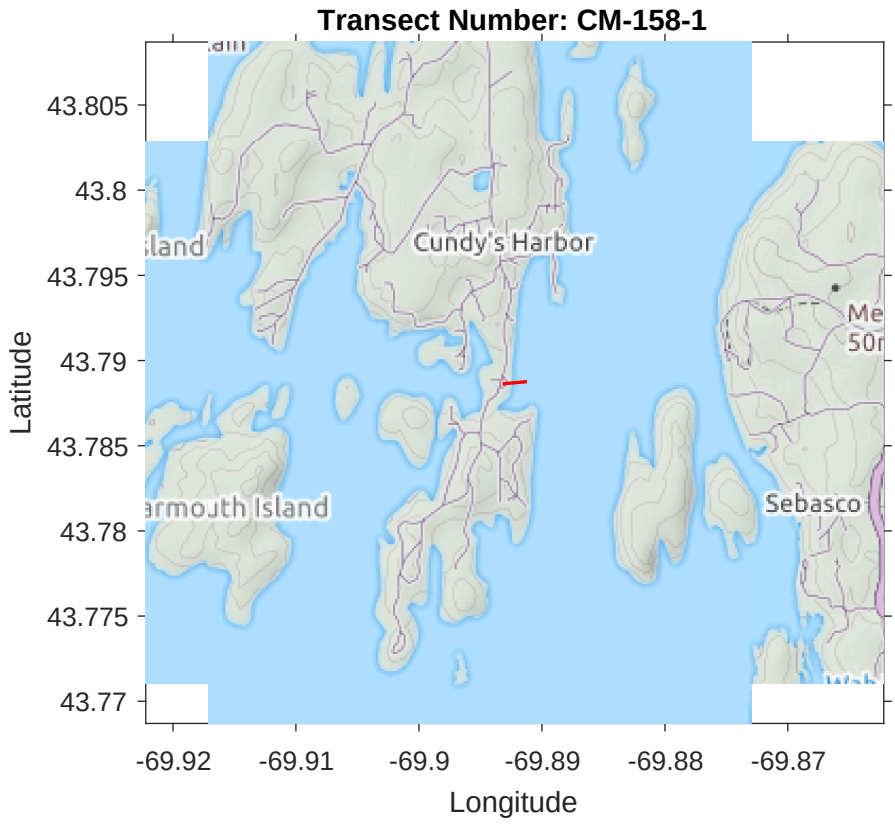
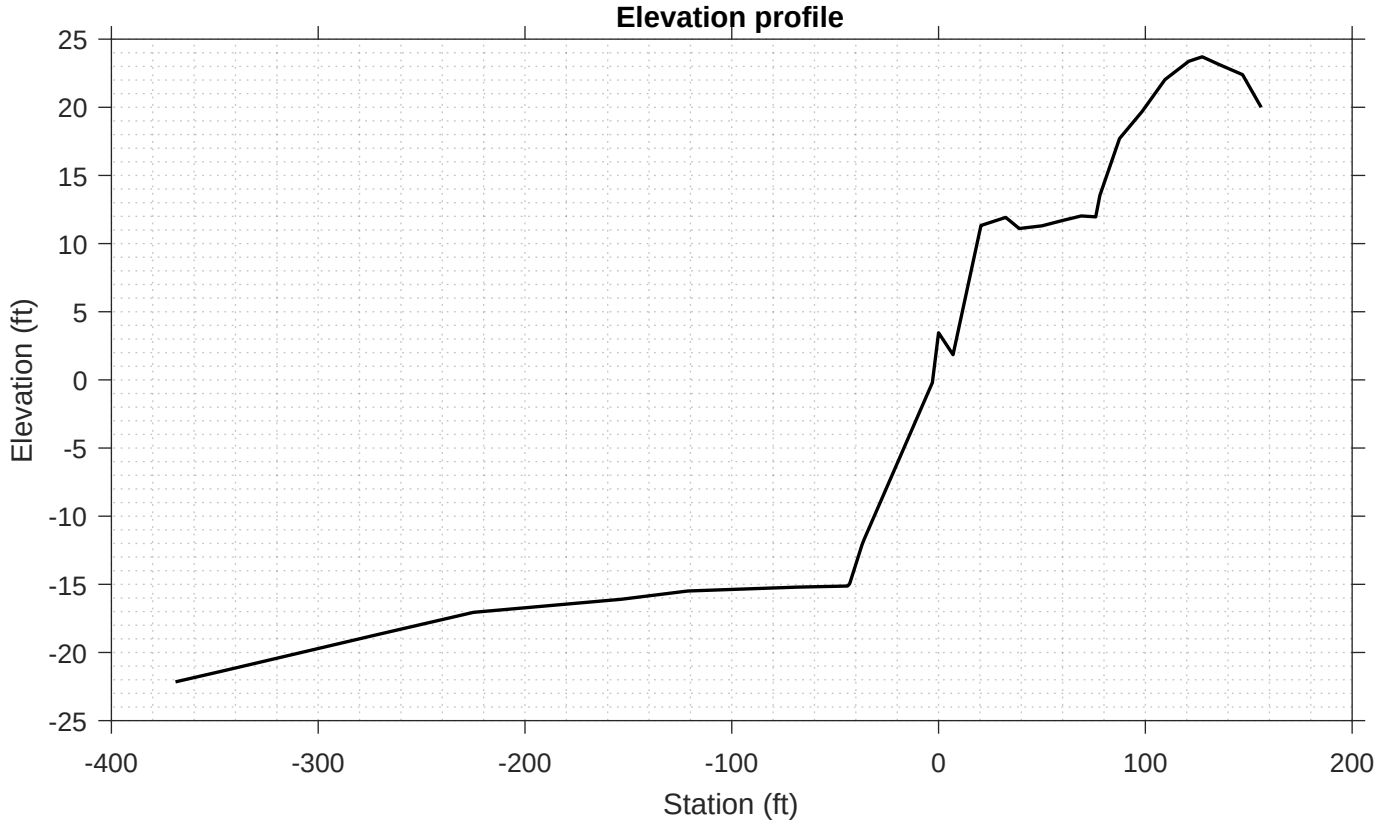

Transect Number: CM-158-1
Cundy's Harbor
Sebasco
43.805
43.8
43.795
43.79
43.785
43.78
43.775
43.77
Latitude
-69.92
-69.91
-69.9
-69.89
-69.88
-69.87
Longitude


Elevation profile
25
20
15
10
5
0
-5
-10
-15
-20
-25
Elevation (ft)
-400
-300
-200
-100
0
100
200
Station (ft)

DATA LOG FOR TRANSECT ID: CM-158-1

PART 1: USER INPUT

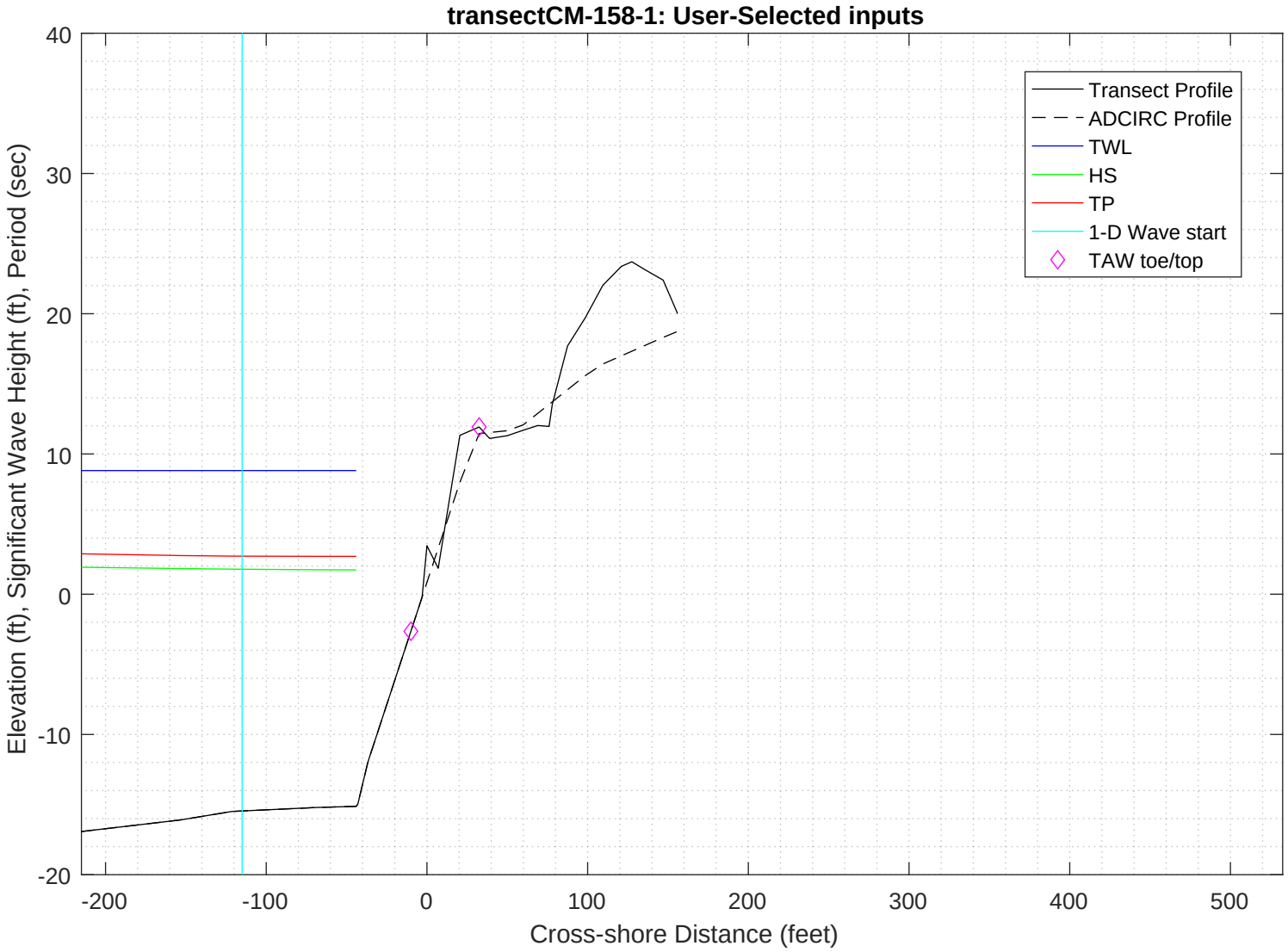
SWAN 1-D / WHAFIS input

station: -115 ft
LON: -69.8922 deg E
LAT: 43.7887 deg N
Bottom ELEV: -15.4563 ft-NAVD88
TWL: 8.816 ft-NAVD88
HS: 1.7814 ft
TP: 2.713 sec
Wave Direction bin: 180 deg CCW from East (90 deg sector)
Transect Direction: 183.6576 deg CCW from East

