Appendix D1 – Annex 4

Coastal Engineering and Riverine Hydraulics Summary

Monte Carlo Simulation Under With Project Conditions For South San Francisco Bay Shoreline Study

Final Summary Report

Prepared For:
U.S. Army Corps of Engineers
San Francisco District

Prepared By:
Noble Consultants, Inc.

July 2012

Table of Contents

List of Table

List of Figures

1.0 Introduction

The entire South San Francisco Bay Shoreline Study (SSFBSS) project site is located south of the Dumbarton Bridge at the far southern end of San Francisco Bay. The specific study area for the current with-project conditions study is bounded by Coyote Creek and Alviso Slough that encompasses Ponds A9 through A18 as shown in Figure 1-1. These salt ponds were previously used for salt production by Cargill, Inc. Ponds A9 through A17 are now owned by US Fish and Wildlife Service (USFWS) as part of the Don Edwards San Francisco Bay National Wildlife Refuge, while Pond A18 is now owned by the City of San Jose.

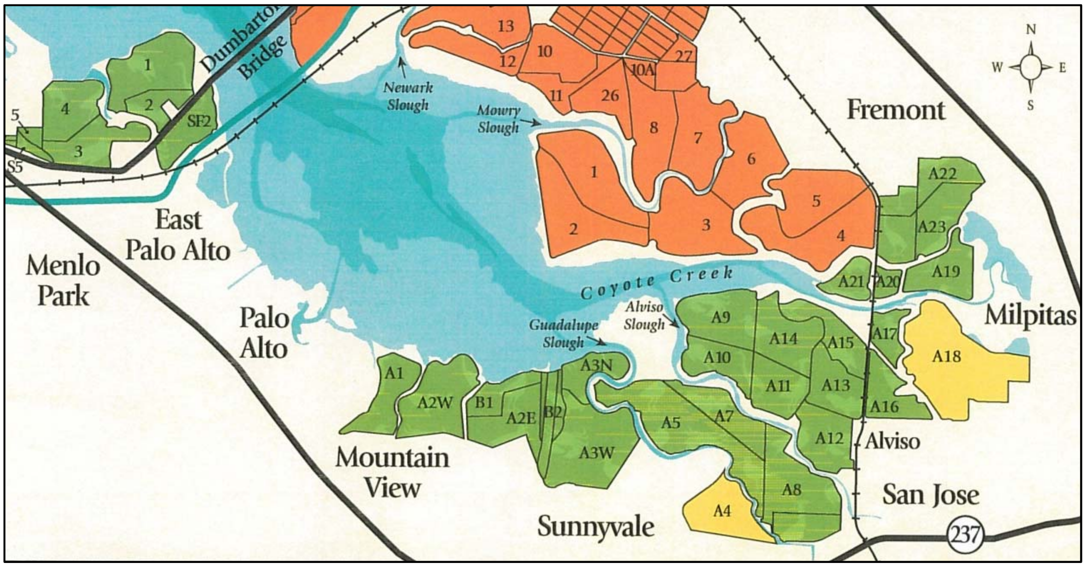


Figure 1-1. Project Site Location

The federally-conducted and locally-sponsored study has dual purposes of providing storm protection for the project basins landward of the protective levee and restoring the included salt ponds for marine habitat enhancement. The flood control component of the study involves the alteration of existing alignment with a well-engineered levee. The zone of potential project protection is in the Alviso Basin located landward of A12, A16 and A18 also shown in Figure 1-1.

The purpose of this summary report is to document the development of the Monte Carlo simulation technique that provides results via the uncertainty analysis for the with-project conditions to support the project decisions, based on the processes within the economic evaluation and environmental consideration.

2.0 Monte Carlo Simulations

The applied Monte Carlo simulation technique is a statistical approach to predict an uncertain system by recreating a random process to solve a problem which cannot be easily evaluated by a standard numerical analysis. This technique allows for the random sampling of a pre-defined (known) occurrence distribution of each individual element to statistically characterize the behavior of the uncertain system. A brief description of various components required to execute the Monte Carlo simulations is provided in the following sections.

2.1 Treatment of Outer Levee Failure

The protective levee within the project area is susceptible to breaching failure, which can be categorized into a static or dynamic failure. A method (ERDC) was formulated and developed by the Engineering Research and Development Center (ERDC) of the Corps of Engineers for the South San Francisco Shoreline Study during the without-project-conditions (F3) study phase to evaluate levee failure under storm water and wave conditions (Lee, 2009a & Lee, 2009b). During the with-project conditions phase (F4), an assessment was also performed to evaluate another method (Dean) that was developed by Dean et al (2010). The Dean method was formulated, based on various field observations and past experiments conducted by European researchers, to assess and validate levee failure resulting from storm wave attack during the Hurricane Katrina event (Dean and Ledden, 2010). Table 2-1 lists the analysis algorithm applied required for both methods to determine whether levee failure occurs under an oceanographic condition (NCI, 2012a).

Table 2-1. Comparison of ERDC and DEAN Methods for Levee Failure Analysis

Mechanism of Levee Failure	ERDC	Levee failure can occur on both sides of a levee structure via the static or dynamic process. For the dynamic process, the following principles was applied: • The critical time for the breach on the outer slope (bay side) depends on the wave-induced shear stress, storm duration, and crest width. • The critical time for the inboard slope (basin side) failure depends on the wave overtopping rate and crest width.
	DEAN	Levee failure occurs only on the inboard slope (basin side) from wave overtopping that induces the erosion work on the inboard slope. No reduction factor for influence of levee berm was applied.

After a comprehensive evaluation and comparison of the two methods, it was determined that the static failure method that was developed in the without-project conditions phase will still be applied to the current analysis. However, the Dean method will be used for the with-project conditions to assess the dynamic levee failure.

The static failure stems primarily from water seepage that creates a vertical exit gradient to cause levee failure. Uncertainty in quantifying the static failure of the protective levee does exist, as it is difficult to formulate a deterministic physical algorithm for the failure evaluation. Therefore, the static failure process is also characterized by a probabilistic approach to statistically determine the breach of the earth levee. The probability of the static failure as a function of the freeboard was developed in the F3 study. The same empirical curve was applied to determine whether a static failure occurs under a pre-determined critical water elevation on a yearly basis. The assessment logic is detailed in the Memorandum Record dated April 10, 2012 (Hubel, 2012). Table 2-2 lists two probabilistic parameters that were used in assessing the static failure.

Table 2-2. Parameters Used for Static Levee Failure

Probabilistic Parameter	*Derivation of Probability of occurrence*
Critical Elevation for Potential Levee Failure	Determined from a pre-generated curve on a yearly basis
Probability of Levee Failure	Obtained from the determined critical water elevation

Dynamic failure due to wave impingement against the protective levee was assessed by a levee erosion model that was proposed by Dean et al (2010). Levee failure occurs on the inboard slope (basin side) from wave overtopping that induces the erosion work on the inboard slope. The potential erosion work against a set of threshold values, based on three types of grass cover, was estimated from wave overtopping computed from empirical formulae that were developed by European researchers.

2.2 Physical Processes of Monte Carlo Simulations

Various physical processes are required to create the lookup database so that the Monte Carlo analysis can be executed. Each physical parameter was simulated for a range of forcing conditions to generate the necessary lookup tables used in the Monte Carlo simulations.

2.2.1 Long Wave Simulations

Long wave simulations consist of various forcing parameters such as tide, residual surge, wind speed, and wind direction under the Year 0 and Year 50 conditions.

The year 0 production simulations were performed for a set of synthesized events that cover the ranges of all the controlling parameters. The selected project plan were evaluated for year 50 (2067) conditions, with 2.13 ft (0.649 m) of sea level rise (SLR). The year 50 simulations incorporated anticipated accretion within the project ponds, as well as estimated channel evolution in the vicinity of the project area. The simulations also included both the without-breach and with-breach conditions at the various outer levee locations (Delta Modeling Associates, 2012). Predicted water levels were tabulated in the lookup tables that allow the interpretation of the responses of all the synthesized events randomly selected by the Monte Carlo Simulation (MCS) process during statistical analysis. Table 2-3 lists the values and conditions of forcing parameters used in the synthesized events for the long wave simulations.

Table 2-3. Parameters Used for Long Wave Model Simulations

Year	*SLR (ft)*	*Outer Levee Breached*	*Predicted Tide (ft, navd)*	*Residual Surge (ft)*	*Wind Direction (deg)*	*Wind Speed (mph)*
Year 0	0	Yes No	5.15 5.85 6.55 7.25	0.5 1.5 2.5	292.5° 315°	20 30 40
Year 50	2.13	Yes No	7.28 7.98 8.68 9.38	0.5 1.5 2.5	292.5° 315°	20 30 40

2.2.2 Short Wave Generation

Due to the sheltering effect provided by the neighboring salt ponds and levees, wind-generated short waves within the project site (Ponds A9 to Ponds A18 in Figure 1-1) are minimal. Therefore, simplified wave growth formulas that predict wave growth based on restricted fetches and duration-limited criteria (ACES, 1991) were applied to estimate the magnitude of short waves approaching the outer and inner levees in accordance with respective restricted fetches and duration. The forcing wind conditions, including wind speed and direction, to estimate wave heights are identical to those used in the long wave simulations (see Table 2-3). The generated wave height lookup tables were interpreted in the Monte Carlo simulations, based on randomly selected wind direction and speed.

2.2.3 Wave Runup and Overtopping

Waves and/or storm water surface elevation (WSE) can overtop the protective levee in three different scenarios as shown in Figure 2-1. Wave overtopping only occurs when the storm WSE is below the levee crest and the wave contribution is the only mechanism (Figure 2-1a). Surge overtopping occurs when the storm WSE is higher than the levee crest elevation without any wave action (see Figure 2-1b). The third scenario would be the combined contribution of waves and WSE overtop the levee (see Figure 2-1c). Three empirical formulae that were developed by ERDC (Hughes, 2007 and Hughes & Nadal, 2009) were employed to estimate the wave overtopping and/or surge flow and coded in the Monte Carlo simulations to determine the overtopped water volume.

2.2.4 Riverine Outbreak

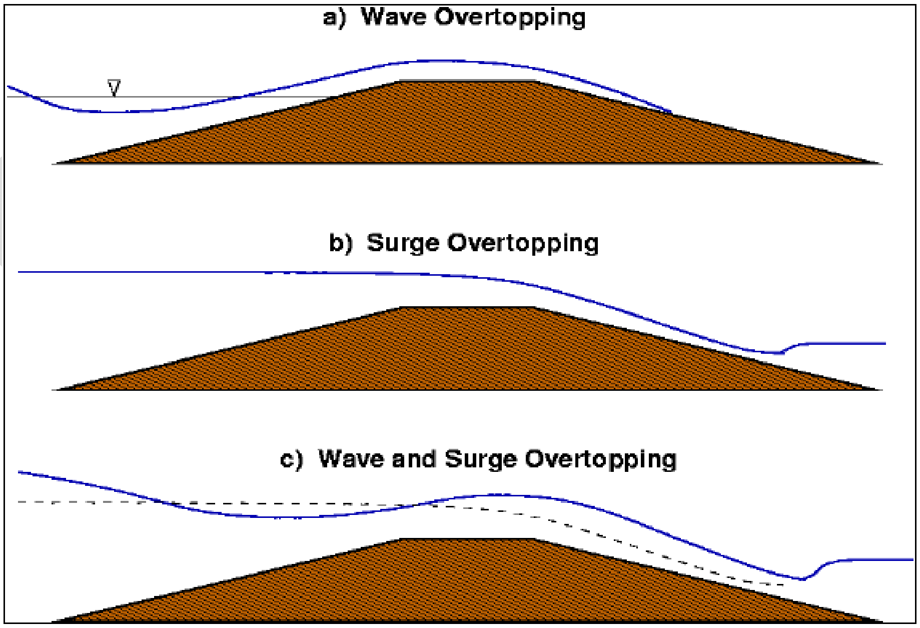
Project basins protected by the inner levee along A18, A16, A13 & A12 (see Figure 1-1) may be susceptible to storm-induced inundation stemming from surge overflow, locally-generated wave overtopping or the river breakout from Guadalupe River or Coyote Creek. The river breakout volume was estimated via a thorough riverine hydraulic analysis using the HEC-RAS model (Snyder, 2012). The lookup tables as presented in Tables 2-4 and 2-5 were also used in the Monte Carlo analysis to determine if the riverine breakout occurs.

