mean=rdh_sum/berm_width
else
   rdh_mean=1
end
gamma_berm=1- rB * (1-rdh_mean)
if gamma_berm > 1
   gamma_berm=1
end
if gamma_berm < 0.6
   gamma_berm =0.6
end
% Iribarren number
slope=(Z2-Ztoe)/(Lslope-berm_width)
Irb=(slope/(sqrt(H0/L0)))
% runup height
gamma_berm
gamma_perm
gamma_beta
gamma_rough
gamma=gamma_berm*gamma_perm*gamma_beta*gamma_rough
% check validity
TAW_VALID=1;
if (Irb*gamma_berm < 0.5 | Irb*gamma_berm > 10 )
   sprintf('!!! - - Iribaren number: %6.2f is outside the valid range (0.5-10), TAW NOT VALID - - !!!\n', Irb*gam
   TAW_VALID=0;
else
   sprintf('!!! - - Iribaren number: %6.2f is in the valid range (0.5-10), TAW RECOMMENDED - - !!!\n', Irb*gamma_
end
islope=1/slope;
if (slope < 1/8 | slope > 1)
   sprintf('!!! - - slope: 1:%3.1f V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!\n', islope)
   TAW_VALID=0;
else
   sprintf('!!! - - slope: 1:%3.1f V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!\n', islope)
end
if TAW_VALID == 0
```

```
       TAW_ALWAYS_VALID=0;
    end

    if (Irb*gamma_berm < 1.8)
       R2_new=gamma*H0*1.77*Irb
    else
       R2_new=gamma*H0*(4.3-(1.6/sqrt(Irb)))
    end
    % check to see if we need to evaluate a shallow foreshore
    if berm_width > 0.25 * L0;
       disp ('!   Berm_width is greater than 1/4 wave length')
       disp ('!   Runup will be weighted average with foreshore calculation assuming depth limited wave height on ber
       % do the foreshore calculation
       fore_H0=0.78*(SWEL_fore-min(Berm_Heights))
       % get upper slope
       fore_toe_sta=-999;
       fore_toe_dep=-999;
       for kk=length(dep)-1:-1:1
          ddep=dep(kk+1)-dep(kk);
          dsta=sta(kk+1)-sta(kk);
          s=ddep/dsta;
          if s < 1/15
             break
          end
          fore_toe_sta=sta(kk);
          fore_toe_dep=dep(kk);
          upper_slope=(Z2-fore_toe_dep)/(top_sta-fore_toe_sta)
       end
       fore_Irb=upper_slope/(sqrt(fore_H0/L0));
       fore_gamma=gamma_perm*gamma_beta*gamma_rough;
       if (fore_Irb < 1.8)
          fore_R2=fore_gamma*fore_H0*1.77*fore_Irb;
       else
          fore_R2=fore_gamma*fore_H0*(4.3-(1.6/sqrt(fore_Irb)));
       end
       if berm_width >= L0
          R2_new=fore_R2
          disp ('berm is wider than one wavelength, use full shallow foreshore solution');
       else
          w2=(berm_width-0.25*L0)/(0.75*L0)
          w1=1-w2
          R2_new=w2*fore_R2 + w1*R2_new
       end
    end % end berm width check
    % convergence criterion
    R2del=abs(R2-R2_new)
    R2_all(iter)=R2_new;
    % get the new top station (for plot purposes)
    Z2=R2_new+SWEL
    top_sta=-999;
    for kk=1:length(sta)-1
       if ((Z2 > dep(kk)) & (Z2 <= dep(kk+1)))    % here is the intersection of z2 with profile
          top_sta=interp1(dep(kk:kk+1),sta(kk:kk+1),Z2)
          break;
       end
    end
    if top_sta==-999
       dy=Z2-dep(end);
       top_sta=sta(end)+dy/S(end);
    end
    topStaAll(iter)=top_sta;
end
ans =
!----------------- STARTING ITERATION 1 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          25.2593016206604
Z2 =
               11.5642598
H0 =
                    1.833
Tp =
                   2.6668
T0 =
          2.42436363636364
R2 =
                    5.499
Z2 =
          14.3667865524122
top_sta =
           81.892021024375
Lslope =
          68.8861979353386
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 8
dh =
         -2.51044044758778
```