TAW/RUNUP input

toe sta: -10 ft
toe elev: -2.6436 ft-NAVD88
top sta: 32.5 ft
top elev: 11.9226 ft-NAVD88
Wave and water level conditions at toe to be calculated in SWAN 1-D

PART 1 COMPLETE


transectCM-158-1: User-Selected inputs
Elevation (ft), Significant Wave Height (ft), Period (sec)
Cross-shore Distance (feet)
40
30
20
10
0
-10
-20
-200
-100
0
100
200
300
400
500
Transect Profile
ADCIRC Profile
TWL
HS
TP
1-D Wave start
TAW toe/top

_____
PART 2: SWAN 1-D

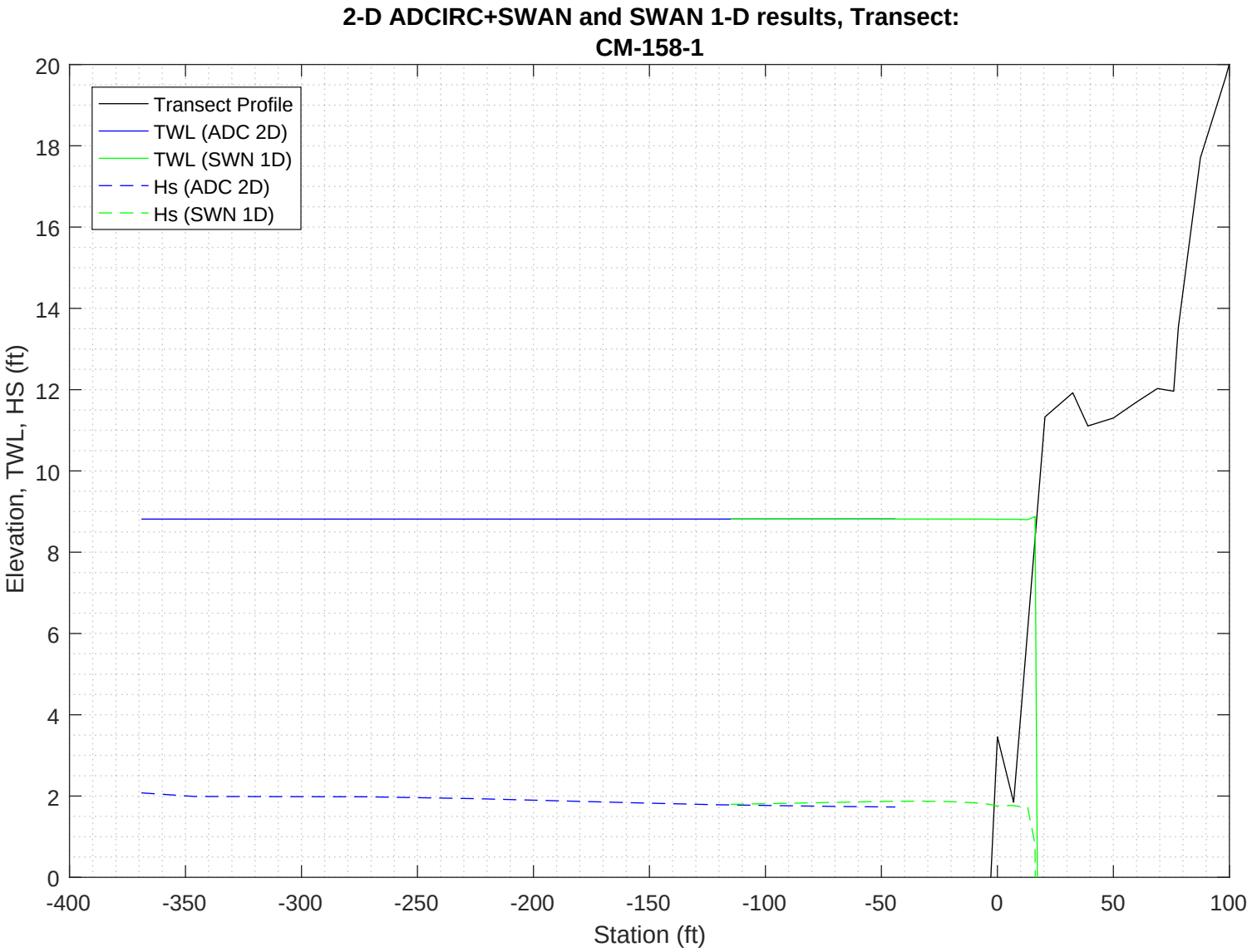
swan input grid name: 2_swan/gridfiles/CM-158-1zmeters_xmeters.grd
swan file name: 2_swan/swanfiles/CM-158-1.swn
swan output name: 2_swan/swanfiles/CM-158-1.dat

Boundary Conditions:
    TWL- 2.6871 meters
    HS- 0.54298 meters
    PER- 2.713 seconds

Batch File: 2_swan/swanfiles/runswan.dat

SWAN maximum additional wave setup: 0.06101 feet
SWAN output at toe:
    SETUP- -0.0012402 feet
    HS-   1.833 feet
    PER-   2.6668 seconds

PART 2 COMPLETE_____


2-D ADCIRC+SWAN and SWAN 1-D results, Transect:
CM-158-1
Transect Profile
TWL (ADC 2D)
TWL (SWN 1D)
Hs (ADC 2D)
Hs (SWN 1D)
Elevation, TWL, HS (ft)
Station (ft)
20
18
16
14
12
10
8
6
4
2
0
-400
-350
-300
-250
-200
-150
-100
-50
0
50
100