Table 2-4. Guadalupe River Riverine Breakout Volumes

Annual Chance Exceedance (%)	50	20	10	5	2	1
Discharge at USGS Gage #11169025 (cfs)	3,602	6,466	8,196	10,790	14,770	18,597
Downstream WSE (ft NAVD 88)	Breakout Volume (ft^3)					
2.83	0	0	0	0	0	0
13.5	0	0	0	0	0	0

Table 2-5. Coyote Creek Riverine Breakout Volumes

Annual Chance Exceedance (%)	50	20	10	5	2	1	0.4	0.2
Discharge at USGS Gage #11169025 (cfs)	3,300	6,200	8,400	10,500	13,000	14,500	16,000	18,000
Downstream WSE (ft NAVD 88)	Breakout Volume (ft^3)							
2.83	0	0	0	0	0	0	127,656	3,398,868
13.5	0	0	0	0	0	0	128,520	3,400,884



Source: USACE, 2008

Figure 2-1. Types of Levee Overtopping

2.2.5 Alviso Basin Inundation

Based on the 2010 LiDAR topography within the Alviso Basin, the relation between the inundation volume versus the basin elevation was derived and presented in Figure 2-2. This correlation curve allows for determination of inundation depth based on the combined water volume into the basin whether it is from the riverine outbreak or wave-induced overtopping.

2.3 Overall Structure of Monte Carlo Simulation

2.3.1 Control Parameters

The ultimate goal of the Monte Carlo simulation is to statistically determine the recurrence of water surface elevation (WSE) at the protective levee so that an optimal engineering design of the levee can be drawn. Various control parameters that dictate the WSE at the protective levee during a storm event include astronomical tide, residual tide, and wind direction and speed (Andes & Wu, 2012). Table 2-6 briefly describes each parameter and the derivation of the associated probability of occurrence. In total, five control parameters are employed in this Monte Carlo simulation.

Using tide measurements at Fort Point in San Francisco Bay for the past 104 years, the cumulative distribution functions (CDFs) for the number of storms per year, astronomical tide, and residual tide were deduced according to the following criteria:

- The measured water surface elevation is ≥ 6.9 feet, Mean Lower Low Water
- The residual tide is ≥ 0.5 feet

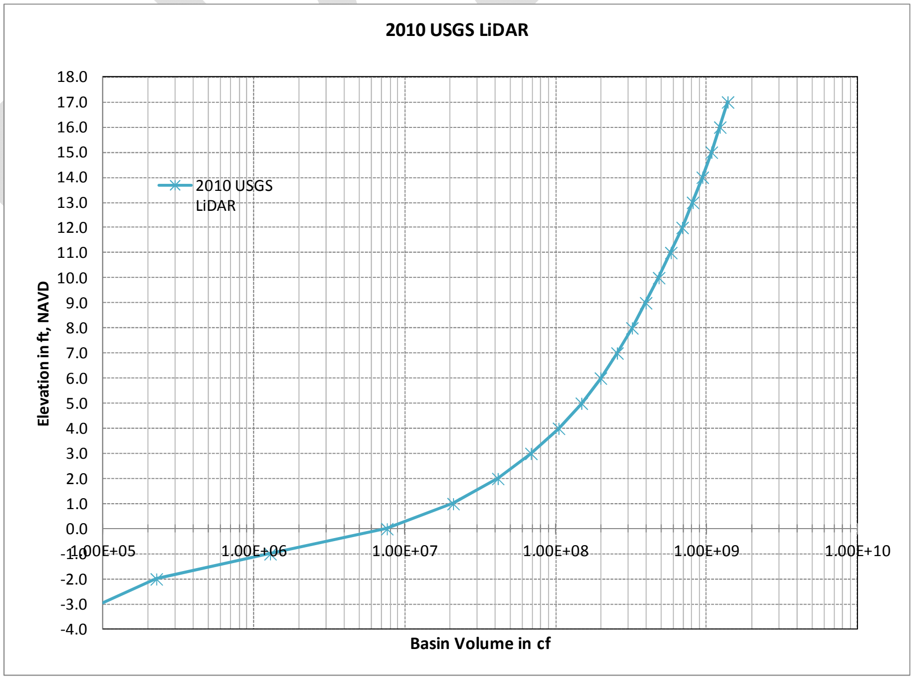


Figure 2-2. Inundation Volume versus Basin Elevation

Figures 2-3 to 2-4 show the deduced CDF curves for these three parameters. Historically recorded data of wind direction and speed at San Francisco Airport (SFO) that is located in the central bay and at Moffat Field located in the south bay were respectively analyzed to derive the CDF curves for assessment of wind-induced setup and locally-generated fetch-limited waves. A preliminary analysis indicates that two primary winds directions of 292.5° and 315° can induce a measurable setup and also produce locally-generated waves due to the major alignment of the south bay and the geographic location of the study area. Wind data at San Francisco Airport was applied, via the UnTRIM long wave model, to estimate the wind-induced setup, while the data at Moffat Field was used for the estimation of short wave conditions (i.e., wave height, Hs and wave period, Ts). Figures 2-5 to 2-9 illustrate the derived CDF curves for wind direction and speed at the two respective gage stations (NCI, 2012b).

Table 2-6. Control parameters for Monte Carlo Simulation

Control Parameter	*Derivation of Probability of occurrence*
Number of Storms Per Year	Based on historical storm events per year that satisfies the sampling criteria of astronomical and residual tide
Astronomical Tide	Predicted tides obtained from selected events
Residual Tide	Residual tides obtained from selected events
Wind Direction	Based on the historical wind data recorded at San Francisco Airport for wind-setup estimate
	Based on the historical wind data recorded at Moffat Field to be used for estimation of locally-generated waves
Wind Speed	Based on wind data recorded at San Francisco Airport for selected wind directions (wind setup estimate)
	Based on wind data recorded at Moffat Field for selected wind directions (estimate of locally-generated waves)