```
rdh_sum =
         0.431923830491403
ans =
Berm Factor Calculation: Iteration 1, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
         0.877877093142679
ans =
!------- End Berm Factor Calculation, Iter: 1 ---------!
berm_width =
     2
rB =
        0.0290333921735286
rdh_mean =
         0.438938546571339
gamma_berm =
         0.983710482789156
slope =
         0.124114197079009
Irb =
         0.502734333043211
gamma_berm =
         0.983710482789156
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
                       0.6
gamma =
         0.590226289673493
ans =
!!! - - Iribaren number:   0.49 is outside the valid range (0.5-10), TAW NOT VALID - - !!!
ans =
!!! - - slope: 1:8.1 V:H is outside the valid range (1:8 - 1:1), TAW NOT VALID - - !!!
R2_new =
         0.962704111229344
R2del =
          4.53629588877066
Z2 =
          9.83049066364156
top_sta =
          18.3667823704713
ans =
!----------------- STARTING ITERATION 2 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          18.3667823704713
Z2 =
          9.83049066364156
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
         0.962704111229344
Z2 =
          9.83049066364156
top_sta =
          18.3667823704713
Lslope =
          5.36095928143491
ans =
!------- End Berm Factor Calculation, Iter: 2 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
         0.702342746135121
Irb =
          2.84489462411141
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
                       0.6
```

```
gamma =
                    0.6
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          3.68586066351234
R2del =
            2.723156552283
Z2 =
          12.5536472159246
ans =
!----------------- STARTING ITERATION 3 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          45.2525808496253
Z2 =
          12.5536472159246
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          3.68586066351234
Z2 =
          12.5536472159246
top_sta =
          45.2525808496253
Lslope =
          32.2467577605889
ans =
Berm Factor Calculation: Iteration 3, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
          0.769370196603533
ans =
Berm Factor Calculation: Iteration 3, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
          1.55626272407322
ans =
!------- End Berm Factor Calculation, Iter: 3 ---------!
berm_width =
     2
rB =
        0.0620217392039438
rdh_mean =
          0.778131362036607
gamma_berm =
            0.9862393211987
slope =
          0.214515138028408
Irb =
          0.868910466187299
gamma_berm =
            0.9862393211987
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
                    0.6
gamma =
          0.59174359271922
ans =
!!! - - Iribaren number:   0.86 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:4.7 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
           1.6681854306886
R2del =
          2.01767523282374
Z2 =
          10.5359719831008
top_sta =
          19.3712516293333
ans =
!----------------- STARTING ITERATION 4 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
```

```
top_sta =
          19.3712516293333
Z2 =
          10.5359719831008
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
           1.6681854306886
Z2 =
          10.5359719831008
top_sta =
          19.3712516293333
Lslope =
          6.36542854029688
ans =
!------- End Berm Factor Calculation, Iter: 4 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
         0.702342686717572
Irb =
          2.84489438343595
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
                       0.6
gamma =
                       0.6
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
           3.6858606193821
R2del =
           2.0176751886935
Z2 =
          12.5536471717943
ans =
!----------------- STARTING ITERATION 5 -----------------!
Ztoe =
                  6.0652598
toe_sta =
          13.0058230890364
top_sta =
          45.2525799578532
Z2 =
          12.5536471717943
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
           3.6858606193821
Z2 =
          12.5536471717943
top_sta =
          45.2525799578532
Lslope =
          32.2467568688168
ans =
Berm Factor Calculation: Iteration 5, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
         0.769370207395069
ans =
Berm Factor Calculation: Iteration 5, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
           1.5562627455625
ans =
!------- End Berm Factor Calculation, Iter: 5 ---------!
```