1

```
                    Execution started at 20200226.095825


                    ---------------------------------------
                                    SWAN
                    SIMULATION OF WAVES IN NEAR SHORE AREAS
                             VERSION NUMBER 41.20A
                    ---------------------------------------


 PROJECT '2018FemaAppeal' '1'
    '100-year Wind and Wave conditions'
 ! -- SET commands ------------------------------------------------------
  SET DEPMIN=0.01 MAXMES=999 MAXERR=3 PWTAIL=4
  SET LEVEL 0
  SET CARTESIAN
 ! -- MODE commands -----------------------------------------------------
  MODE STATIONARY ONED
 !-- COORDINATES commands-----------------------------------------------
 COORDINATES CART
 !
 !  --  computational (CGRID) grid commands -------------------------------------
 !                                  xlenc=length of grid in meters
 !  mxc = number of mesh cells (one less than number of grid points)
 !CGRID REGular [xpc]  [ypc]  [alpc]  [xlenc]   [ylenc]  [mxc]  [myc] &
 !      [ CIRcle|SECtor[dir1] [dir2] ] [mdc] [flow] [fhigh] [msc]
  CGRID REGULAR   0      0       0      41       0.   41     0    &
        CIRCLE                          36    0.03    0.8     30
 Resolution in sigma-space: df/f =  0.1157
 ! -- READgrid ---- not used in 1-D mode ----------------------------------
 ! -- INPgrid commands  -------------------------------------------------------
 !INPgrid BOTtom REGular [xpinp] [ypinp] [alpinp] [mxinp] [myinp] [dxinp] [dyinp]
 !
  INPGRID BOTTOM REGULAR    0       0        0      41   0      1       1
 !READinp BOTtom [fac]  'fname1'  [idla]  [nhedf]  [FREe|FORmat[form]|UNFormatted]
  READ    BOTTOM   -1. '../gridfiles/CM-158-1zmeters_xmeters.grd'    1      0       FREE
 !-------------------------------------------------------------------------
 ! -- WIND [vel] [dir]
 WIND      25.1  0
 ! -- BOUnd SHAPespec
 BOUND SHAPE JONSWAP 3.3  PEAK DSPR POWER
 ! -- BOUndspec
 ! BOU SIDE W CCW CON FILE 'swanspec.txt' 1
 BOUN SIDE W CCW CONSTANT PAR   0.54298     2.713     0  2
 !-- BOUndnest1  - optional for boundary from parent run
 !-- BOUndnest2
 !-- BOUndnest3
 !-- INITial -- usest to specify initial values
 !
```

```
!---------------- P H Y S I C S ------------------------
!-- GEN1 [cf10] [cf20] [cf30] [cf40] [edm1pm] [cdrag] [umin] [cfpm]
!-- GEN2 [cf10] [cf20] [cf30] [cf40] [cf50] [cf60] [edm1pm] [cdrag] [umin] [cfpm]
    GEN3 KOMEN
!   whitecapping ( on by default)
!-- WCAPping KOMen [cds2] [stpm] [powst] [delta] [powk]
    WCAP KOM
!   quadruplet wave interactions
!-- QUADrupl [iquad] [lambda] [Cn14] [Csh1] [Csh2]
! -- BREaking CONstant [alpha] [gamma]
     BREAK    CON       1.      0.73
!-- FRICtion JONswap CONstant [cfjon]
    FRIC     JONSWAP CON       0.038
!-- TRIad [itriad] [trfac] [cutfr]   [a]  [b]  [urcrit] [urslim]
! TRIAD        1      0.65    2.5    0.95 -0.75  0.2       0.01
  TRIAD
!-- VEGEtation [height] [diamtr] [nstems] [drag]
!-- MUD [layer] [rhom] [viscm]
!- LIMiter [ursell] [qb] deactivates quadruplets with Ursell number exceeds ursell
!-- OBSTacle -- not in 1-D
!-- SETUP [supcor]
   SETUP    0
!
! ----------------- N U M E R I C S -----------------------------
!
!-- PROP  can use BBST or GSE instead of default
! -- NUMeric -- lots of options
!     NUM ACCUR npnts=100. stat 30
     NUMeric STOPC
!
! -------------------O U T P U T ------------------------------
!
!OUTPut OPTIons "comment' (TABLE [field]) (BLOck [ndec] [len]) (SPEC [ndec])
 OUTPUT OPTIONS '%' TABLE 16
 $BLOCK 9 1000 SPEC 8
!CURve 'sname' [xp1] [yp1] <[int]   [xp]   [yp] >
 CURVE 'curve' 0     0      41  41    0
!TABLe 'sname' < HEADer|NOHEADer|INDexed > 'fname' <output parameters> (output time)
 Table 'curve'   HEADER 'CM-158-1.dat' XP YP HSIGN TPS RTP TMM10 DIR &
 DSPR DEPTH SETUP
!QUANTITY XP hexp=99999
!
!---------------------------------------------
COMPUTE STATIONARY
-----------------------------------------------------------------
                 COMPUTATIONAL PART OF SWAN
-----------------------------------------------------------------
```

```
One-dimensional mode of SWAN is activated
Gridresolution       : MXC              42 MYC              1
                     : MCGRD            43
                     : MSC              31 MDC             36
                     : MTC               1
                     : NSTATC            0 ITERMX          50
Propagation flags    : ITFRE             1 IREFR            1
Source term flags    : IBOT              1 ISURF            1
                     : IWCAP             1 IWIND            3
                     : ITRIAD            1 IQUAD            2
                     : IVEG              0 ITURBV           0
                     : IMUD              0
Spatial step         : DX       0.1000E+01 DY      0.1000E+01
Spectral bin         : df/f     0.1157E+00 DDIR    0.1000E+02
Physical constants   : GRAV     0.9810E+01 RHO     0.1025E+04
Wind input           : WSPEED   0.2510E+02 DIR     0.0000E+00
Tail parameters      : E(f)     0.4000E+01 E(k)    0.2500E+01
                     : A(f)     0.5000E+01 A(k)    0.3000E+01
Accuracy parameters  : DREL     0.1000E-01 NPNTS   0.9950E+02
                     : DHABS    0.0000E+00 CURVAT  0.5000E-02
                     : GRWMX    0.1000E+00
Drying/flooding      : LEVEL    0.0000E+00 DEPMIN  0.1000E-01
The Cartesian convention for wind and wave directions is used
Scheme for geographic propagation is SORDUP
Scheme geogr. space  : PROPSC            2 ICMAX            7
Scheme spectral space: CSS      0.5000E+00 CDD     0.5000E+00
Current is off
Quadruplets          : IQUAD             2
                     : LAMBDA   0.2500E+00 CNL4    0.3000E+08
                     : CSH1     0.5500E+01 CSH2    0.8330E+00
                     : CSH3    -0.1250E+01
Maximum Ursell nr for Snl4 :    0.1000E+02
Triads               : ITRIAD            1 TRFAC   0.8000E+00
                     : CUTFR    0.2500E+01 URCRI   0.2000E+00
Minimum Ursell nr for Snl3 :    0.1000E-01
JONSWAP (‘73)        : GAMMA    0.3800E-01
Vegetation is off
Turbulence is off
Fluid mud is off
W-cap Komen (‘84)    : EMPCOF (CDS2):   0.2360E-04
W-cap Komen (‘84)    : APM (STPM)   :   0.3020E-02
W-cap Komen (‘84)    : POWST        :   0.2000E+01
W-cap Komen (‘84)    : DELTA        :   0.1000E+01
W-cap Komen (‘84)    : POWK         :   0.1000E+01
Wind drag is fit
Snyder/Komen wind input
Battjes&Janssen (‘78): ALPHA    0.1000E+01 GAMMA   0.7300E+00
Set-up               : SUPCOR   0.0000E+00
Diffraction is off
Janssen (‘89,‘90)    : ALPHA    0.1000E-01 KAPPA   0.4100E+00
Janssen (‘89,‘90)    : RHOA     0.1280E+01 RHOW    0.1025E+04

1st and 2nd gen. wind: CF10     0.1880E+03 CF20    0.5900E+00
                     : CF30     0.1200E+00 CF40    0.2500E+03
                     : CF50     0.2300E-02 CF60   -0.2230E+00
                     : CF70     0.0000E+00 CF80   -0.5600E+00
                     : RHOAW    0.1249E-02 EDMLPM  0.3600E-02
                     : CDRAG    0.1230E-02 UMIN    0.1000E+01
                     : LIM_PM   0.1300E+00

 ---------------------------------------------------------------
 First guess by 2nd generation model flags for first iteration:
 ITER       1 GRWMX      0.1000E+23 ALFA       0.0000E+00
 IWIND      2 IWCAP      0 IQUAD      0
 ITRIAD     1 IBOT       1 ISURF      1
 IVEG       0 ITURBV     0 IMUD       0
 ---------------------------------------------------------------
iteration    1; sweep 1
iteration    1; sweep 2
iteration    1; sweep 3
iteration    1; sweep 4
not possible to compute, first iteration

 ---------------------------------------------------------------
 Options given by user are activated for proceeding calculation:
 ITER       2 GRWMX      0.1000E+00 ALFA       0.0000E+00
 IWIND      3 IWCAP      1 IQUAD      2
 ITRIAD     1 IBOT       1 ISURF      1
 IVEG       0 ITURBV     0 IMUD       0
 ---------------------------------------------------------------
iteration    2; sweep 1
iteration    2; sweep 2
iteration    2; sweep 3
iteration    2; sweep 4
accuracy OK in   9.76 % of wet grid points ( 99.50 % required)

iteration    3; sweep 1
iteration    3; sweep 2
iteration    3; sweep 3
```

```
iteration    3; sweep 4
accuracy OK in   2.44 % of wet grid points ( 99.50 % required)

iteration    4; sweep 1
iteration    4; sweep 2
iteration    4; sweep 3
iteration    4; sweep 4
accuracy OK in  12.20 % of wet grid points ( 99.50 % required)

iteration    5; sweep 1
iteration    5; sweep 2
iteration    5; sweep 3
iteration    5; sweep 4
accuracy OK in  85.37 % of wet grid points ( 99.50 % required)

iteration    6; sweep 1
iteration    6; sweep 2
iteration    6; sweep 3
iteration    6; sweep 4
accuracy OK in  97.57 % of wet grid points ( 99.50 % required)

iteration    7; sweep 1
iteration    7; sweep 2
iteration    7; sweep 3
iteration    7; sweep 4
accuracy OK in  97.57 % of wet grid points ( 99.50 % required)

iteration    8; sweep 1
iteration    8; sweep 2
iteration    8; sweep 3
iteration    8; sweep 4
accuracy OK in  97.57 % of wet grid points ( 99.50 % required)

iteration    9; sweep 1
iteration    9; sweep 2
iteration    9; sweep 3
iteration    9; sweep 4
accuracy OK in  97.57 % of wet grid points ( 99.50 % required)

iteration   10; sweep 1
iteration   10; sweep 2
iteration   10; sweep 3
iteration   10; sweep 4
accuracy OK in  97.57 % of wet grid points ( 99.50 % required)

iteration   11; sweep 1
iteration   11; sweep 2
iteration   11; sweep 3
iteration   11; sweep 4
accuracy OK in 100.00 % of wet grid points ( 99.50 % required)


STOP
```