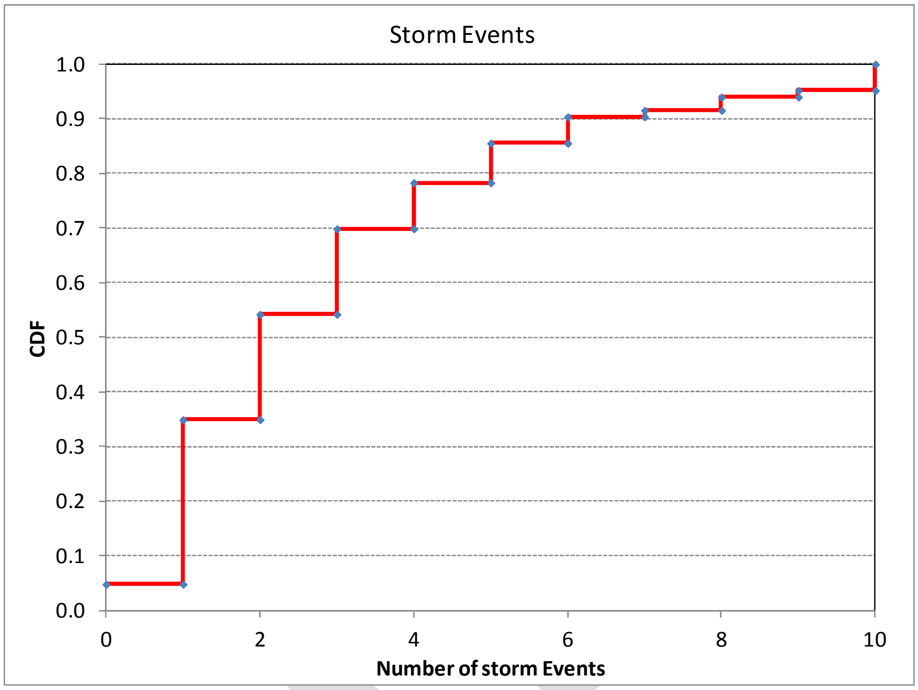


Figure 2-3. Cumulative Distribution Function for Number of Storm Events Annually

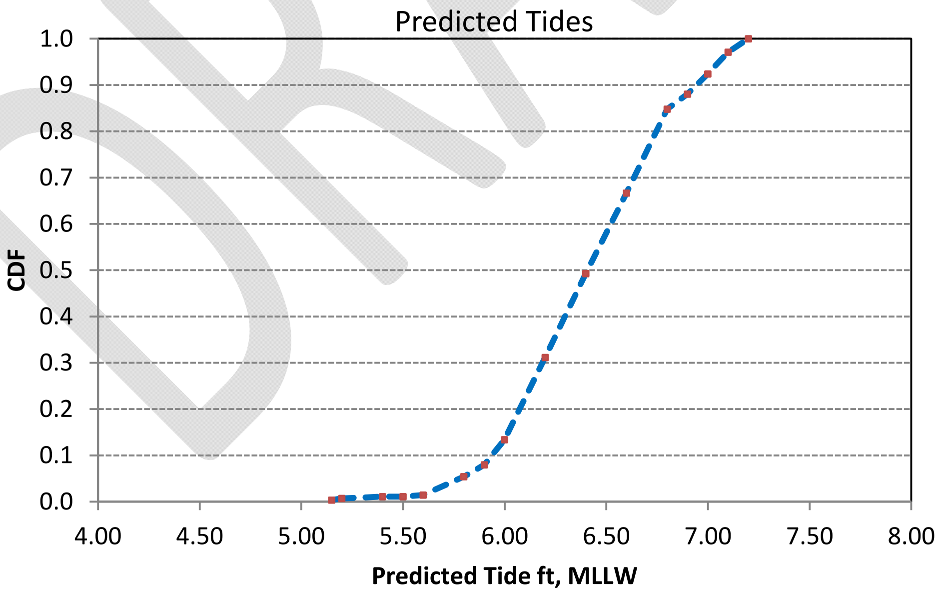


Figure 2-4. Cumulative Distribution Function for Astronomical Tide

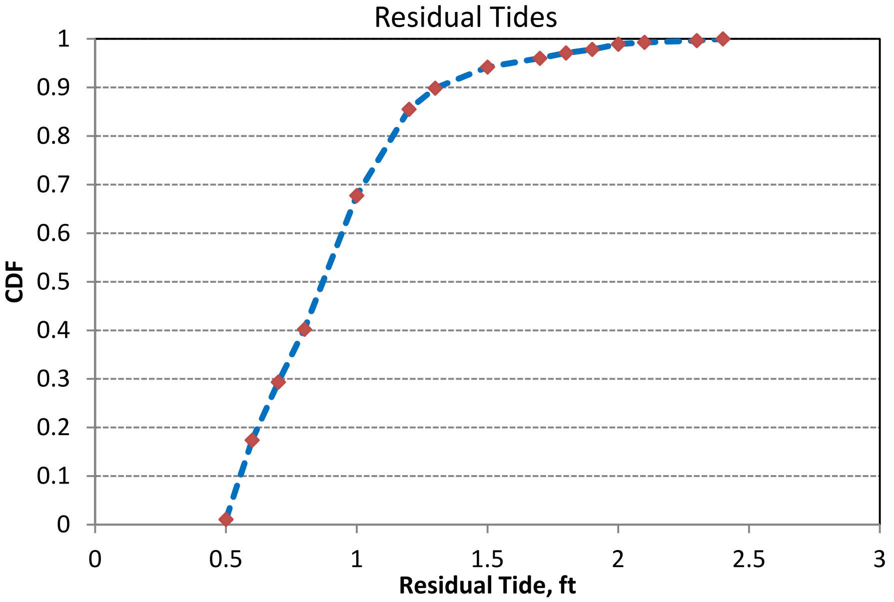
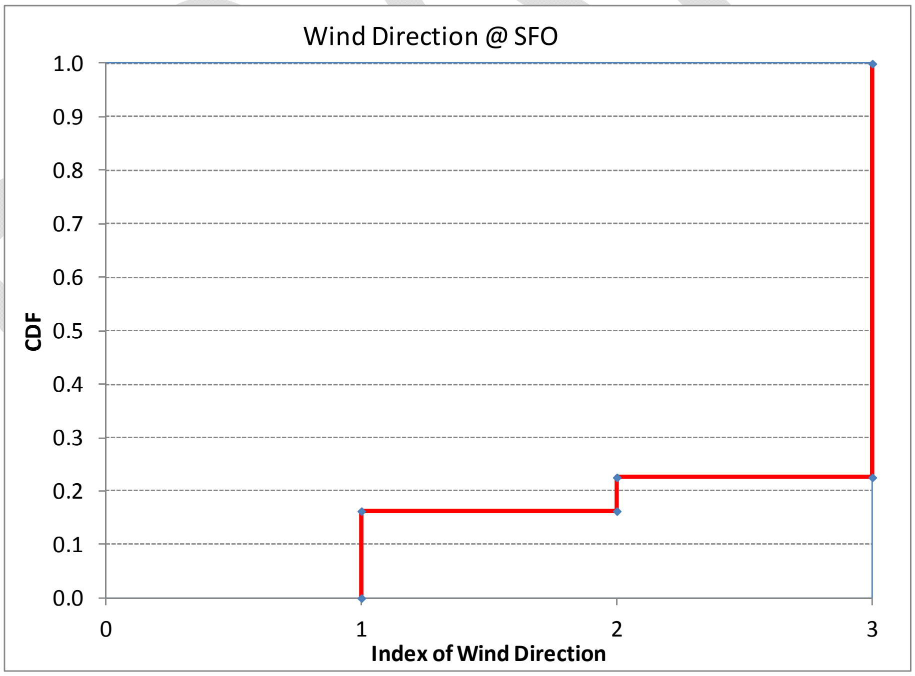


Figure 2-5. Cumulative Distribution Function for Residual Tide



Note: 292.5° =Index 1, 315° =Index 2 & all other directions =Index 3

Figure 2-6. Cumulative Distribution Function for Wind Direction at San Francisco Airport

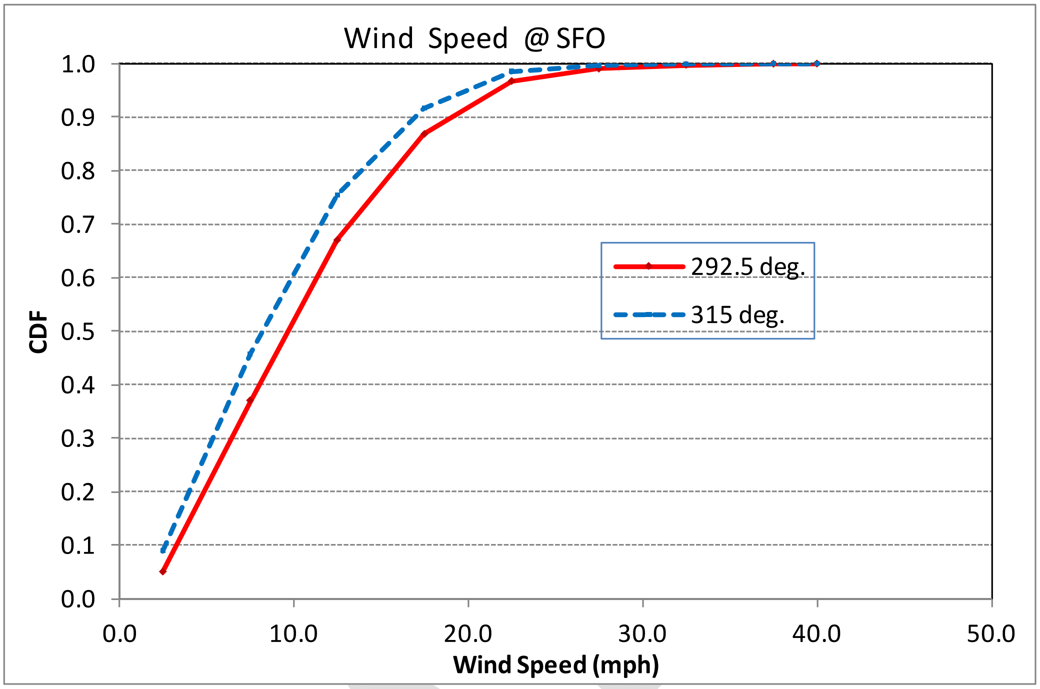
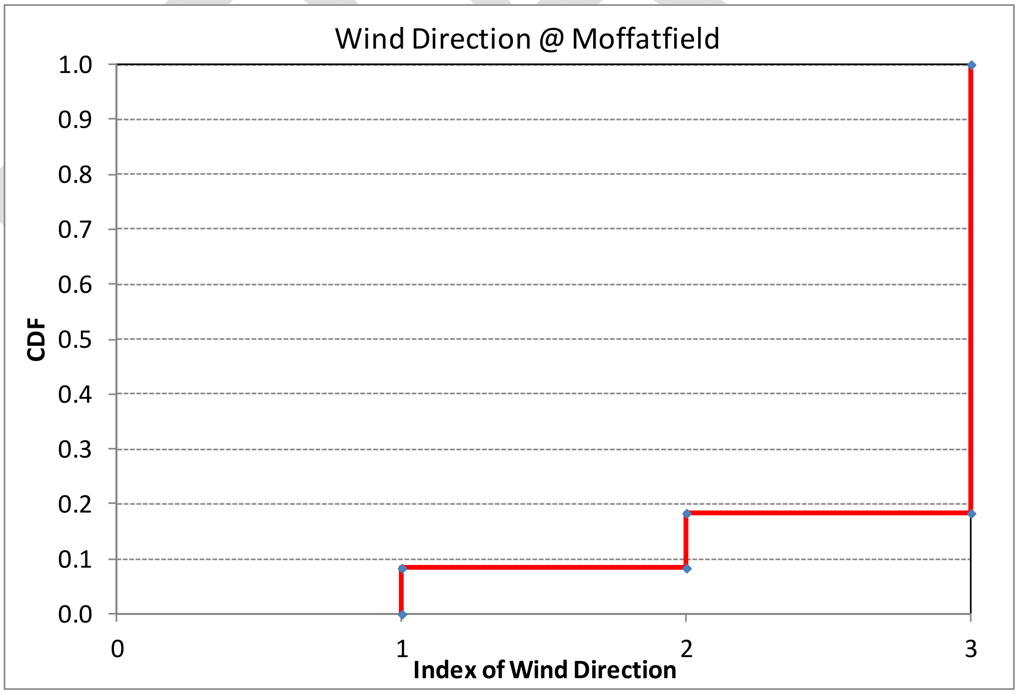


Figure 2-7. Cumulative Distribution Function for Wind Speed at San Francisco Airport



Note: 292.5° =Index 1, 315° =Index 2 & all other directions =Index 3

Figure 2-8. Cumulative Distribution Function for Wind Direction at Moffat Field

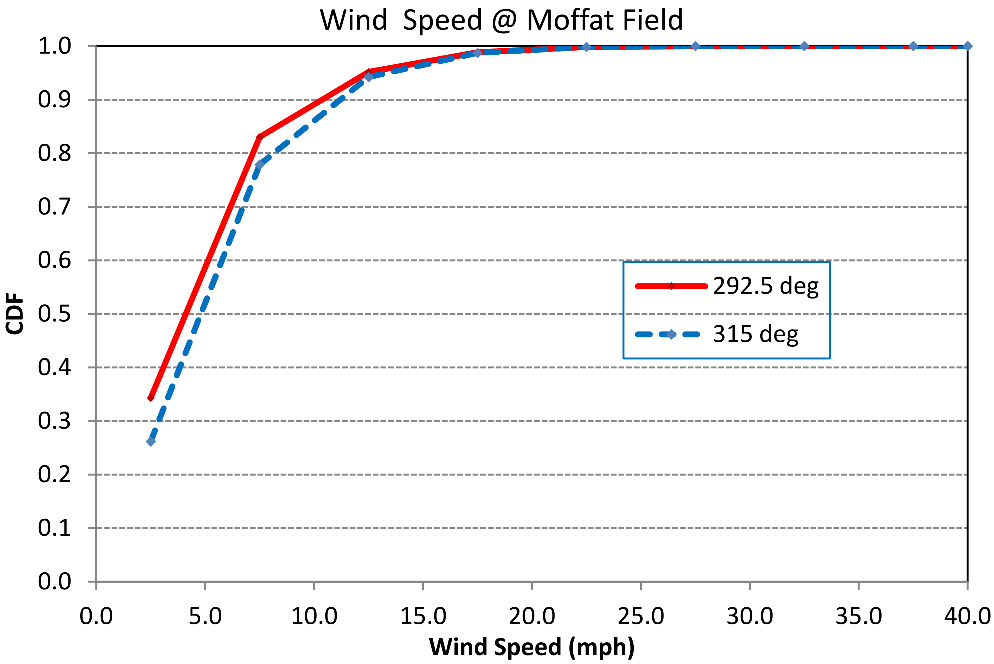


Figure 2-9. Cumulative Distribution Function for Wind Speed at Moffat Field

2.3.2 Algorithm of Monte Carlo Simulations

The procedure to execute the Monte Carlo Simulation in determining the water surface elevation at inner levee and the potential inundation within the project basins is presented in Figure 2-10 and delineated in the following:

- Randomly select the number of storms for each simulation year, based on the derived CDF.
- Randomly select an astronomical tide and a residual tide at the Presidio gage, based on the derived CDFs for each storm event.
- Determine the water surface elevation (WSE) at outer levee from the generated long wave lookup table via the UnTRIM modeling.
- Randomly select a wind direction and the associated speed, based on the derived CDFs at San Francisco Airport, and determine the wind setup via a wind-setup lookup table.
- Perform a static failure analysis from the combined water surface elevation at outer levee.
- Select the water surface elevation (WSE) at inner levee under the outer levee breached conditions for subsequent analysis, if a static failure occurs.
- Continue with the dynamic failure analysis of outer levee, if a static failure does not occur.
- Randomly select a wind direction and associated speed, based on the derived CDFs at Moffat Field and determine the short wave conditions (Hs & Ts).
- Perform a dynamic failure analysis of outer levee with the derived wave conditions.
- Select the water surface elevation at inner levee under the outer levee breached conditions for subsequent analysis, if a dynamic failure occurs.

- Otherwise, select the water surface elevation at inner levee under the intact outer levee conditions for subsequent analysis, if a dynamic failure does not occur.
- Compute short wave conditions (Hs & Ts) within salt ponds located bayward of the protective inner levee.
- Estimate water volume that enters the project basins due to wave overtopping and surge overflow, if occurs.
- Determine the river discharge rate during the selected storm event via an empirical formula as a function of the residual tide.
- Interpolate the lookout tables to obtain the breakout water volume from the two rivers (i. e., Guadalupe River and Coyote Creek).
- Estimate the inundation depth within the basins from the rating curve between the water volume and flooding depth for one storm event.
- Repeat the same procedure for all storm events in a year.
- Repeat the same procedure for the entire simulation years to complete one simulation.