```
berm_width =
     2
rB =
        0.0620217409191322
rdh_mean =
         0.778131372781251
gamma_berm =
         0.986239321484555
slope =
            0.214515142894
Irb =
         0.868910485895764
gamma_berm =
         0.986239321484555
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
                       0.6
gamma =
         0.591743592890733
ans =
!!! - - Iribaren number:   0.86 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:4.7 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
         1.66818546900959
R2del =
         2.01767515037252
Z2 =
         10.5359720214218
top_sta =
         19.3712516838949
ans =
!----------------- STARTING ITERATION 6 -----------------!
Ztoe =
                6.0652598
toe_sta =
         13.0058230890364
top_sta =
         19.3712516838949
Z2 =
         10.5359720214218
H0 =
                     1.833
Tp =
                    2.6668
T0 =
         2.42436363636364
R2 =
         1.66818546900959
Z2 =
         10.5359720214218
top_sta =
         19.3712516838949
Lslope =
         6.36542859485851
ans =
!------- End Berm Factor Calculation, Iter: 6 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
         0.702342686717574
Irb =
         2.84489438343596
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
                       0.6
gamma =
                       0.6
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
         3.68586061938211
R2del =
         2.01767515037252
```

```
Z2 =
         12.5536471717943
ans =
!----------------- STARTING ITERATION 7 -----------------!
Ztoe =
                6.0652598
toe_sta =
         13.0058230890364
top_sta =
         45.2525799578532
Z2 =
         12.5536471717943
H0 =
                  1.833
Tp =
                 2.6668
T0 =
         2.42436363636364
R2 =
         3.68586061938211
Z2 =
         12.5536471717943
top_sta =
         45.2525799578532
Lslope =
         32.2467568688168
ans =
Berm Factor Calculation: Iteration 7, Profile Segment: 8
dh =
        -2.51044044758778
rdh_sum =
         0.769370207395069
ans =
Berm Factor Calculation: Iteration 7, Profile Segment: 9
dh =
        -2.55992644758778
rdh_sum =
          1.5562627455625
ans =
!------- End Berm Factor Calculation, Iter: 7 ---------!
berm_width =
     2
rB =
       0.0620217409191321
rdh_mean =
         0.778131372781251
gamma_berm =
         0.986239321484555
slope =
           0.214515142894
Irb =
         0.868910485895763
gamma_berm =
         0.986239321484555
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
                    0.6
gamma =
         0.591743592890733
ans =
!!! - - Iribaren number:   0.86 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:4.7 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
         1.66818546900958
R2del =
         2.01767515037252
Z2 =
         10.5359720214218
top_sta =
         19.3712516838949
ans =
!----------------- STARTING ITERATION 8 -----------------!
Ztoe =
                6.0652598
toe_sta =
         13.0058230890364
top_sta =
         19.3712516838949
Z2 =
         10.5359720214218
H0 =
                  1.833
Tp =
                 2.6668
T0 =
         2.42436363636364
```

```
R2 =
          1.66818546900958
Z2 =
          10.5359720214218
top_sta =
          19.3712516838949
Lslope =
          6.36542859485851
ans =
!------- End Berm Factor Calculation, Iter: 8 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
         0.702342686717574
Irb =
          2.84489438343596
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
                       0.6
gamma =
                       0.6
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          3.68586061938211
R2del =
          2.01767515037252
Z2 =
          12.5536471717943
ans =
!----------------- STARTING ITERATION 9 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          45.2525799578532
Z2 =
          12.5536471717943
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          3.68586061938211
Z2 =
          12.5536471717943
top_sta =
          45.2525799578532
Lslope =
          32.2467568688168
ans =
Berm Factor Calculation: Iteration 9, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
         0.769370207395069
ans =
Berm Factor Calculation: Iteration 9, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
           1.5562627455625
ans =
!------- End Berm Factor Calculation, Iter: 9 ---------!
berm_width =
     2
rB =
        0.0620217409191321
rdh_mean =
          0.778131372781251
gamma_berm =
          0.986239321484555
slope =
             0.214515142894
```