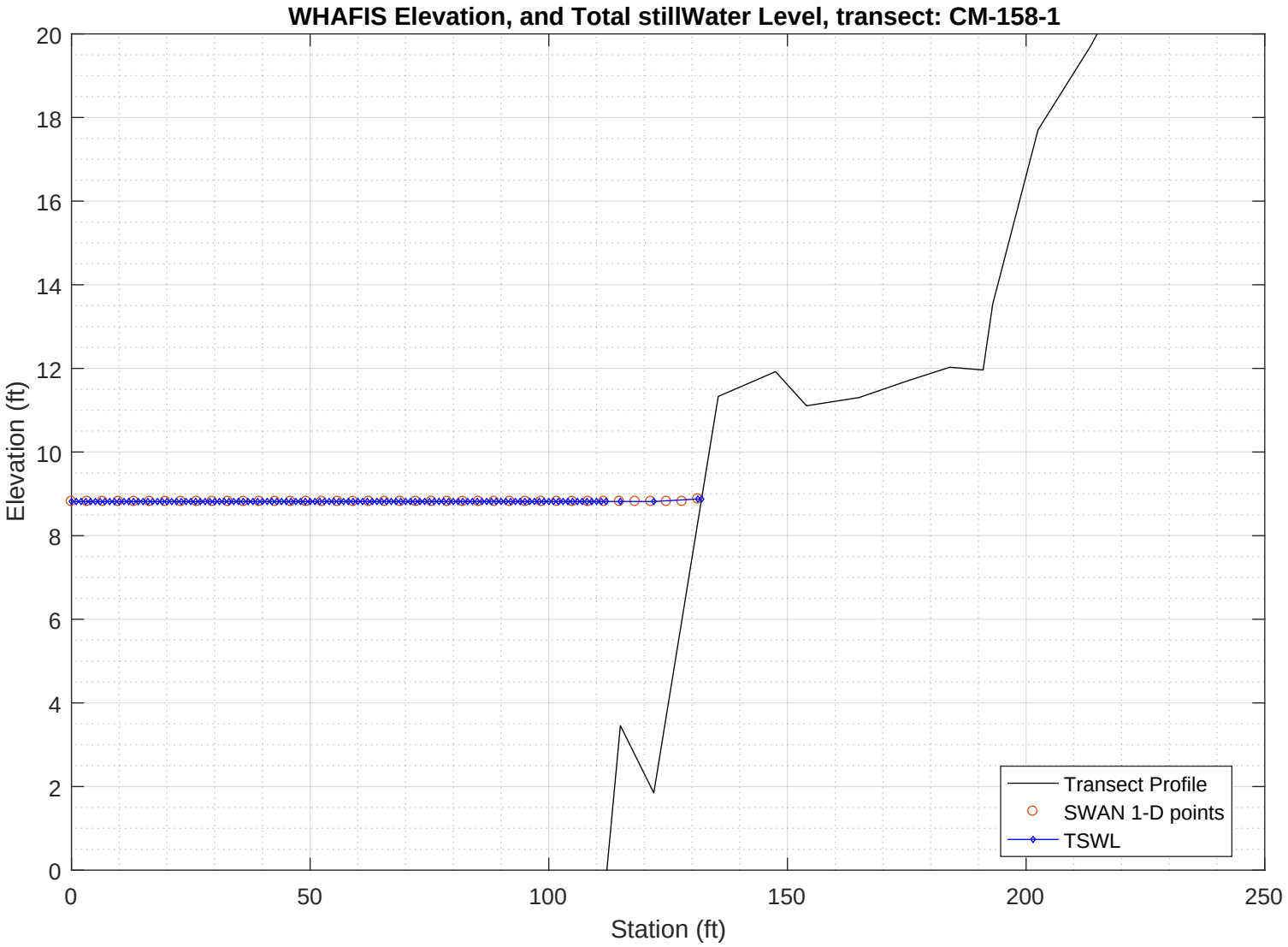
% Run:1 Table:curve SWAN version:41.20A

Xp [m]	Yp [m]	Hsig [m]	TPsmoo [sec]	RTpeak [sec]	Tm_10 [sec]	Dir [degr]	Dspr [degr]	Depth [m]	Setup [m]
0.	0.	0.54730	2.6744	2.6893	2.4321	0.000	31.5245	7.4000	0.000000
1.	0.	0.54837	2.6743	2.6893	2.4298	0.000	31.5427	7.3900	-0.000003
2.	0.	0.54945	2.6742	2.6893	2.4276	0.000	31.5611	7.3900	-0.000006
3.	0.	0.55053	2.6742	2.6893	2.4253	0.000	31.5797	7.3800	-0.000009
4.	0.	0.55161	2.6741	2.6893	2.4230	0.000	31.5986	7.3800	-0.000011
5.	0.	0.55269	2.6740	2.6893	2.4208	0.000	31.6179	7.3700	-0.000014
6.	0.	0.55377	2.6740	2.6893	2.4185	0.000	31.6373	7.3700	-0.000017
7.	0.	0.55485	2.6739	2.6893	2.4163	0.000	31.6571	7.3600	-0.000020
8.	0.	0.55594	2.6739	2.6893	2.4140	0.000	31.6772	7.3600	-0.000023
9.	0.	0.55701	2.6738	2.6893	2.4118	0.000	31.6972	7.3500	-0.000026
10.	0.	0.55809	2.6737	2.6893	2.4096	0.000	31.7177	7.3400	-0.000029
11.	0.	0.55917	2.6737	2.6893	2.4074	0.000	31.7386	7.3400	-0.000032
12.	0.	0.56025	2.6736	2.6893	2.4052	0.000	31.7597	7.3300	-0.000035
13.	0.	0.56133	2.6735	2.6893	2.4031	0.000	31.7811	7.3300	-0.000037
14.	0.	0.56241	2.6735	2.6893	2.4010	0.000	31.8028	7.3200	-0.000040
15.	0.	0.56349	2.6734	2.6893	2.3989	0.000	31.8252	7.3200	-0.000043
16.	0.	0.56457	2.6734	2.6893	2.3968	0.001	31.8476	7.3200	-0.000046
17.	0.	0.56564	2.6733	2.6893	2.3947	0.001	31.8701	7.3100	-0.000049
18.	0.	0.56672	2.6733	2.6893	2.3927	0.001	31.8932	7.3099	-0.000052
19.	0.	0.56779	2.6732	2.6893	2.3907	0.002	31.9161	7.3099	-0.000055
20.	0.	0.56885	2.6731	2.6893	2.3887	0.002	31.9392	7.2999	-0.000058
21.	0.	0.56992	2.6731	2.6893	2.3867	0.002	31.9598	7.2999	-0.000061
22.	0.	0.57083	2.6730	2.6893	2.3844	0.002	31.9631	7.2199	-0.000065
23.	0.	0.57106	2.6725	2.6893	2.3806	0.003	31.9440	6.7399	-0.000073
24.	0.	0.57106	2.6720	2.6893	2.3764	0.003	31.9146	6.2899	-0.000084
25.	0.	0.57099	2.6715	2.6893	2.3723	0.003	31.8779	5.9299	-0.000097
26.	0.	0.57069	2.6709	2.6893	2.3679	0.004	31.8311	5.5899	-0.000112
27.	0.	0.57001	2.6702	2.6893	2.3630	0.004	31.7654	5.2399	-0.000131
28.	0.	0.56891	2.6694	2.6893	2.3576	0.004	31.6802	4.8898	-0.000157
29.	0.	0.56729	2.6686	2.6893	2.3516	0.005	31.5671	4.5398	-0.000190
30.	0.	0.56509	2.6679	2.6893	2.3451	0.005	31.4110	4.1898	-0.000235
31.	0.	0.56223	2.6672	2.6893	2.3382	0.006	31.2057	3.8397	-0.000295
32.	0.	0.55869	2.6668	2.6893	2.3309	0.006	30.9301	3.4896	-0.000378
33.	0.	0.55461	2.6668	2.6893	2.3236	0.006	30.5694	3.1495	-0.000488
34.	0.	0.54775	2.6672	2.6893	2.3149	0.007	29.3745	2.7993	-0.000656
35.	0.	0.53502	2.6718	2.6893	2.3165	0.008	28.2936	1.6983	-0.001744
36.	0.	0.53731	2.6708	2.6893	2.3119	0.008	28.6332	1.8485	-0.001493
37.	0.	0.53880	2.6696	2.6893	2.3064	0.008	28.2307	2.0788	-0.001202
38.	0.	0.53106	2.6729	2.6893	2.3156	0.008	25.9581	1.5479	-0.002143
39.	0.	0.53101	2.6783	2.6893	2.2609	0.149	20.2429	0.8451	-0.004915
40.	0.	0.24846	2.7245	2.6893	2.1823	353.208	22.9392	0.1686	0.018596
41.	0.	-9.00000	-9.0000	-9.0000	-9.0000	-999.000	-9.0000	-99.0000	-9.000000

PART 3: WHAFIS

WHAFIS input: CM-158-1.dat
WHAFIS output: CM-158-1.out

PART 3 COMPLETE


WHAFIS Elevation, and Total stillWater Level, transect: CM-158-1
Elevation (ft)
Station (ft)
0
2
4
6
8
10
12
14
16
18
20
0
50
100
150
200
250
Transect Profile
SWAN 1-D points
TSWL

WAVE HEIGHT COMPUTATIONS FOR FLOOD INSURANCE STUDIES (WHAFIS VERSION 4.0G, 08_2007)
Executed on: Wed Feb 26 13:21:30 2020
Input file: C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswell\Additional_Transects\3_whafis\whafis4\CM-158-1.dat
Output file: C:\FEMA-TransectAnalysis\LOMR-TransectAnalysis-Harpswell\Additional_Transects\3_whafis\whafis4\CM-158-1.out

header

THIS IS A 100-YEAR CASE
THE FOLLOWING NON-DEFAULT WIND SPEEDS ARE BEING USED
WINDIF 56.14 WINDOF 56.14 WINDVH 60.00

PART1 INPUT

IE	0.000	-15.456	1.000	1.000	8.816	2.850	2.713	56.140	0.006	0.000
OF	1.000	-15.450	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	2.000	-15.445	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	3.000	-15.440	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	4.000	-15.434	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	5.000	-15.429	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	6.000	-15.424	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	7.000	-15.418	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	8.000	-15.413	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	9.000	-15.407	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	10.000	-15.402	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	11.000	-15.397	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	12.000	-15.391	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	13.000	-15.386	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	14.000	-15.380	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	15.000	-15.375	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	16.000	-15.370	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	17.000	-15.364	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	18.000	-15.359	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	19.000	-15.354	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	20.000	-15.348	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	21.000	-15.343	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	22.000	-15.337	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	23.000	-15.332	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	24.000	-15.327	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	25.000	-15.321	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	26.000	-15.316	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	27.000	-15.310	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	28.000	-15.305	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	29.000	-15.300	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	30.000	-15.294	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
OF	31.000	-15.289	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
OF	32.000	-15.284	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
OF	33.000	-15.278	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
OF	34.000	-15.273	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
OF	35.000	-15.267	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
OF	36.000	-15.262	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
OF	37.000	-15.257	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
OF	38.000	-15.251	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
OF	39.000	-15.246	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
OF	40.000	-15.241	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
OF	41.000	-15.235	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
OF	42.000	-15.230	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
OF	43.000	-15.224	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
OF	44.000	-15.219	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
OF	45.000	-15.214	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
OF	46.000	-15.208	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
OF	47.000	-15.204	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
OF	48.000	-15.200	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
OF	49.000	-15.197	0.000	8.817	0.000	0.000	0.000	0.000	0.003	0.000
OF	50.000	-15.194	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
OF	51.000	-15.190	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
OF	52.000	-15.187	0.000	8.817	0.000	0.000	0.000	0.000	0.003	0.000
OF	53.000	-15.184	0.000	8.817	0.000	0.000	0.000	0.000	0.003	0.000
OF	54.000	-15.181	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
OF	55.000	-15.177	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
OF	56.000	-15.174	0.000	8.817	0.000	0.000	0.000	0.000	0.003	0.000
OF	57.000	-15.171	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
OF	58.000	-15.167	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
OF	59.000	-15.164	0.000	8.817	0.000	0.000	0.000	0.000	0.003	0.000
OF	60.000	-15.161	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
OF	61.000	-15.157	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
OF	62.000	-15.154	0.000	8.817	0.000	0.000	0.000	0.000	0.003	0.000
OF	63.000	-15.151	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
OF	64.000	-15.147	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
OF	65.000	-15.144	0.000	8.817	0.000	0.000	0.000	0.000	0.003	0.000
OF	66.000	-15.141	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
OF	67.000	-15.137	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
OF	68.000	-15.134	0.000	8.817	0.000	0.000	0.000	0.000	0.003	0.000
OF	69.000	-15.131	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
OF	70.000	-15.127	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
OF	71.000	-15.124	0.000	8.817	0.000	0.000	0.000	0.000	0.083	0.000
OF	72.000	-14.960	0.000	8.817	0.000	0.000	0.000	0.000	0.321	0.000
OF	73.000	-14.482	0.000	8.817	0.000	0.000	0.000	0.000	0.479	0.000
OF	74.000	-14.003	0.000	8.817	0.000	0.000	0.000	0.000	0.479	0.000
OF	75.000	-13.525	0.000	8.817	0.000	0.000	0.000	0.000	0.478	0.000
OF	76.000	-13.047	0.000	8.817	0.000	0.000	0.000	0.000	0.479	0.000
OF	77.000	-12.568	0.000	8.817	0.000	0.000	0.000	0.000	0.479	0.000
OF	78.000	-12.090	0.000	8.817	0.000	0.000	0.000	0.000	0.431	0.000
OF	79.000	-11.705	0.000	8.817	0.000	0.000	0.000	0.000	0.368	0.000
OF	80.000	-11.355	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
OF	81.000	-11.007	0.000	8.817	0.000	0.000	0.000	0.000	0.348	0.000
OF	82.000	-10.659	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
OF	83.000	-10.310	0.000	8.817	0.000	0.000	0.000	0.000	0.348	0.000
OF	84.000	-9.962	0.000	8.817	0.000	0.000	0.000	0.000	0.348	0.000
OF	85.000	-9.614	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
OF	86.000	-9.265	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
OF	87.000	-8.917	0.000	8.817	0.000	0.000	0.000	0.000	0.348	0.000
OF	88.000	-8.568	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
OF	89.000	-8.219	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
OF	90.000	-7.871	0.000	8.817	0.000	0.000	0.000	0.000	0.348	0.000
OF	91.000	-7.523	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
OF	92.000	-7.174	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000