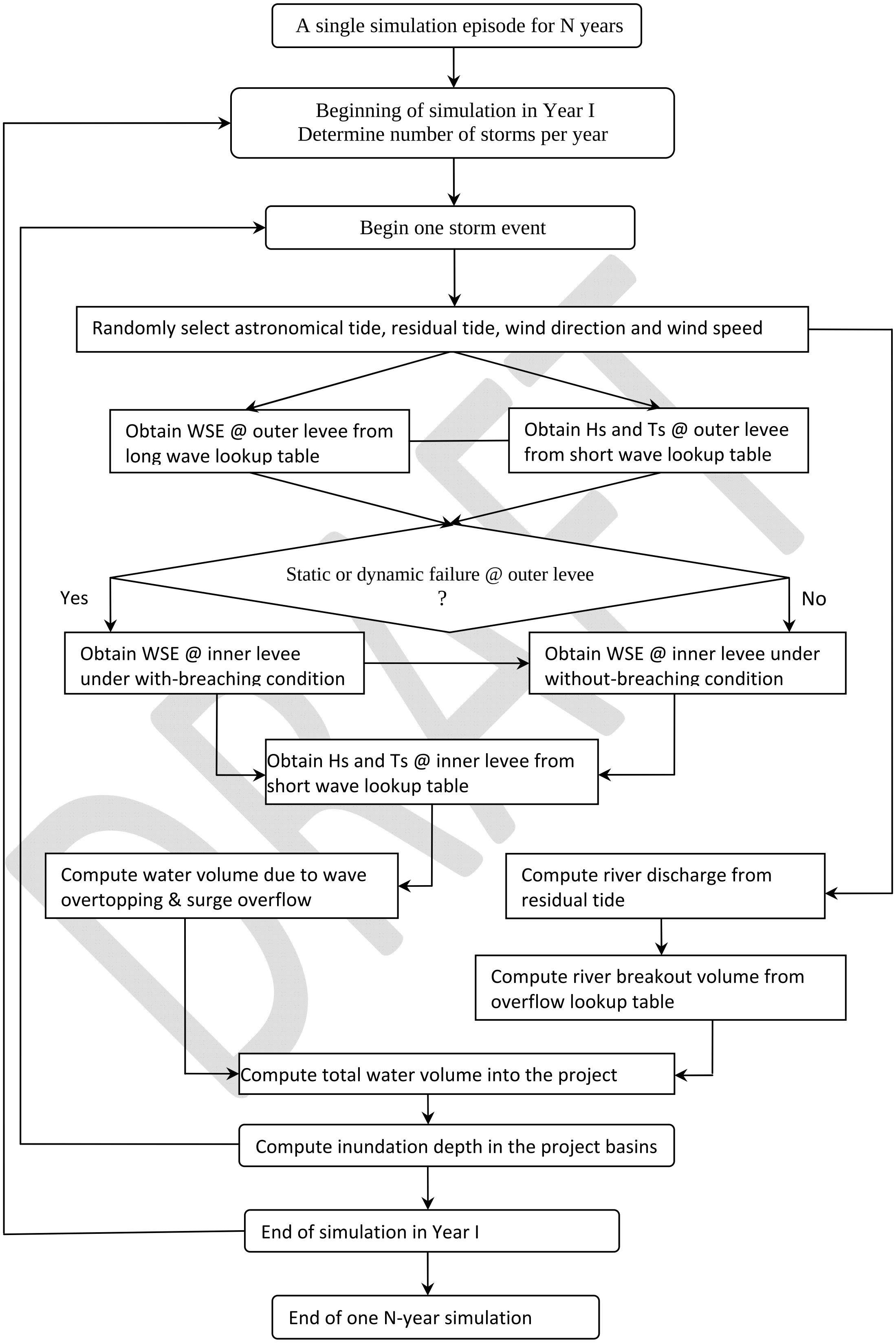


Figure 2-10. Flow Chart of Monte Carlo Simulation

3.0 Simulated Results

Each Monte Carlo simulation was executed for a 500-year duration. A comparison was made between 100, 200 and 400 simulations for determining the statistics of the 1000-year return period. It was found that the difference of the results from the three numbers of simulations is minimal. Therefore, 100 simulations, each with a duration of 500 years, were selected for all simulation cases to derive the statistical representation. The results from multiple Monte Carlo Simulations including both project conditions in Year 0 and Year 50 are respectively presented herein.

3.1 Simulation Layouts

Flood frequency was computed for two levee layouts, the locally preferred alignment (LPA) and the national economic development (NED) alignment. Figure 3-1 outlines both levee layouts with yellow lines. The layouts are very similar and identical east of Artesian Slough (Points 14 through 17). The primary difference in both layouts occurs at New Chicago Marsh. The NED layout includes the protection of New Chicago Marsh, Points 18 through 22, while the LPA excludes the marsh. Accordingly, the flood frequency analysis for the LPA does not include Points 18 through 22 because they are not located on the bay-side of the proposed layout. Both layouts have a similar alignment east of Artesian Slough, Points 14 through 17, and south of New Chicago Marsh, Points 10 through 11.

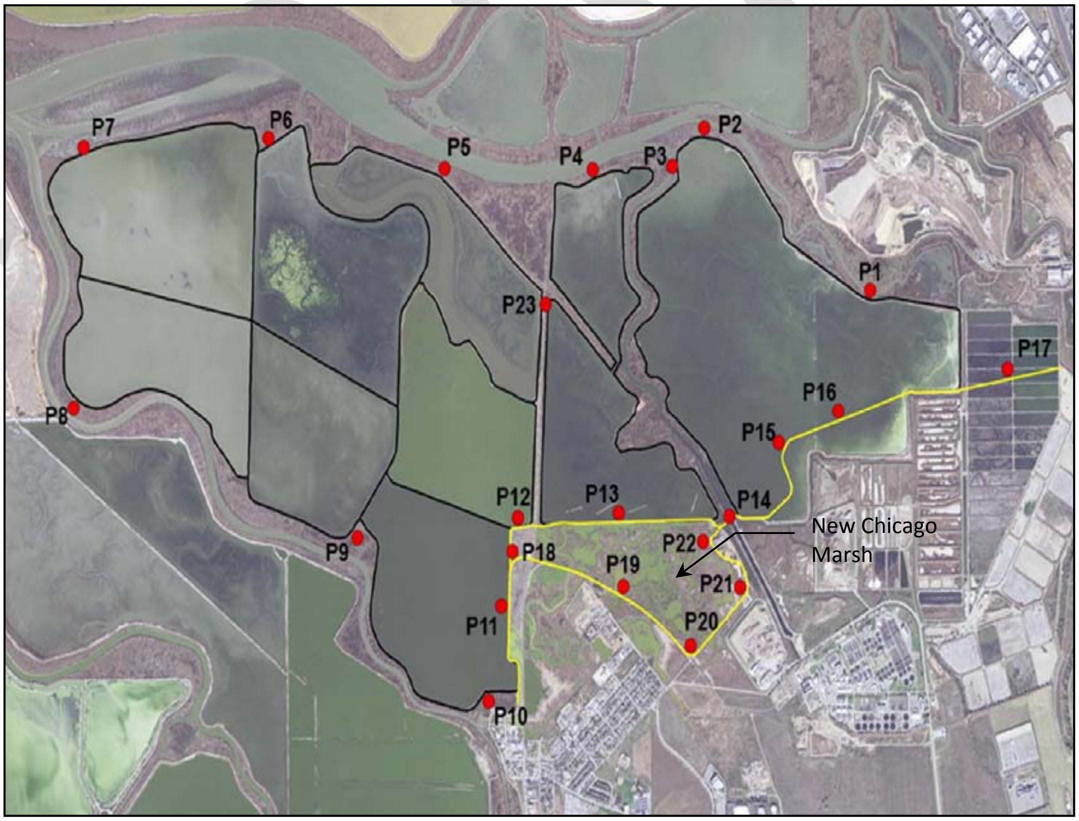


Figure 3-1. Hydrodynamic model simulation output locations

3.2 Simulation Results

MCS was used to estimate water surface elevation in terms of flood stage frequency at ten points along the outer levees and thirteen points along the inner levee of the Tentative NED and LPA alignments, summarized in Table 3-1, for both Yr-0 and Yr-50 conditions.

Table 3-1. Selected hydrodynamic Point locations to estimate peak water level

	Points Selected
Outer Levee	1,2,3,4,5,6,7,8,9, and10
Inner Levee	11,12,13,14,15,16,17,18,19,20,21,22, and 23

The results for two representative outer levee points and three or four representative inner levee points for both Tentative NED and LPA alignments under Yr-0 and Yr-50 conditions were presented in the following sections.

3.2.1 Yr-0 Tentative NED Results

Representative flood stage frequency results were presented at the outer levees, Points 3 and 7, in Figures and Tables 3-2 through 3-3 for the Yr-0 condition. Flood stage frequency results were also presented in Figures and Tables 3-4 to 3-6 for Points 14, 16 and 20 along the inner levees for the Yr-0 condition. All the statistical results presented include 5%, 50%, and 95% confidence limits.

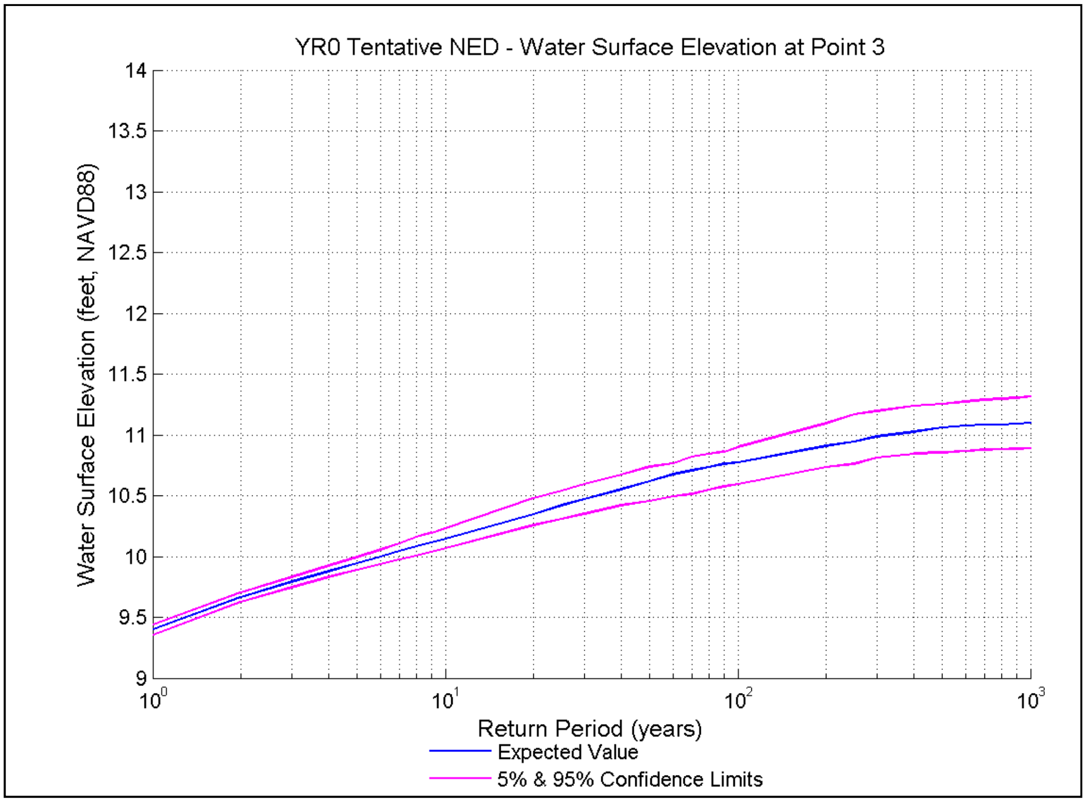


Figure 3-2. Flood stage frequency at Point 3 of Yr-0 Tentative NED alignment

Table 3-2. Flood stage frequency at point 3 of Yr-0 Tentative NED alignment

Return Period (yrs)	5% (feet, NAVD88)	50% (feet, NAVD88)	95% (feet, NAVD88)
1	9.35	9.40	9.44
2	9.63	9.67	9.70
5	9.89	9.94	10.00
10	10.07	10.15	10.23
25	10.31	10.42	10.54
50	10.46	10.62	10.74
100	10.59	10.77	10.90
250	10.76	10.95	11.17
500	10.86	11.06	11.25
1000	10.89	11.10	11.32