```
Irb =
         0.868910485895763
gamma_berm =
         0.986239321484555
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
                        0.6
gamma =
         0.591743592890733
ans =
!!! - - Iribaren number:   0.86 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:4.7 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
         1.66818546900958
R2del =
         2.01767515037252
Z2 =
         10.5359720214218
top_sta =
         19.3712516838949
ans =
!----------------- STARTING ITERATION 10 -----------------!
Ztoe =
                6.0652598
toe_sta =
         13.0058230890364
top_sta =
         19.3712516838949
Z2 =
         10.5359720214218
H0 =
                    1.833
Tp =
                   2.6668
T0 =
         2.42436363636364
R2 =
         1.66818546900958
Z2 =
         10.5359720214218
top_sta =
         19.3712516838949
Lslope =
         6.36542859485851
ans =
!------- End Berm Factor Calculation, Iter: 10 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
         0.702342686717574
Irb =
          2.84489438343596
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
                        0.6
gamma =
                        0.6
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
         3.68586061938211
R2del =
         2.01767515037252
Z2 =
         12.5536471717943
ans =
!----------------- STARTING ITERATION 11 -----------------!
Ztoe =
                6.0652598
toe_sta =
         13.0058230890364
top_sta =
         45.2525799578532
```

```
Z2 =
          12.5536471717943
H0 =
                      1.833
Tp =
                     2.6668
T0 =
          2.42436363636364
R2 =
          3.68586061938211
Z2 =
          12.5536471717943
top_sta =
          45.2525799578532
Lslope =
          32.2467568688168
ans =
Berm Factor Calculation: Iteration 11, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
          0.769370207395069
ans =
Berm Factor Calculation: Iteration 11, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
           1.5562627455625
ans =
!------- End Berm Factor Calculation, Iter: 11 ---------!
berm_width =
     2
rB =
        0.0620217409191321
rdh_mean =
          0.778131372781251
gamma_berm =
          0.986239321484555
slope =
            0.214515142894
Irb =
          0.868910485895763
gamma_berm =
          0.986239321484555
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
                       0.6
gamma =
          0.591743592890733
ans =
!!! - - Iribaren number:   0.86 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:4.7 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          1.66818546900958
R2del =
          2.01767515037252
Z2 =
          10.5359720214218
top_sta =
          19.3712516838949
ans =
!----------------- STARTING ITERATION 12 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          19.3712516838949
Z2 =
          10.5359720214218
H0 =
                      1.833
Tp =
                     2.6668
T0 =
          2.42436363636364
R2 =
          1.66818546900958
Z2 =
          10.5359720214218
top_sta =
          19.3712516838949
Lslope =
          6.36542859485851
ans =
!------- End Berm Factor Calculation, Iter: 12 ---------!
```

```
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
         0.702342686717574
Irb =
          2.84489438343596
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
                        0.6
gamma =
                        0.6
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          3.68586061938211
R2del =
          2.01767515037252
Z2 =
          12.5536471717943
ans =
!----------------- STARTING ITERATION 13 -----------------!
Ztoe =
                 6.0652598
toe_sta =
          13.0058230890364
top_sta =
          45.2525799578532
Z2 =
          12.5536471717943
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          3.68586061938211
Z2 =
          12.5536471717943
top_sta =
          45.2525799578532
Lslope =
          32.2467568688168
ans =
Berm Factor Calculation: Iteration 13, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
         0.769370207395069
ans =
Berm Factor Calculation: Iteration 13, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
           1.5562627455625
ans =
!------- End Berm Factor Calculation, Iter: 13 ---------!
berm_width =
     2
rB =
        0.0620217409191321
rdh_mean =
          0.778131372781251
gamma_berm =
          0.986239321484555
slope =
             0.214515142894
Irb =
          0.868910485895763
gamma_berm =
          0.986239321484555
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
                        0.6
```