OF	93.000	-6.826	0.000	8.817	0.000	0.000	0.000	0.000	0.348	0.000
OF	94.000	-6.477	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
OF	95.000	-6.128	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
OF	96.000	-5.780	0.000	8.817	0.000	0.000	0.000	0.000	0.348	0.000
OF	97.000	-5.432	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
OF	98.000	-5.083	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
OF	99.000	-4.734	0.000	8.817	0.000	0.000	0.000	0.000	0.348	0.000
OF	100.000	-4.386	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
OF	101.000	-4.037	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
OF	102.000	-3.689	0.000	8.817	0.000	0.000	0.000	0.000	0.348	0.000
OF	103.000	-3.341	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
OF	104.000	-2.992	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
OF	105.000	-2.644	0.000	8.817	0.000	0.000	0.000	0.000	0.348	0.000
OF	106.000	-2.295	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
OF	107.000	-1.946	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
OF	108.000	-1.598	0.000	8.817	0.000	0.000	0.000	0.000	0.348	0.000
OF	109.000	-1.250	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
OF	110.000	-0.901	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
OF	111.000	-0.553	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
OF	112.000	-0.203	0.000	8.817	0.000	0.000	0.000	0.000	1.003	0.000
IF	115.000	3.458	0.000	8.817	0.000	0.000	0.000	0.000	0.205	0.000
IF	122.000	1.847	0.000	8.817	0.000	0.000	0.000	0.000	0.301	0.000
IF	131.200	8.332	0.000	8.877	0.000	0.000	0.000	0.000	0.703	0.000
IF	132.000	8.877	0.000	8.877	0.000	0.000	0.000	0.000	0.681	0.000
ET	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

1

	END STATION	END ELEVATION	FETCH LENGTH	SURGE ELEV 10-YEAR	SURGE ELEV 100-YEAR	INITIAL WAVE HEIGHT	INITIAL W. PERIOD		BOTTOM SLOPE	AVERAGE A-ZONES
IE	0.000	-15.456	1.000	1.000	8.816	2.850	2.713	56.140	0.006	0.000
	END STATION	END ELEVATION	NEW SURGE 10-YEAR	NEW SURGE 100-YEAR					BOTTOM SLOPE	AVERAGE A-ZONES
OF	1.000	-15.450	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END STATION	END ELEVATION	NEW SURGE 10-YEAR	NEW SURGE 100-YEAR					BOTTOM SLOPE	AVERAGE A-ZONES
OF	2.000	-15.445	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END STATION	END ELEVATION	NEW SURGE 10-YEAR	NEW SURGE 100-YEAR					BOTTOM SLOPE	AVERAGE A-ZONES
OF	3.000	-15.440	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END STATION	END ELEVATION	NEW SURGE 10-YEAR	NEW SURGE 100-YEAR					BOTTOM SLOPE	AVERAGE A-ZONES
OF	4.000	-15.434	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END STATION	END ELEVATION	NEW SURGE 10-YEAR	NEW SURGE 100-YEAR					BOTTOM SLOPE	AVERAGE A-ZONES
OF	5.000	-15.429	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END STATION	END ELEVATION	NEW SURGE 10-YEAR	NEW SURGE 100-YEAR					BOTTOM SLOPE	AVERAGE A-ZONES
OF	6.000	-15.424	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END STATION	END ELEVATION	NEW SURGE 10-YEAR	NEW SURGE 100-YEAR					BOTTOM SLOPE	AVERAGE A-ZONES
OF	7.000	-15.418	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END STATION	END ELEVATION	NEW SURGE 10-YEAR	NEW SURGE 100-YEAR					BOTTOM SLOPE	AVERAGE A-ZONES
OF	8.000	-15.413	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END STATION	END ELEVATION	NEW SURGE 10-YEAR	NEW SURGE 100-YEAR					BOTTOM SLOPE	AVERAGE A-ZONES
OF	9.000	-15.407	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END STATION	END ELEVATION	NEW SURGE 10-YEAR	NEW SURGE 100-YEAR					BOTTOM SLOPE	AVERAGE A-ZONES
OF	10.000	-15.402	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END STATION	END ELEVATION	NEW SURGE 10-YEAR	NEW SURGE 100-YEAR					BOTTOM SLOPE	AVERAGE A-ZONES
OF	11.000	-15.397	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END STATION	END ELEVATION	NEW SURGE 10-YEAR	NEW SURGE 100-YEAR					BOTTOM SLOPE	AVERAGE A-ZONES
OF	12.000	-15.391	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END STATION	END ELEVATION	NEW SURGE 10-YEAR	NEW SURGE 100-YEAR					BOTTOM SLOPE	AVERAGE A-ZONES
OF	13.000	-15.386	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END STATION	END ELEVATION	NEW SURGE 10-YEAR	NEW SURGE 100-YEAR					BOTTOM SLOPE	AVERAGE A-ZONES
OF	14.000	-15.380	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END STATION	END ELEVATION	NEW SURGE 10-YEAR	NEW SURGE 100-YEAR					BOTTOM SLOPE	AVERAGE A-ZONES
OF	15.000	-15.375	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END STATION	END ELEVATION	NEW SURGE 10-YEAR	NEW SURGE 100-YEAR					BOTTOM SLOPE	AVERAGE A-ZONES
OF	16.000	-15.370	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END STATION	END ELEVATION	NEW SURGE 10-YEAR	NEW SURGE 100-YEAR					BOTTOM SLOPE	AVERAGE A-ZONES
OF	17.000	-15.364	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END STATION	END ELEVATION	NEW SURGE 10-YEAR	NEW SURGE 100-YEAR					BOTTOM SLOPE	AVERAGE A-ZONES
OF	18.000	-15.359	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END STATION	END ELEVATION	NEW SURGE 10-YEAR	NEW SURGE 100-YEAR					BOTTOM SLOPE	AVERAGE A-ZONES
OF	19.000	-15.354	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END STATION	END ELEVATION	NEW SURGE 10-YEAR	NEW SURGE 100-YEAR					BOTTOM SLOPE	AVERAGE A-ZONES
OF	20.000	-15.348	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END STATION	END ELEVATION	NEW SURGE 10-YEAR	NEW SURGE 100-YEAR					BOTTOM SLOPE	AVERAGE A-ZONES
OF	21.000	-15.343	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END STATION	END ELEVATION	NEW SURGE 10-YEAR	NEW SURGE 100-YEAR					BOTTOM SLOPE	AVERAGE A-ZONES
OF	22.000	-15.337	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END STATION	END ELEVATION	NEW SURGE 10-YEAR	NEW SURGE 100-YEAR					BOTTOM SLOPE	AVERAGE A-ZONES
OF	23.000	-15.332	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END STATION	END ELEVATION	NEW SURGE 10-YEAR	NEW SURGE 100-YEAR					BOTTOM SLOPE	AVERAGE A-ZONES
OF	24.000	-15.327	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE

	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	25.000	-15.321	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	26.000	-15.316	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	27.000	-15.310	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	28.000	-15.305	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	29.000	-15.300	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	30.000	-15.294	0.000	8.816	0.000	0.000	0.000	0.000	0.005	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	31.000	-15.289	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	32.000	-15.284	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	33.000	-15.278	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	34.000	-15.273	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	35.000	-15.267	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	36.000	-15.262	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	37.000	-15.257	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	38.000	-15.251	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	39.000	-15.246	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	40.000	-15.241	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	41.000	-15.235	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	42.000	-15.230	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	43.000	-15.224	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	44.000	-15.219	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	45.000	-15.214	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	46.000	-15.208	0.000	8.817	0.000	0.000	0.000	0.000	0.005	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	47.000	-15.204	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	48.000	-15.200	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	49.000	-15.197	0.000	8.817	0.000	0.000	0.000	0.000	0.003	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	50.000	-15.194	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	51.000	-15.190	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	52.000	-15.187	0.000	8.817	0.000	0.000	0.000	0.000	0.003	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	53.000	-15.184	0.000	8.817	0.000	0.000	0.000	0.000	0.003	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	54.000	-15.181	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	55.000	-15.177	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	56.000	-15.174	0.000	8.817	0.000	0.000	0.000	0.000	0.003	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	57.000	-15.171	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	58.000	-15.167	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE