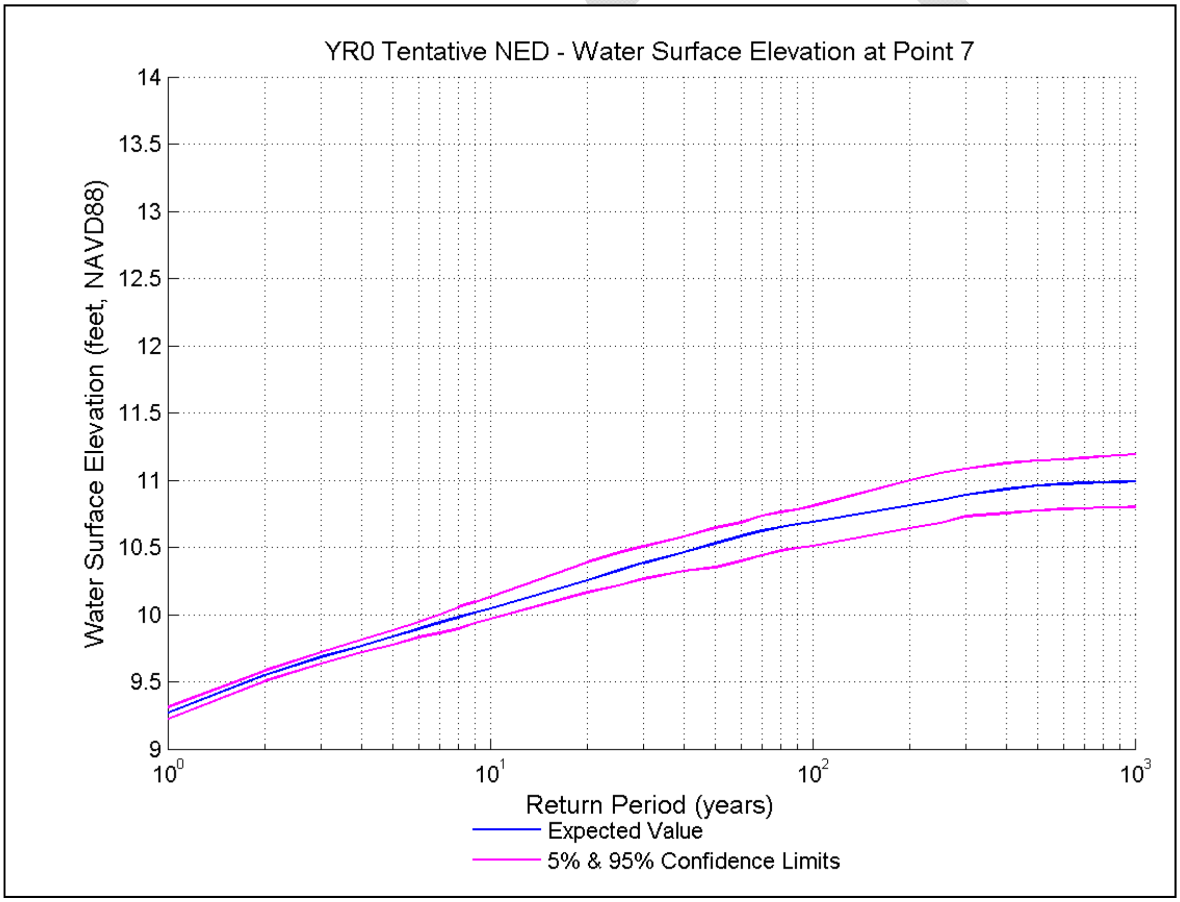


Figure 3-3. Flood stage frequency at point 7 of Yr-0 Tentative NED alignment

Table 3-3. Flood stage frequency at point 7 of Yr-0 Tentative NED alignment

Return Period (yrs)	5% (feet, NAVD88)	50% (feet, NAVD88)	95% (feet, NAVD88)
1	9.22	9.27	9.31
2	9.51	9.55	9.58
5	9.78	9.84	9.88
10	9.97	10.05	10.14
25	10.22	10.33	10.46
50	10.35	10.53	10.65
100	10.51	10.69	10.81
250	10.68	10.85	11.05
500	10.78	10.96	11.15
1000	10.80	10.99	11.19

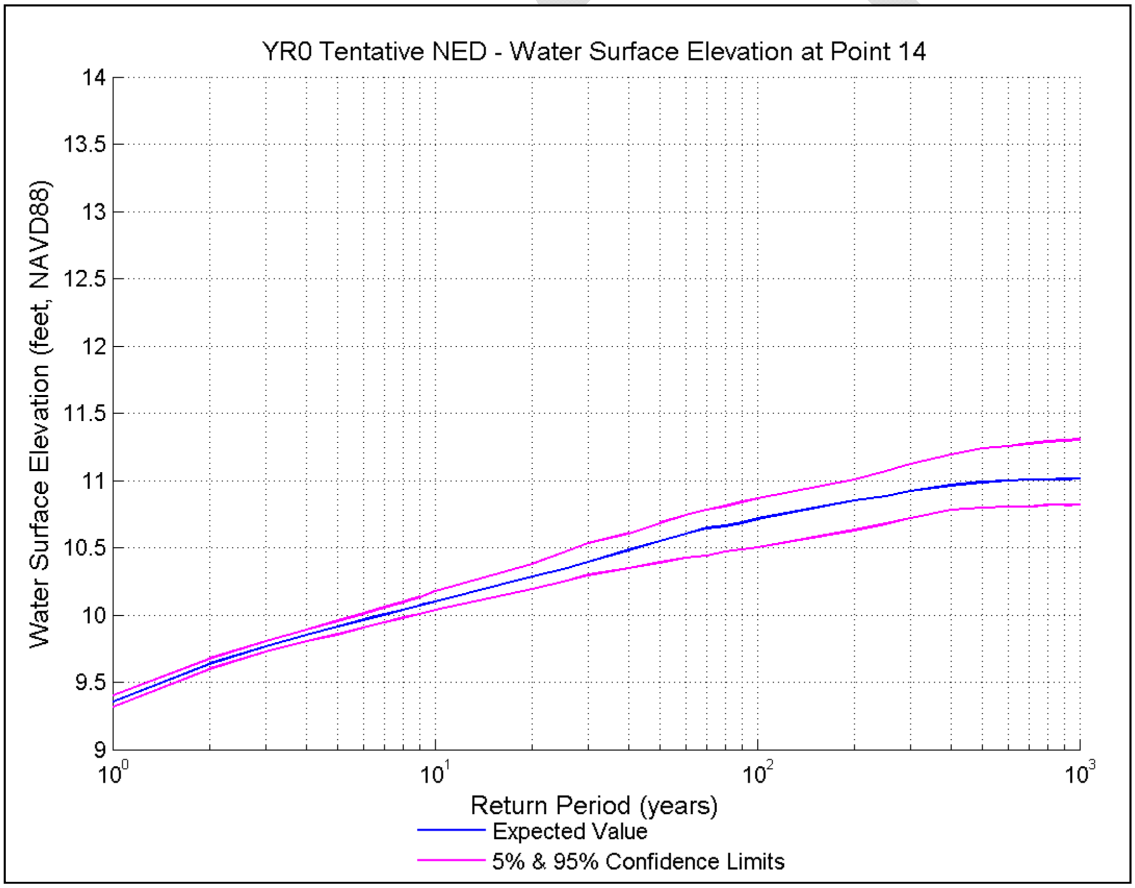


Figure 3-4. Flood stage frequency at Point 14 of Yr-0 Tentative NED alignment

Table 3-4. Flood frequency at Point 14 of Yr-0 Tentative NED alignment

Return Period (yrs)	5% (feet, NAVD88)	50% (feet, NAVD88)	95% (feet, NAVD88)
1	9.31	9.35	9.40
2	9.60	9.64	9.68
5	9.86	9.91	9.95
10	10.04	10.10	10.18
25	10.25	10.34	10.47
50	10.39	10.55	10.68
100	10.50	10.72	10.87
250	10.68	10.88	11.07
500	10.80	10.99	11.24
1000	10.82	11.02	11.31

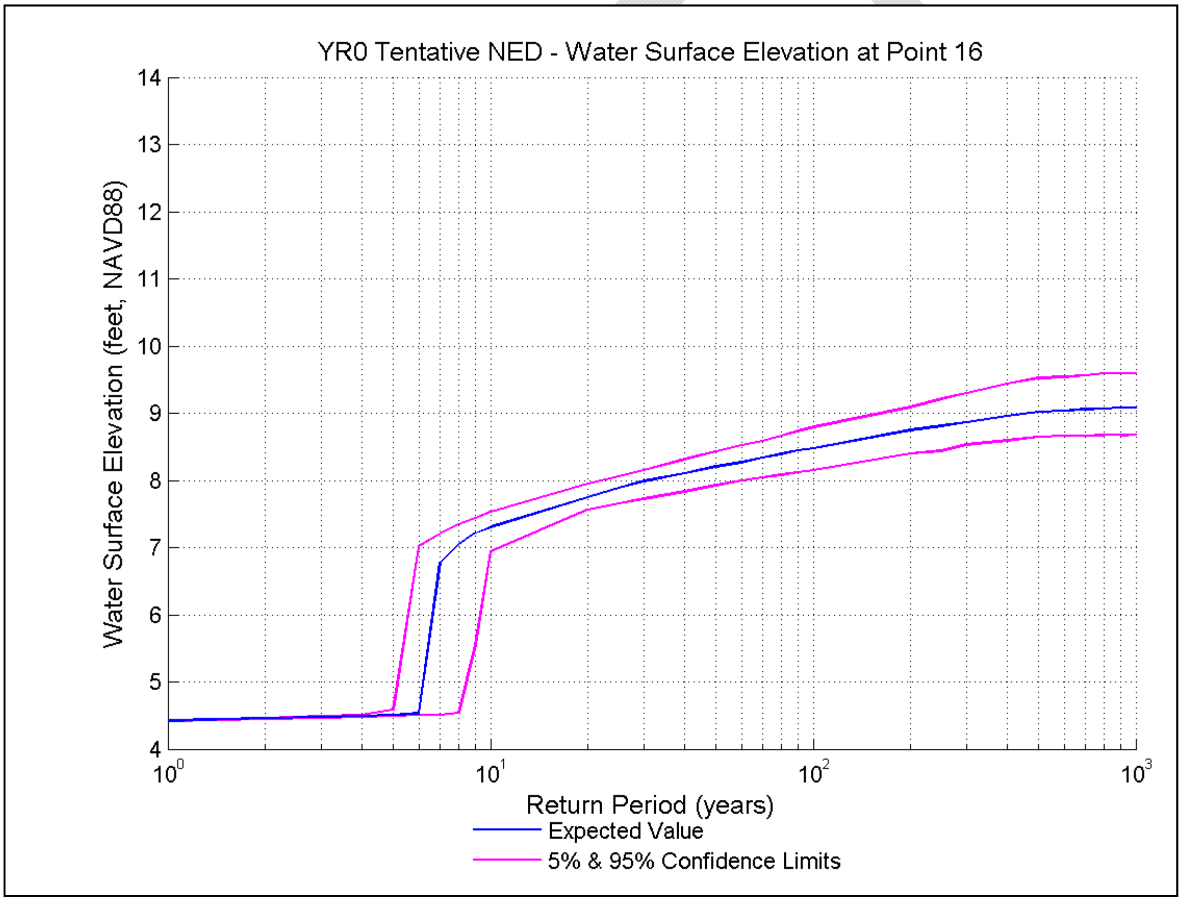


Figure 3-5. Flood stage frequency at Point 16 of Yr-0 Tentative NED alignment

Table 3-5. Flood stage frequency at Point 16 of Yr-0 Tentative NED alignment

Return Period (yrs)	5% (feet, NAVD88)	50% (feet, NAVD88)	95% (feet, NAVD88)
1	4.42	4.42	4.43
2	4.45	4.46	4.46
5	4.49	4.51	4.59
10	6.94	7.30	7.53
25	7.65	7.89	8.05
50	7.93	8.20	8.43
100	8.16	8.48	8.79
250	8.44	8.81	9.21
500	8.65	9.02	9.52
1000	8.68	9.09	9.59