```
gamma =
         0.591743592890733
ans =
!!! - - Iribaren number:   0.86 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:4.7 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          1.66818546900958
R2del =
          2.01767515037252
Z2 =
          10.5359720214218
top_sta =
          19.3712516838949
ans =
!----------------- STARTING ITERATION 14 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          19.3712516838949
Z2 =
          10.5359720214218
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          1.66818546900958
Z2 =
          10.5359720214218
top_sta =
          19.3712516838949
Lslope =
          6.36542859485851
ans =
!------- End Berm Factor Calculation, Iter: 14 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
         0.702342686717574
Irb =
          2.84489438343596
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
                       0.6
gamma =
                       0.6
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          3.68586061938211
R2del =
          2.01767515037252
Z2 =
          12.5536471717943
ans =
!----------------- STARTING ITERATION 15 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          45.2525799578532
Z2 =
          12.5536471717943
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          3.68586061938211
```

```
Z2 =
         12.5536471717943
top_sta =
         45.2525799578532
Lslope =
         32.2467568688168
ans =
Berm Factor Calculation: Iteration 15, Profile Segment: 8
dh =
        -2.51044044758778
rdh_sum =
        0.769370207395069
ans =
Berm Factor Calculation: Iteration 15, Profile Segment: 9
dh =
        -2.55992644758778
rdh_sum =
          1.5562627455625
ans =
!------- End Berm Factor Calculation, Iter: 15 ---------!
berm_width =
     2
rB =
       0.0620217409191321
rdh_mean =
        0.778131372781251
gamma_berm =
        0.986239321484555
slope =
           0.214515142894
Irb =
        0.868910485895763
gamma_berm =
        0.986239321484555
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
                      0.6
gamma =
        0.591743592890733
ans =
!!! - - Iribaren number:   0.86 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:4.7 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
         1.66818546900958
R2del =
         2.01767515037252
Z2 =
         10.5359720214218
top_sta =
         19.3712516838949
ans =
!----------------- STARTING ITERATION 16 -----------------!
Ztoe =
                6.0652598
toe_sta =
         13.0058230890364
top_sta =
         19.3712516838949
Z2 =
         10.5359720214218
H0 =
                    1.833
Tp =
                   2.6668
T0 =
         2.42436363636364
R2 =
         1.66818546900958
Z2 =
         10.5359720214218
top_sta =
         19.3712516838949
Lslope =
         6.36542859485851
ans =
!------- End Berm Factor Calculation, Iter: 16 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
        0.702342686717574
```

```
Irb =
          2.84489438343596
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
                        0.6
gamma =
                        0.6
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          3.68586061938211
R2del =
          2.01767515037252
Z2 =
          12.5536471717943
ans =
!----------------- STARTING ITERATION 17 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          45.2525799578532
Z2 =
          12.5536471717943
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          3.68586061938211
Z2 =
          12.5536471717943
top_sta =
          45.2525799578532
Lslope =
          32.2467568688168
ans =
Berm Factor Calculation: Iteration 17, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
         0.769370207395069
ans =
Berm Factor Calculation: Iteration 17, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
           1.5562627455625
ans =
!------- End Berm Factor Calculation, Iter: 17 ---------!
berm_width =
     2
rB =
        0.0620217409191321
rdh_mean =
          0.778131372781251
gamma_berm =
          0.986239321484555
slope =
             0.214515142894
Irb =
          0.868910485895763
gamma_berm =
          0.986239321484555
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
                        0.6
gamma =
          0.591743592890733
ans =
!!! - - Iribaren number:   0.86 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:4.7 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          1.66818546900958
R2del =
          2.01767515037252
```

```
Z2 =
         10.5359720214218
top_sta =
         19.3712516838949
ans =
!----------------- STARTING ITERATION 18 -----------------!
Ztoe =
                6.0652598
toe_sta =
         13.0058230890364
top_sta =
         19.3712516838949
Z2 =
         10.5359720214218
H0 =
                   1.833
Tp =
                  2.6668
T0 =
         2.42436363636364
R2 =
         1.66818546900958
Z2 =
         10.5359720214218
top_sta =
         19.3712516838949
Lslope =
         6.36542859485851
ans =
!------- End Berm Factor Calculation, Iter: 18 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
        0.702342686717574
Irb =
         2.84489438343596
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
                     0.6
gamma =
                     0.6
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
         3.68586061938211
R2del =
         2.01767515037252
Z2 =
         12.5536471717943
ans =
!----------------- STARTING ITERATION 19 -----------------!
Ztoe =
                6.0652598
toe_sta =
         13.0058230890364
top_sta =
         45.2525799578532
Z2 =
         12.5536471717943
H0 =
                   1.833
Tp =
                  2.6668
T0 =
         2.42436363636364
R2 =
         3.68586061938211
Z2 =
         12.5536471717943
top_sta =
         45.2525799578532
Lslope =
         32.2467568688168
ans =
Berm Factor Calculation: Iteration 19, Profile Segment: 8
dh =
        -2.51044044758778
```