	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	59.000	-15.164	0.000	8.817	0.000	0.000	0.000	0.000	0.003	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	60.000	-15.161	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	61.000	-15.157	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	62.000	-15.154	0.000	8.817	0.000	0.000	0.000	0.000	0.003	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	63.000	-15.151	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	64.000	-15.147	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	65.000	-15.144	0.000	8.817	0.000	0.000	0.000	0.000	0.003	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	66.000	-15.141	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	67.000	-15.137	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	68.000	-15.134	0.000	8.817	0.000	0.000	0.000	0.000	0.003	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	69.000	-15.131	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	70.000	-15.127	0.000	8.817	0.000	0.000	0.000	0.000	0.004	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	71.000	-15.124	0.000	8.817	0.000	0.000	0.000	0.000	0.083	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	72.000	-14.960	0.000	8.817	0.000	0.000	0.000	0.000	0.321	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	73.000	-14.482	0.000	8.817	0.000	0.000	0.000	0.000	0.479	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	74.000	-14.003	0.000	8.817	0.000	0.000	0.000	0.000	0.479	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	75.000	-13.525	0.000	8.817	0.000	0.000	0.000	0.000	0.478	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	76.000	-13.047	0.000	8.817	0.000	0.000	0.000	0.000	0.479	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	77.000	-12.568	0.000	8.817	0.000	0.000	0.000	0.000	0.479	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	78.000	-12.090	0.000	8.817	0.000	0.000	0.000	0.000	0.431	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	79.000	-11.705	0.000	8.817	0.000	0.000	0.000	0.000	0.368	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	80.000	-11.355	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	81.000	-11.007	0.000	8.817	0.000	0.000	0.000	0.000	0.348	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	82.000	-10.659	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	83.000	-10.310	0.000	8.817	0.000	0.000	0.000	0.000	0.348	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	84.000	-9.962	0.000	8.817	0.000	0.000	0.000	0.000	0.348	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	85.000	-9.614	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	86.000	-9.265	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	87.000	-8.917	0.000	8.817	0.000	0.000	0.000	0.000	0.348	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	88.000	-8.568	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	89.000	-8.219	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	90.000	-7.871	0.000	8.817	0.000	0.000	0.000	0.000	0.348	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	91.000	-7.523	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	92.000	-7.174	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE

	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	93.000	-6.826	0.000	8.817	0.000	0.000	0.000	0.000	0.348	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	94.000	-6.477	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	95.000	-6.128	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	96.000	-5.780	0.000	8.817	0.000	0.000	0.000	0.000	0.348	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	97.000	-5.432	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	98.000	-5.083	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	99.000	-4.734	0.000	8.817	0.000	0.000	0.000	0.000	0.348	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	100.000	-4.386	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	101.000	-4.037	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	102.000	-3.689	0.000	8.817	0.000	0.000	0.000	0.000	0.348	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	103.000	-3.341	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	104.000	-2.992	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	105.000	-2.644	0.000	8.817	0.000	0.000	0.000	0.000	0.348	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	106.000	-2.295	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	107.000	-1.946	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	108.000	-1.598	0.000	8.817	0.000	0.000	0.000	0.000	0.348	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	109.000	-1.250	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	110.000	-0.901	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	111.000	-0.553	0.000	8.817	0.000	0.000	0.000	0.000	0.349	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
OF	112.000	-0.203	0.000	8.817	0.000	0.000	0.000	0.000	1.003	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
IF	115.000	3.458	0.000	8.817	0.000	0.000	0.000	0.000	0.205	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
IF	122.000	1.847	0.000	8.817	0.000	0.000	0.000	0.000	0.301	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
IF	131.200	8.332	0.000	8.877	0.000	0.000	0.000	0.000	0.703	0.000
	END	END	NEW SURGE	NEW SURGE					BOTTOM	AVERAGE
	STATION	ELEVATION	10-YEAR	100-YEAR					SLOPE	A-ZONES
IF	132.000	8.877	0.000	8.877	0.000	0.000	0.000	0.000	0.681	0.000

----------END OF TRANSECT----------

NOTE:
SURGE ELEVATION INCLUDES CONTRIBUTIONS FROM ASTRONOMICAL AND STORM TIDES.

1

PART2: CONTROLLING WAVE HEIGHTS, SPECTRAL PEAK WAVE PERIOD, AND WAVE CREST ELEVATIONS

	LOCATION	CONTROLLING WAVE HEIGHT	SPECTRAL PEAK WAVE PERIOD	WAVE CREST ELEVATION
IE	0.00	2.85	2.71	10.81
OF	1.00	2.85	2.71	10.81
OF	2.00	2.85	2.71	10.81
OF	3.00	2.85	2.71	10.81
OF	4.00	2.85	2.71	10.81
OF	5.00	2.85	2.71	10.81
OF	6.00	2.85	2.71	10.81
OF	7.00	2.85	2.71	10.81
OF	8.00	2.85	2.71	10.81
OF	9.00	2.85	2.71	10.81
OF	10.00	2.86	2.71	10.81
OF	11.00	2.86	2.71	10.81
OF	12.00	2.86	2.71	10.82
OF	13.00	2.86	2.71	10.82
OF	14.00	2.86	2.71	10.82
OF	15.00	2.86	2.71	10.82
OF	16.00	2.86	2.71	10.82
OF	17.00	2.86	2.72	10.82
OF	18.00	2.86	2.72	10.82
OF	19.00	2.86	2.72	10.82
OF	20.00	2.86	2.72	10.82
OF	21.00	2.86	2.72	10.82
OF	22.00	2.86	2.72	10.82

OF	23.00	2.86	2.72	10.82
OF	24.00	2.86	2.72	10.82
OF	25.00	2.86	2.72	10.82
OF	26.00	2.86	2.72	10.82
OF	27.00	2.86	2.72	10.82
OF	28.00	2.86	2.72	10.82
OF	29.00	2.86	2.72	10.82
OF	30.00	2.87	2.72	10.82
OF	31.00	2.87	2.72	10.82
OF	32.00	2.87	2.72	10.82
OF	33.00	2.87	2.72	10.82
OF	34.00	2.87	2.72	10.82
OF	35.00	2.87	2.72	10.82
OF	36.00	2.87	2.72	10.82
OF	37.00	2.87	2.72	10.83
OF	38.00	2.87	2.72	10.83
OF	39.00	2.87	2.72	10.83
OF	40.00	2.87	2.72	10.83
OF	41.00	2.87	2.72	10.83
OF	42.00	2.87	2.72	10.83
OF	43.00	2.87	2.72	10.83
OF	44.00	2.87	2.72	10.83
OF	45.00	2.87	2.72	10.83
OF	46.00	2.87	2.72	10.83
OF	47.00	2.87	2.72	10.83
OF	48.00	2.87	2.72	10.83
OF	49.00	2.88	2.72	10.83
OF	50.00	2.88	2.72	10.83
OF	51.00	2.88	2.72	10.83
OF	52.00	2.88	2.72	10.83
OF	53.00	2.88	2.72	10.83
OF	54.00	2.88	2.72	10.83
OF	55.00	2.88	2.72	10.83
OF	56.00	2.88	2.72	10.83
OF	57.00	2.88	2.72	10.83
OF	58.00	2.88	2.72	10.83
OF	59.00	2.88	2.72	10.83
OF	60.00	2.88	2.72	10.83
OF	61.00	2.88	2.72	10.83
OF	62.00	2.88	2.72	10.83
OF	63.00	2.88	2.72	10.83
OF	64.00	2.88	2.72	10.84
OF	65.00	2.88	2.72	10.84
OF	66.00	2.88	2.72	10.84
OF	67.00	2.88	2.72	10.84
OF	68.00	2.88	2.72	10.84
OF	69.00	2.89	2.72	10.84
OF	70.00	2.89	2.72	10.84
OF	71.00	2.89	2.72	10.84
OF	72.00	2.89	2.72	10.84
OF	73.00	2.89	2.72	10.84
OF	74.00	2.89	2.72	10.84
OF	75.00	2.89	2.72	10.84
OF	76.00	2.89	2.72	10.84
OF	77.00	2.88	2.72	10.84
OF	78.00	2.88	2.72	10.84
OF	79.00	2.88	2.72	10.84
OF	80.00	2.88	2.72	10.83
OF	81.00	2.88	2.72	10.83
OF	82.00	2.88	2.72	10.83
OF	83.00	2.88	2.72	10.83
OF	84.00	2.88	2.72	10.83
OF	85.00	2.87	2.72	10.83
OF	86.00	2.87	2.72	10.83
OF	87.00	2.87	2.72	10.83
OF	88.00	2.87	2.72	10.82
OF	89.00	2.86	2.72	10.82
OF	90.00	2.86	2.72	10.82
OF	91.00	2.86	2.72	10.82
OF	92.00	2.85	2.72	10.81
OF	93.00	2.85	2.72	10.81
OF	94.00	2.84	2.72	10.81
OF	95.00	2.84	2.72	10.80
OF	96.00	2.83	2.72	10.80
OF	97.00	2.83	2.72	10.80
OF	98.00	2.82	2.72	10.79
OF	99.00	2.81	2.73	10.79
OF	100.00	2.81	2.73	10.78
OF	101.00	2.80	2.73	10.78
OF	102.00	2.79	2.73	10.77
OF	103.00	2.78	2.73	10.76
OF	104.00	2.77	2.73	10.76
OF	105.00	2.76	2.73	10.75
OF	106.00	2.75	2.73	10.74
OF	107.00	2.74	2.73	10.74
OF	108.00	2.73	2.73	10.73
OF	109.00	2.72	2.73	10.72
OF	110.00	2.71	2.73	10.71
OF	111.00	2.70	2.73	10.71
OF	112.00	2.69	2.73	10.70
IF	115.00	2.66	2.73	10.68
IF	122.00	2.65	2.73	10.67
IF	131.20	0.42	2.73	9.17
IF	132.00	0.01	2.73	8.88