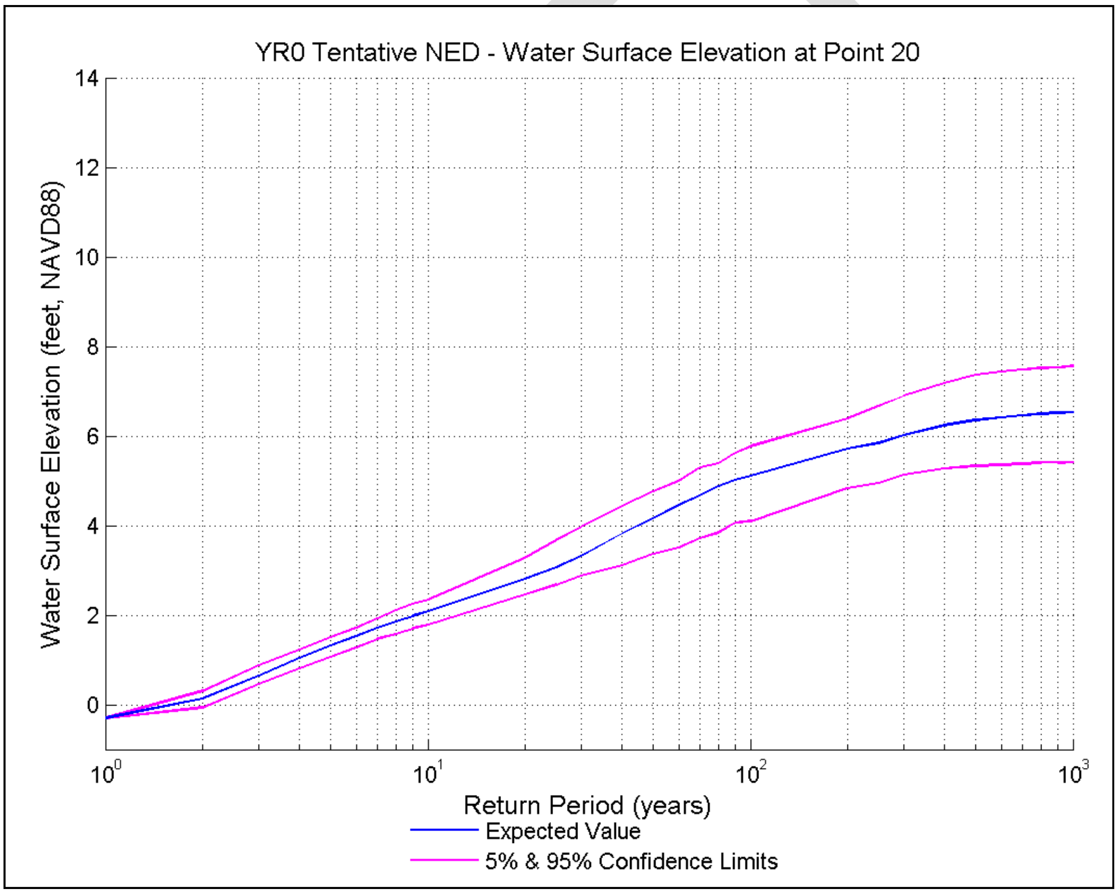


Figure 3-6. Flood stage frequency at Point 20 of Yr-0 Tentative NED alignment

Table 3-6. Flood stage frequency at Point 20 of Yr-0 Tentative NED alignment

Return Period (yrs)	5% (feet, NAVD88)	50% (feet, NAVD88)	95% (feet, NAVD88)
1	-0.30	-0.30	-0.30
2	-0.06	0.13	0.31
5	1.08	1.32	1.51
10	1.78	2.09	2.34
25	2.68	3.08	3.67
50	3.37	4.17	4.77
100	4.10	5.12	5.77
250	4.95	5.84	6.66
500	5.33	6.36	7.36
1000	5.42	6.53	7.56

3.2.2 Yr-50 Tentative NED Results

Representative flood stage frequency results were presented at the outer levees, Point 3 and 7, in Figures and Tables 3-7 through 3-8 for the Yr-50 condition. Flood stage frequency results were also derived for Points 14, 16 and 20 along the inner levees for the Yr-50 condition as presented in Figures and Tables 3-9 to 3-11. All the statistical results presented include 5%, 50%, and 95% confidence limits.

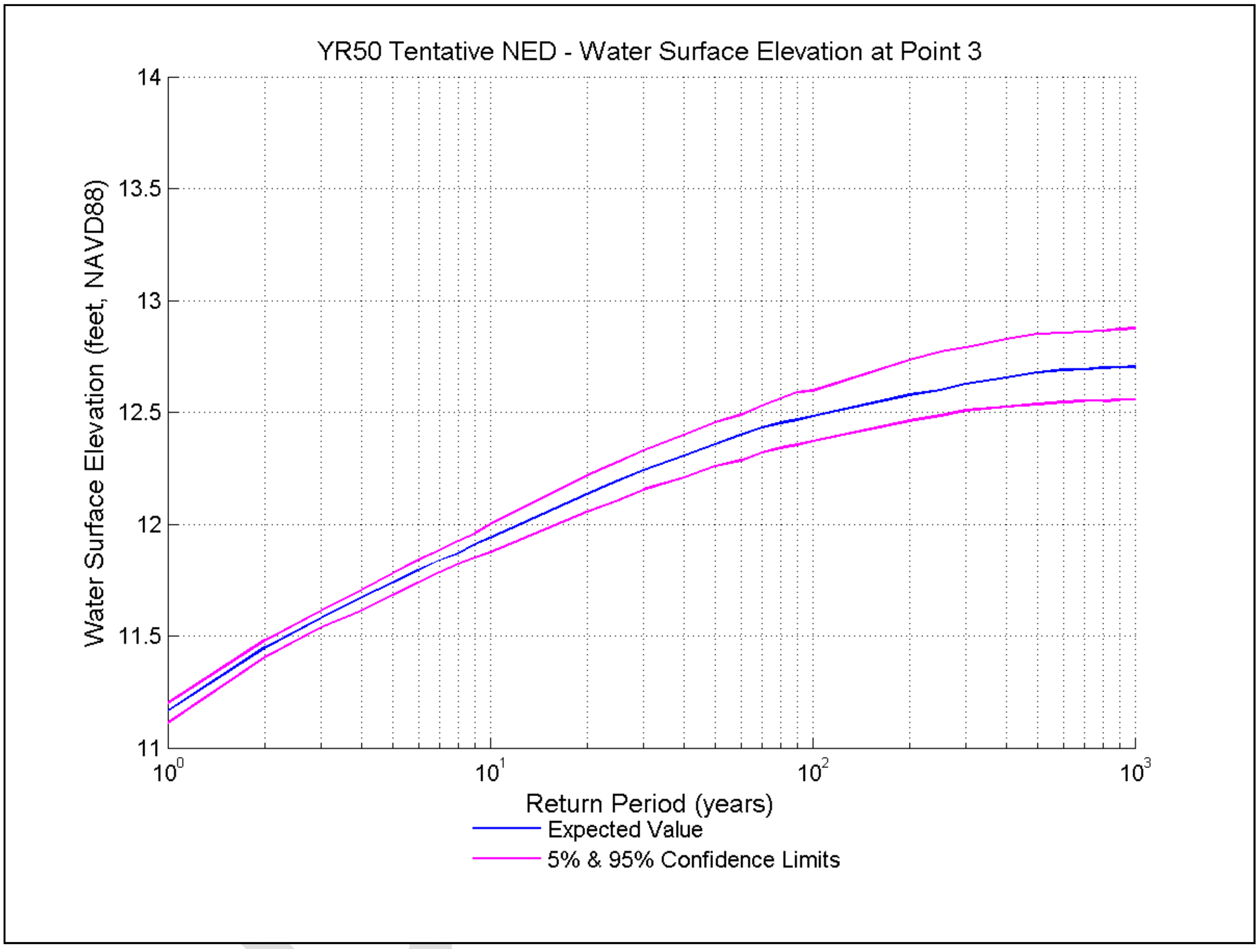


Figure 3-7. Flood stage frequency at Point 3 of Yr-50 Tentative NED alignment

Table 3-7. Flood stage frequency at Point 3 of Yr-50 Tentative NED alignment

Return Period (yrs)	5% (feet, NAVD88)	50% (feet, NAVD88)	95% (feet, NAVD88)
1	11.11	11.16	11.20
2	11.41	11.45	11.48
5	11.68	11.74	11.78
10	11.88	11.94	12.00
25	12.11	12.20	12.28
50	12.26	12.36	12.45
100	12.37	12.48	12.60
250	12.48	12.60	12.77
500	12.54	12.68	12.85
1000	12.56	12.70	12.88