```
rdh_sum =
         0.769370207395069
ans =
Berm Factor Calculation: Iteration 19, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
           1.5562627455625
ans =
!------- End Berm Factor Calculation, Iter: 19 ---------!
berm_width =
     2
rB =
        0.0620217409191321
rdh_mean =
         0.778131372781251
gamma_berm =
         0.986239321484555
slope =
            0.214515142894
Irb =
         0.868910485895763
gamma_berm =
         0.986239321484555
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
                       0.6
gamma =
         0.591743592890733
ans =
!!! - - Iribaren number:   0.86 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:4.7 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          1.66818546900958
R2del =
          2.01767515037252
Z2 =
          10.5359720214218
top_sta =
          19.3712516838949
ans =
!----------------- STARTING ITERATION 20 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          19.3712516838949
Z2 =
          10.5359720214218
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          1.66818546900958
Z2 =
          10.5359720214218
top_sta =
          19.3712516838949
Lslope =
          6.36542859485851
ans =
!------- End Berm Factor Calculation, Iter: 20 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
         0.702342686717574
Irb =
          2.84489438343596
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
                       0.6
```

```
gamma =
                    0.6
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          3.68586061938211
R2del =
          2.01767515037252
Z2 =
          12.5536471717943
ans =
!----------------- STARTING ITERATION 21 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          45.2525799578532
Z2 =
          12.5536471717943
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          3.68586061938211
Z2 =
          12.5536471717943
top_sta =
          45.2525799578532
Lslope =
          32.2467568688168
ans =
Berm Factor Calculation: Iteration 21, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
          0.769370207395069
ans =
Berm Factor Calculation: Iteration 21, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
           1.5562627455625
ans =
!------- End Berm Factor Calculation, Iter: 21 ---------!
berm_width =
     2
rB =
        0.0620217409191321
rdh_mean =
          0.778131372781251
gamma_berm =
          0.986239321484555
slope =
           0.214515142894
Irb =
          0.868910485895763
gamma_berm =
          0.986239321484555
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
                    0.6
gamma =
          0.591743592890733
ans =
!!! - - Iribaren number:   0.86 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:4.7 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          1.66818546900958
R2del =
          2.01767515037252
Z2 =
          10.5359720214218
top_sta =
          19.3712516838949
ans =
!----------------- STARTING ITERATION 22 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
```

```
top_sta =
          19.3712516838949
Z2 =
          10.5359720214218
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          1.66818546900958
Z2 =
          10.5359720214218
top_sta =
          19.3712516838949
Lslope =
          6.36542859485851
ans =
!------- End Berm Factor Calculation, Iter: 22 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
         0.702342686717574
Irb =
          2.84489438343596
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
                       0.6
gamma =
                       0.6
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          3.68586061938211
R2del =
          2.01767515037252
Z2 =
          12.5536471717943
ans =
!----------------- STARTING ITERATION 23 -----------------!
Ztoe =
                  6.0652598
toe_sta =
          13.0058230890364
top_sta =
          45.2525799578532
Z2 =
          12.5536471717943
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          3.68586061938211
Z2 =
          12.5536471717943
top_sta =
          45.2525799578532
Lslope =
          32.2467568688168
ans =
Berm Factor Calculation: Iteration 23, Profile Segment: 8
dh =
         -2.51044044758778
rdh_sum =
         0.769370207395069
ans =
Berm Factor Calculation: Iteration 23, Profile Segment: 9
dh =
         -2.55992644758778
rdh_sum =
           1.5562627455625
ans =
!------- End Berm Factor Calculation, Iter: 23 ---------!
```