PART3 LOCATION OF AREAS ABOVE 100-YEAR SURGE

NO AREAS ABOVE 100-YEAR SURGE IN THIS TRANSECT

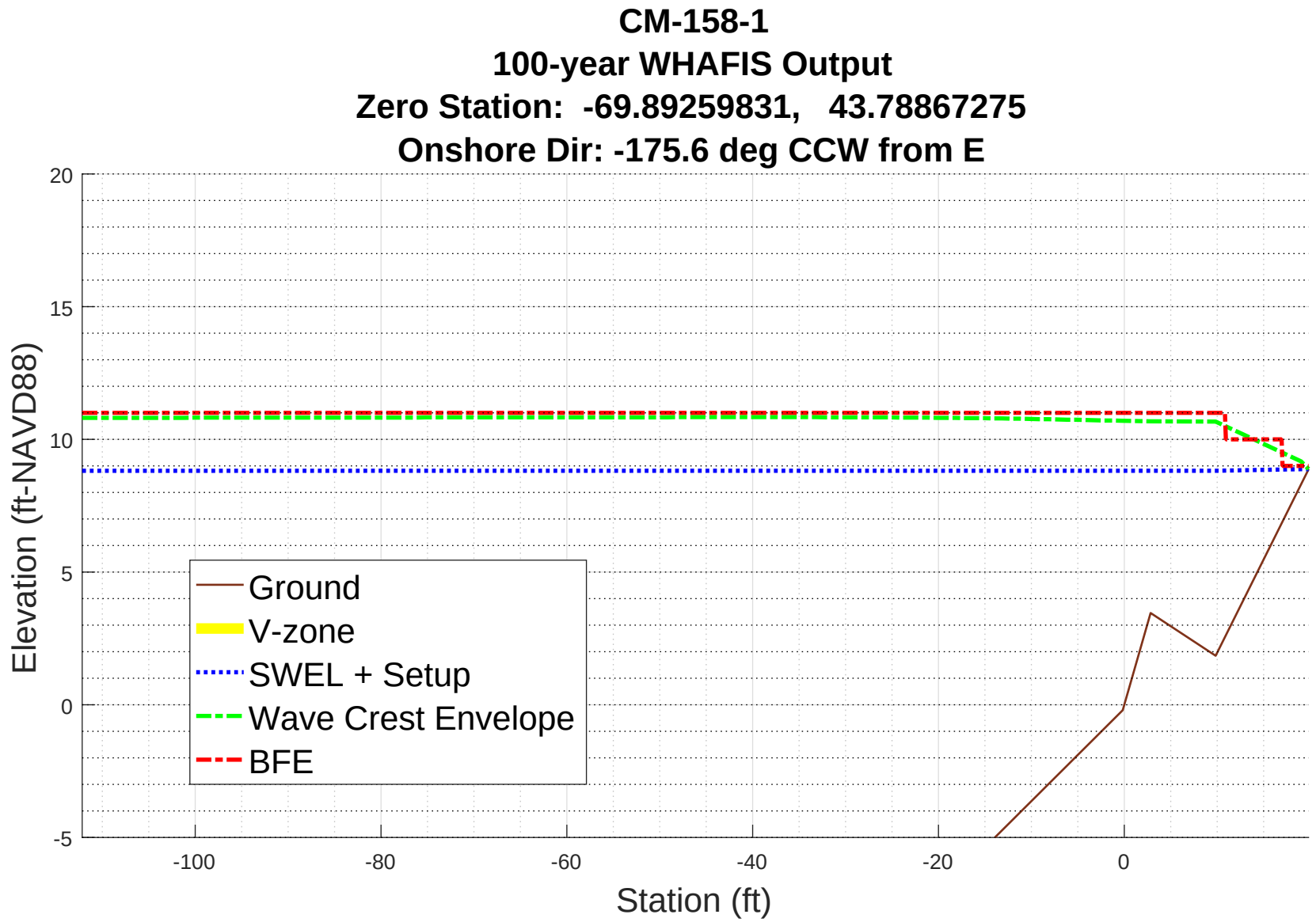
PART4 LOCATION OF SURGE CHANGES

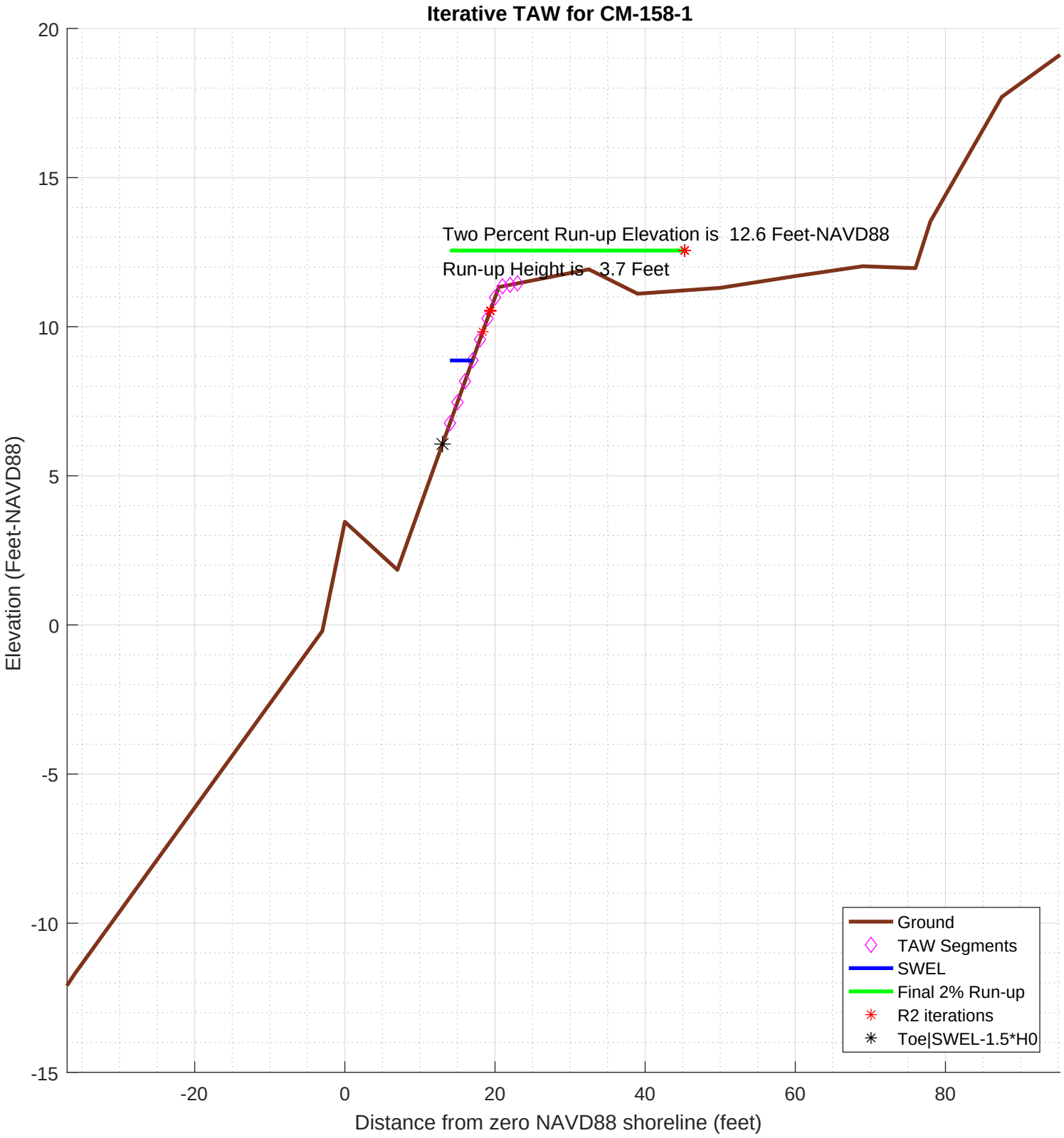
STATION	10-YEAR SURGE	100-YEAR SURGE
31.00	1.00	8.82
131.20	1.00	8.88

PART6 NUMBERED A ZONES AND V ZONES

STATION	OF	GUTTER ELEVATION	ZONE DESIGNATION	FHF

```
                                0.00          10.81
                                                                A19  EL=11      95
                               30.00          10.82
                                                                A19  EL=11      95
                               31.00          10.82
                                                                A19  EL=11      95
                              122.00          10.67
                                                                A19  EL=11      95
                              123.05          10.50
                                                                A19  EL=10      95
                              129.17           9.50
                                                                A19  EL= 9      95
                              131.20           9.17
                                                                A19  EL= 9      95
                              132.00           8.88
                                   ZONE TERMINATED AT END OF TRANSECT
                                   PART 7   POSTSCRIPT NOTES
                         PS#  1   START(428217.1638,4848791.376)
                         PS#  2   END(428137.5438,4848785.2138)
-1.000000e+00
```


CM-158-1
100-year WHAFIS Output
Zero Station: -69.89259831, 43.78867275
Onshore Dir: -175.6 deg CCW from E
Elevation (ft-NAVD88)
Station (ft)
20
15
10
5
0
-5
-100
-80
-60
-40
-20
0
Ground
V-zone
SWEL + Setup
Wave Crest Envelope
BFE


Iterative TAW for CM-158-1
Two Percent Run-up Elevation is 12.6 Feet-NAVD88
Run-up Height is 3.7 Feet
Elevation (Feet-NAVD88)
Distance from zero NAVD88 shoreline (feet)
20
15
10
5
0
-5
-10
-15
-20
0
20
40
60
80
Ground
TAW Segments
SWEL
Final 2% Run-up
R2 iterations
Toe|SWEL-1.5*H0

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

PART 5: RUNUP2

for transect: CM-158-1

Station locations shifted by: -2.83 feet from their
original location to set the shoreline to
elevation 0 for RUNUP2 input

RUNUP2 INPUT CONVERSIONS
for transect: CM-158-1

Incident significant wave height: 1.78 feet

Peak wave period: 2.71 seconds

Mean wave height: 1.12 feet

Local Depth below SWEL: 24.27 feet

Mean wave height deshoaled using Hunt approximation for
celerity assuming constant wave energy flux.
References: R.G. Dean and R.A. Dalrymple. 2000. Water

Wave Mechanics for Engineers and Scientists. World
Scientific Publishing Company, River Edge New Jersy

USACE (1985), Direct Methods for Calculating Wavelength, CETN-1-17
US Army Engineer Waterways Experiment Station Coastel Engineering
Research Center, Vicksburg, MS

also see Coastal Engineering Manual Part II-3
for discussion of shoaling coefficient