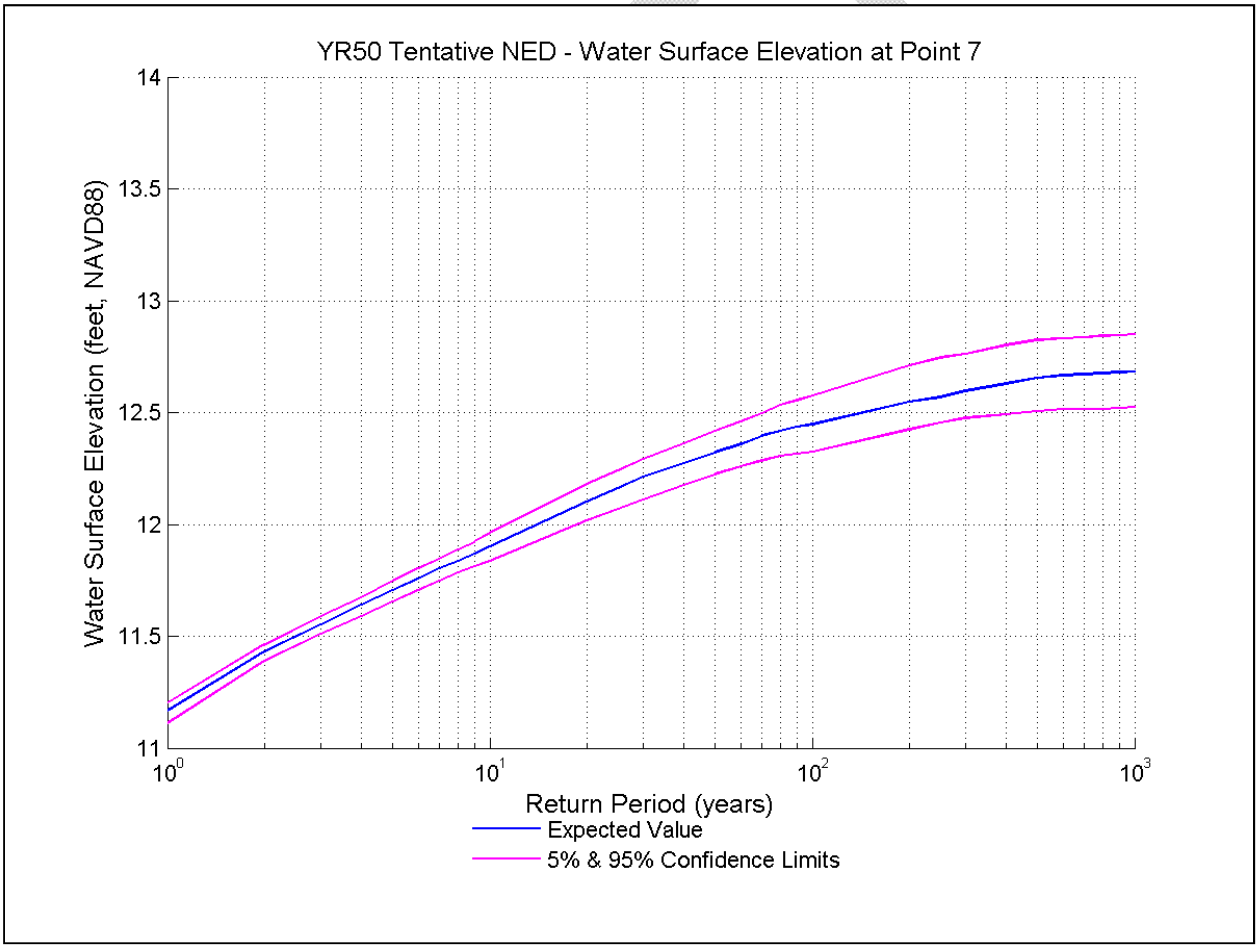


Figure 3-8. Flood stage frequency at Point 7 of Yr-50 Tentative NED alignment

Table 3-8. Flood stage frequency at Point 7 of Yr-50 Tentative NED alignment

Return Period (yrs)	5% (feet, NAVD88)	50% (feet, NAVD88)	95% (feet, NAVD88)
1	11.11	11.17	11.20
2	11.39	11.43	11.46
5	11.66	11.71	11.75
10	11.84	11.90	11.96
25	12.07	12.16	12.24
50	12.23	12.32	12.42
100	12.32	12.45	12.57
250	12.46	12.57	12.75
500	12.51	12.66	12.82
1000	12.52	12.68	12.85

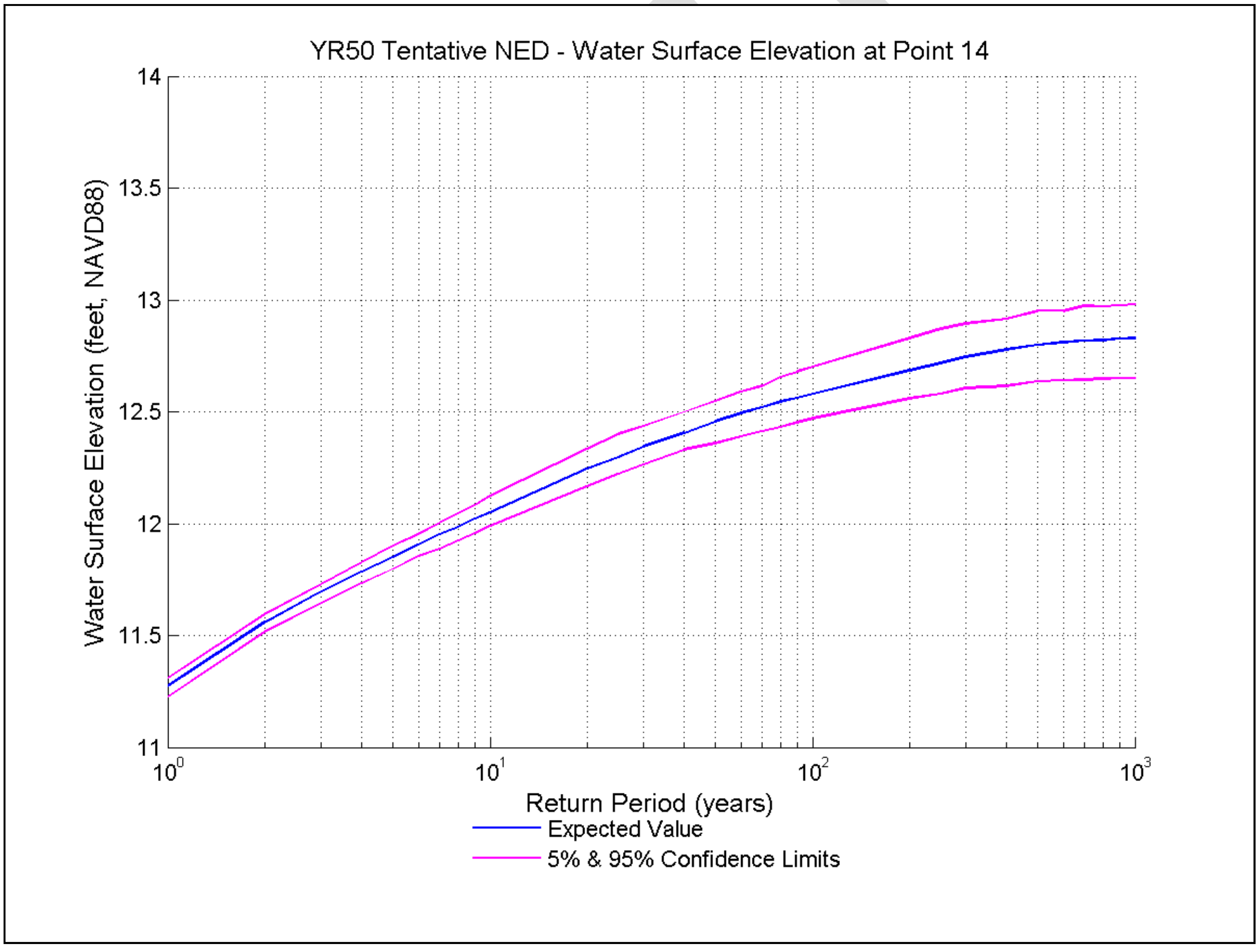


Figure 3-9. Flood stage frequency at Point 14 of Yr-50 Tentative NED alignment

Table 3-9. Flood stage frequency at Point 14 of Yr-50 Tentative NED alignment

Return Period (yrs)	5% (feet, NAVD88)	50% (feet, NAVD88)	95% (feet, NAVD88)
1	11.23	11.27	11.31
2	11.52	11.56	11.59
5	11.80	11.85	11.90
10	11.99	12.05	12.13
25	12.23	12.30	12.40
50	12.36	12.46	12.55
100	12.47	12.58	12.70
250	12.58	12.72	12.87
500	12.64	12.80	12.95
1000	12.65	12.83	12.98

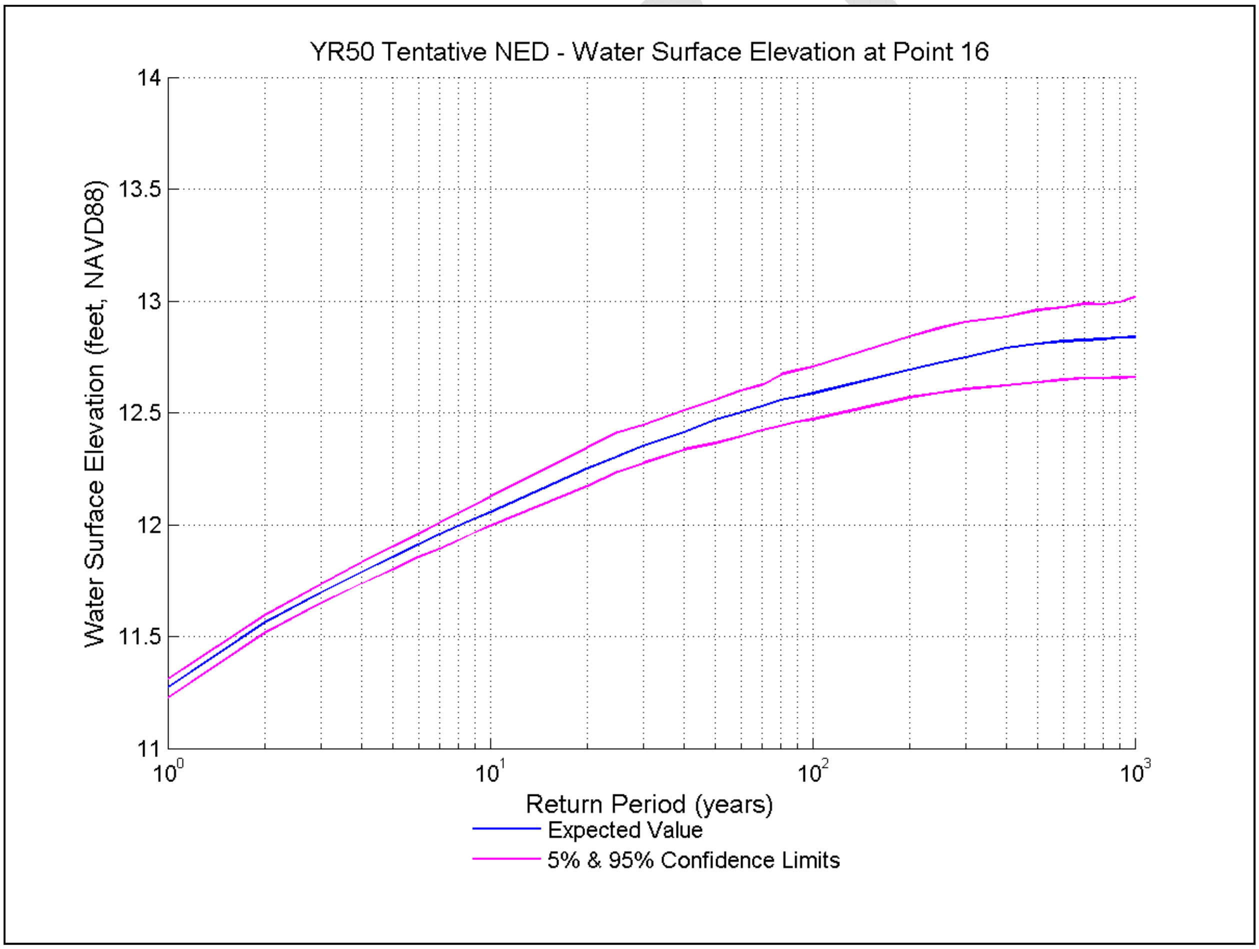


Figure 3-10. Flood stage frequency at Point 16 of Yr-50 Tentative NED alignment

Table 3-10. Flood stage frequency at Point 16 of Yr-50 Tentative NED alignment

Return Period (yrs)	5% (feet, NAVD88)	50% (feet, NAVD88)	95% (feet, NAVD88)
1	11.23	11.27	11.31
2	11.52	11.56	11.59
5	11.80	11.86	11.90
10	12.00	12.06	12.13
25	12.23	12.31	12.41
50	12.37	12.47	12.56
100	12.47	12.59	12.70
250	12.59	12.73	12.88
500	12.64	12.81	12.96
1000	12.66	12.84	13.02