```
berm_width =
     2
rB =
        0.0620217409191321
rdh_mean =
         0.778131372781251
gamma_berm =
         0.986239321484555
slope =
            0.214515142894
Irb =
         0.868910485895763
gamma_berm =
         0.986239321484555
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
                       0.6
gamma =
         0.591743592890733
ans =
!!! - - Iribaren number:   0.86 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:4.7 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          1.66818546900958
R2del =
          2.01767515037252
Z2 =
          10.5359720214218
top_sta =
          19.3712516838949
ans =
!----------------- STARTING ITERATION 24 -----------------!
Ztoe =
                6.0652598
toe_sta =
          13.0058230890364
top_sta =
          19.3712516838949
Z2 =
          10.5359720214218
H0 =
                     1.833
Tp =
                    2.6668
T0 =
          2.42436363636364
R2 =
          1.66818546900958
Z2 =
          10.5359720214218
top_sta =
          19.3712516838949
Lslope =
          6.36542859485851
ans =
!------- End Berm Factor Calculation, Iter: 24 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
         0.702342686717574
Irb =
          2.84489438343596
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
                       0.6
gamma =
                       0.6
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          3.68586061938211
R2del =
          2.01767515037252
```

```
Z2 =
         12.5536471717943
ans =
!----------------- STARTING ITERATION 25 -----------------!
Ztoe =
                6.0652598
toe_sta =
         13.0058230890364
top_sta =
         45.2525799578532
Z2 =
         12.5536471717943
H0 =
                   1.833
Tp =
                  2.6668
T0 =
         2.42436363636364
R2 =
         3.68586061938211
Z2 =
         12.5536471717943
top_sta =
         45.2525799578532
Lslope =
         32.2467568688168
ans =
Berm Factor Calculation: Iteration 25, Profile Segment: 8
dh =
        -2.51044044758778
rdh_sum =
         0.769370207395069
ans =
Berm Factor Calculation: Iteration 25, Profile Segment: 9
dh =
        -2.55992644758778
rdh_sum =
          1.5562627455625
ans =
!------- End Berm Factor Calculation, Iter: 25 ---------!
berm_width =
     2
rB =
       0.0620217409191321
rdh_mean =
         0.778131372781251
gamma_berm =
         0.986239321484555
slope =
           0.214515142894
Irb =
         0.868910485895763
gamma_berm =
         0.986239321484555
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
                      0.6
gamma =
         0.591743592890733
ans =
!!! - - Iribaren number:   0.86 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:4.7 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
         1.66818546900958
R2del =
         2.01767515037252
Z2 =
         10.5359720214218
top_sta =
         19.3712516838949
ans =
!----------------- STARTING ITERATION 26 -----------------!
Ztoe =
                6.0652598
toe_sta =
         13.0058230890364
top_sta =
         19.3712516838949
Z2 =
         10.5359720214218
H0 =
                   1.833
Tp =
                  2.6668
T0 =
         2.42436363636364
```

```
R2 =
          1.66818546900958
Z2 =
          10.5359720214218
top_sta =
          19.3712516838949
Lslope =
          6.36542859485851
ans =
!------- End Berm Factor Calculation, Iter: 26 ---------!
berm_width =
     0
rB =
     0
rdh_mean =
     1
gamma_berm =
     1
slope =
         0.702342686717574
Irb =
          2.84489438343596
gamma_berm =
     1
gamma_perm =
     1
gamma_beta =
     1
gamma_rough =
                       0.6
gamma =
                       0.6
ans =
!!! - - Iribaren number:   2.84 is in the valid range (0.5-10), TAW RECOMMENDED - - !!!
ans =
!!! - - slope: 1:1.4 V:H is in the valid range (1:8 - 1:1), TAW RECOMMENDED - - !!!
R2_new =
          3.68586061938211
R2del =
          2.01767515037252
Z2 =
          12.5536471717943
% final 2% runup elevation
Z2=R2_new+SWEL
Z2 =
          12.5536471717943
diary off
-1.000000e+00
-1.000000e+00
```