Depth, D = 24.27
Period, T = 2.31
Waveheight, H = 1.12
Deep water wavelength, L0 (ft)
L0 = g*T*T/twopi
L0 = 32.17*2.31*2.31/6.28 = 27.23
Deep water wave celerity, C0 (ft/s)
C0 = L0/T
C0 = 27.23/2.31 = 11.81
Angular frequency, sigma (rad/s)
sigma = twopi/T
sigma = 6.28/2.31 = 2.72
Hunts (1979) approximation for Celerity C1H (ft/s) at Depth D (ft)
y = sigma.*sigma.*D./g
y = 2.72*2.72*24.27/32.17 = 5.60
C1H = sqrt(g.*D./(y+1./(1 + 0.6522.*y + 0.4622.*y.^2 + 0.0864.*y.^4 +0.0675.*y.^5)))
C1H = 11.81
Shoaling Coefficient KsH
KsH = sqrt(C0/C1H)
KsH = sqrt(11.81/11.81) = 1.00
Deepwater Wave Height H0_H (ft)
H0_H = H/KsH
H0_H = 1.12/1.00 = 1.12

Deepwater mean wave height: 1.12 feet

END RUNUP2 CONVERSIONS

RUNUP2 RESULTS
for transect: CM-158-1

RUNUP2 SWEL:
8.80
8.80
8.80
8.80

8.80
8.80
8.80
8.80
8.80

RUNUP2 deepwater mean wave heights:
1.06
1.06
1.06
1.12
1.12
1.12
1.17
1.17
1.17

RUNUP2 mean wave periods:
2.19
2.31
2.42
2.19
2.31
2.42
2.19
2.31
2.42

RUNUP2 runup above SWEL:
1.21
1.23
1.25
1.26
1.29
1.31
1.31
1.33
1.35

RUNUP2 Mean runup height above SWEL: 1.28 feet

RUNUP2 2-percent runup height above SWEL: 2.82 feet

RUNUP2 2-percent runup elevation: 11.62 feet-NAVD88

RUNUP2 Messages:
No Messages

_____END RUNUP2 RESULTS_____

_____ACES BEACH RUNUP_____

Incident significant wave height: 1.78 feet

Significant wave height deshoaled using Hunt equation

Deepwater significant wave height: 1.56 feet

Peak wave period: 2.71 seconds

Average beach Slope: 1:5.39 (H:V)

ACES RUNUP CALCULATED USING 'Aces_Beach_Runup.m'

ACES Beach 2-percent runup height above SWEL: 2.72 feet

ACES Beach 2-percent runup elevation: 11.52 feet-NAVD88

ACES BEACH RUNUP is valid

_____END ACES BEACH RESULTS_____

PART 5 COMPLETE_____

```
  FEMA                                                  sjh            job 2
  RUNUP2 transect: CM-158-1                                               1
21.0
  -15.46 -112.2 0.6
  -15.21 -66.2  0.6
  -15.20 -65.2  0.6
  -15.12 -41.2  0.6
  -14.96 -40.2  0.6
  -12.09 -34.2  0.6
  -11.70 -33.2  0.6
  -10.31 -29.2  0.6
  -9.61 -27.2  0.6
  -8.22 -23.2  0.6
  -6.48 -18.2  0.6
  -5.08 -14.2  0.6
  -4.39 -12.2  0.6
  -1.95 -5.2   0.6
  -1.25 -3.2   0.6
  -0.20 -0.2   0.6
  3.46  2.8    0.6
  3.46  9.8    0.6
  11.33 23.3   0.6
1 11.92 35.3   0.6
 8.8   1.06  2.19
 8.8   1.06  2.31
 8.8   1.06  2.42
 8.8   1.12  2.19
 8.8   1.12  2.31
 8.8   1.12  2.42
 8.8   1.17  2.19
 8.8   1.17  2.31
 8.8   1.17  2.42
```

****

CROSS SECTION PROFILE

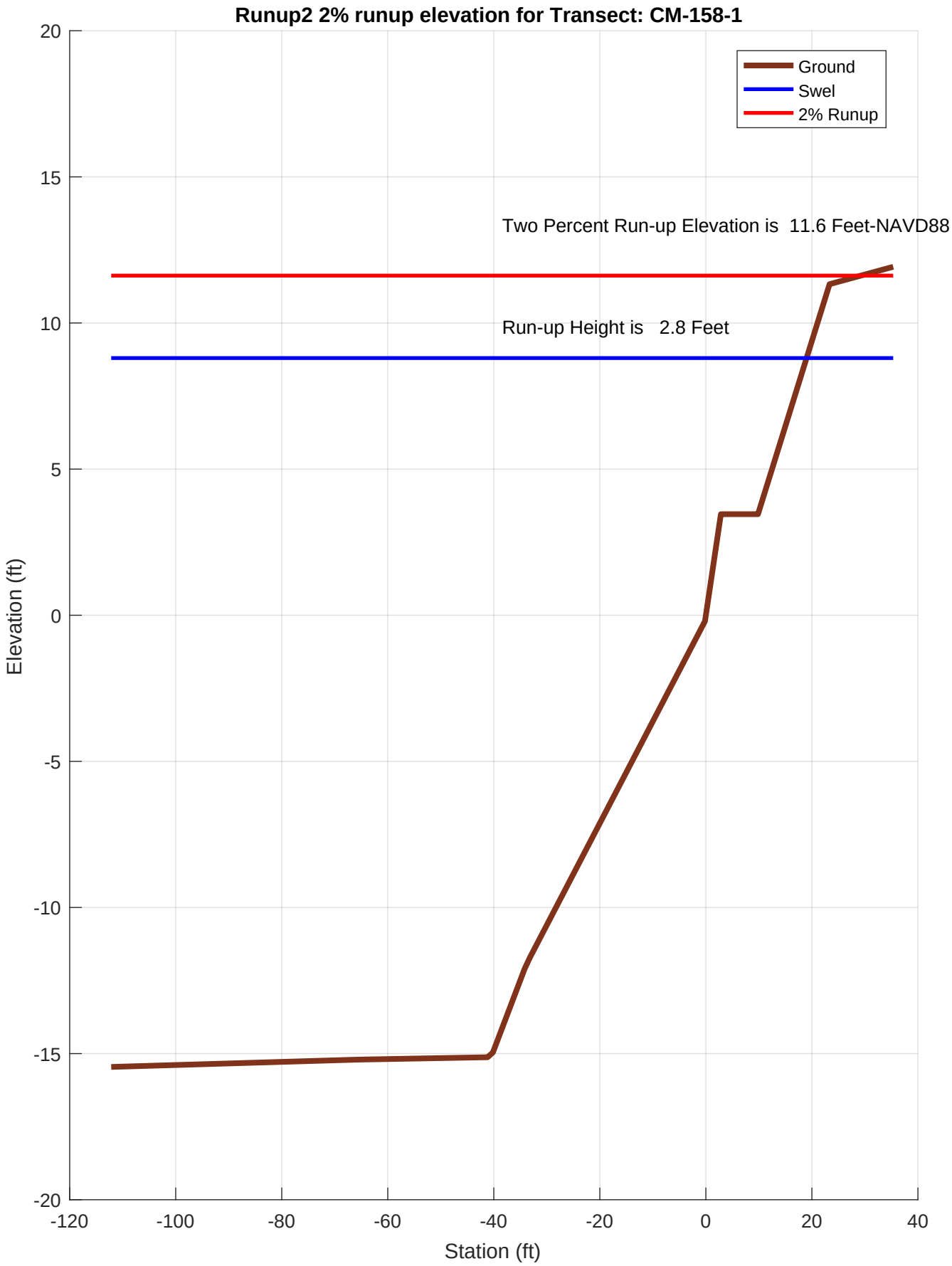
	LENGTH	ELEV.	SLOPE	ROUGHNESS
1	-112.0	-15.4		
			.00	.60
2	-66.2	-15.2		
			FLAT	.60
3	-65.2	-15.2		
			240.00	.60
4	-41.2	-15.1		
			5.00	.60
5	-40.2	-14.9		
			2.07	.60
6	-34.2	-12.0		
			3.33	.60
7	-33.2	-11.7		
			2.86	.60
8	-29.2	-10.3		
			2.90	.60
9	-27.2	-9.6		
			2.88	.60
10	-23.2	-8.2		
			2.87	.60
11	-18.2	-6.5		
			2.86	.60
12	-14.2	-5.1		
			2.90	.60
13	-12.2	-4.4		
			2.87	.60
14	-5.2	-1.9		
			2.86	.60
15	-3.2	-1.2		
			2.86	.60
16	-.2	-.2		
			.82	.60
17	2.8	3.5		
			FLAT	.60
18	9.8	3.5		
			1.72	.60
19	23.3	11.3		
			20.34	.60
20	35.3	11.9		

LAST SLOPE 21.00 LAST ROUGHNESS .60

****

OUTPUT TABLE

INPUT PARAMETERS			RUNUP RESULTS			
WATER LEVEL ABOVE DATUM (FT.)	DEEP WATER WAVE HEIGHT (FT.)	WAVE PERIOD (SEC.)	BREAKING SLOPE NUMBER	RUNUP SLOPE NUMBER	RUNUP ABOVE WATER LEVEL (FT.)	BREAKER DEPTH (FT.)
8.80	1.06	2.19	11	18	1.21	1.35
8.80	1.06	2.31	11	18	1.23	1.35
8.80	1.06	2.42	11	18	1.25	1.35
8.80	1.12	2.19	11	18	1.26	1.42
8.80	1.12	2.31	11	18	1.29	1.42
8.80	1.12	2.42	11	18	1.31	1.42
8.80	1.17	2.19	11	18	1.31	1.49
8.80	1.17	2.31	11	18	1.33	1.49
8.80	1.17	2.42	11	18	1.35	1.49


Runup2 2% runup elevation for Transect: CM-158-1
Ground
Swel
2% Runup
Two Percent Run-up Elevation is 11.6 Feet-NAVD88
Run-up Height is 2.8 Feet
Elevation (ft)
Station (ft)
20
15
10
5
0
-5
-10
-15
-20
-120
-100
-80
-60
-40
-20
0
20
40