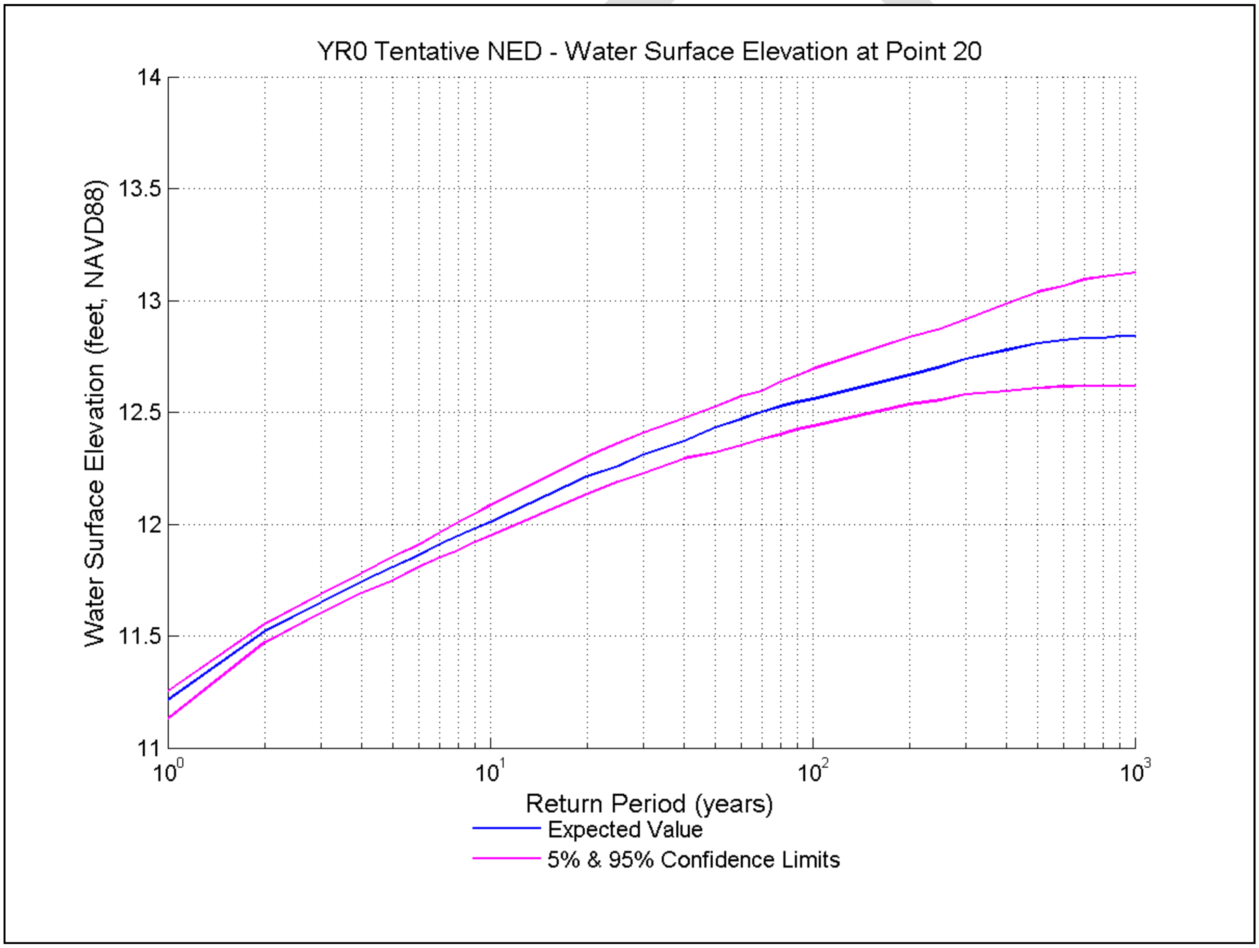


Figure 3-11. Flood stage frequency at Point 20 of Yr-50 Tentative NED alignment

Table 3-11. Flood stage frequency at Point 20 of Yr-50 Tentative NED alignment

Return Period (yrs)	5% (feet, NAVD88)	50% (feet, NAVD88)	95% (feet, NAVD88)
1	11.13	11.21	11.25
2	11.47	11.52	11.55
5	11.75	11.81	11.86
10	11.95	12.01	12.09
25	12.19	12.26	12.36
50	12.32	12.43	12.52
100	12.44	12.56	12.70
250	12.56	12.70	12.87
500	12.61	12.81	13.04
1000	12.62	12.84	13.13

3.2.3 Yr-0 LPA Results

Representative flood stage frequency results were presented at the outer levees, Point 3 and 7, in Figures and Tables 3-12 through 3-13 for the Yr-0 condition. Flood stage frequency results were also presented fort points 13, 14 and 16 along the inner levees for the Yr-0 condition as shown in Figures and Tables 3-14 to 3-16). All the statistical results presented include 5%, 50%, and 95% confidence limits.

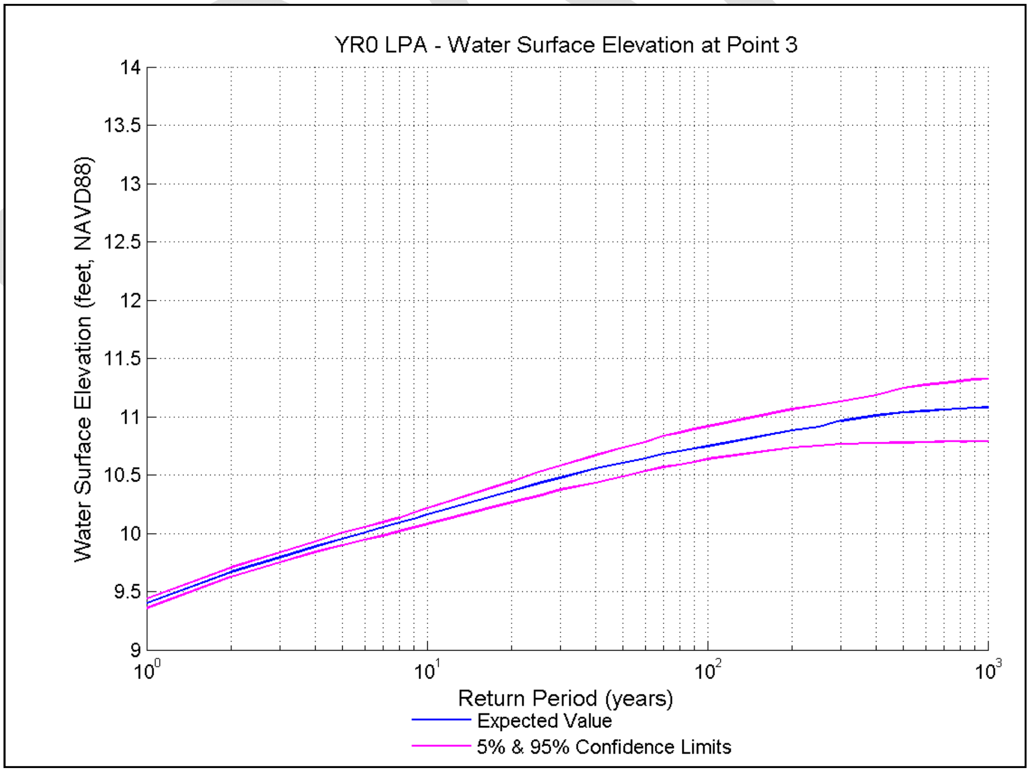


Figure 3-12. Flood stage frequency at point 3 of Yr-0 LPA alignment

Table 3-12. Flood stage frequency at point 3 of Yr-0 LPA alignment

Return Period (yrs)	5% (feet, NAVD88)	50% (feet, NAVD88)	95% (feet, NAVD88)
1	9.36	9.40	9.44
2	9.63	9.67	9.71
5	9.90	9.95	10.00
10	10.08	10.16	10.22
25	10.32	10.43	10.53
50	10.49	10.60	10.73
100	10.64	10.75	10.91
250	10.75	10.91	11.10
500	10.78	11.04	11.24
1000	10.79	11.08	11.32

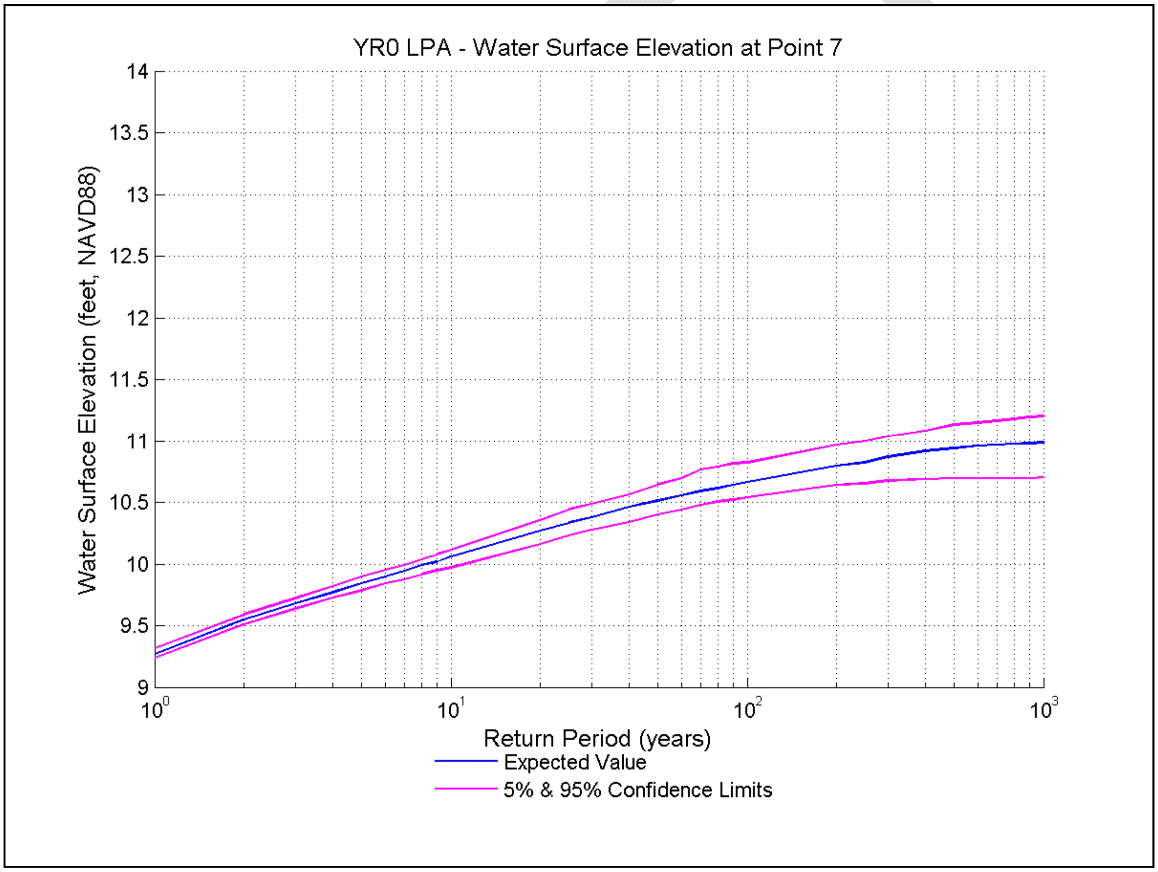


Figure 3-13. Flood stage frequency at Point 7 of Yr-0 LPA alignment

Table 3-13. Flood stage frequency at Point 7 of Yr-0 LPA alignment

Return Period (yrs)	5% (feet, NAVD88)	50% (feet, NAVD88)	95% (feet, NAVD88)
1	9.24	9.27	9.31
2	9.51	9.55	9.59
5	9.79	9.84	9.90
10	9.97	10.06	10.11
25	10.23	10.33	10.45
50	10.40	10.52	10.65
100	10.54	10.66	10.83
250	10.65	10.82	11.00
500	10.70	10.94	11.14
1000	10.71	10.98	11.21

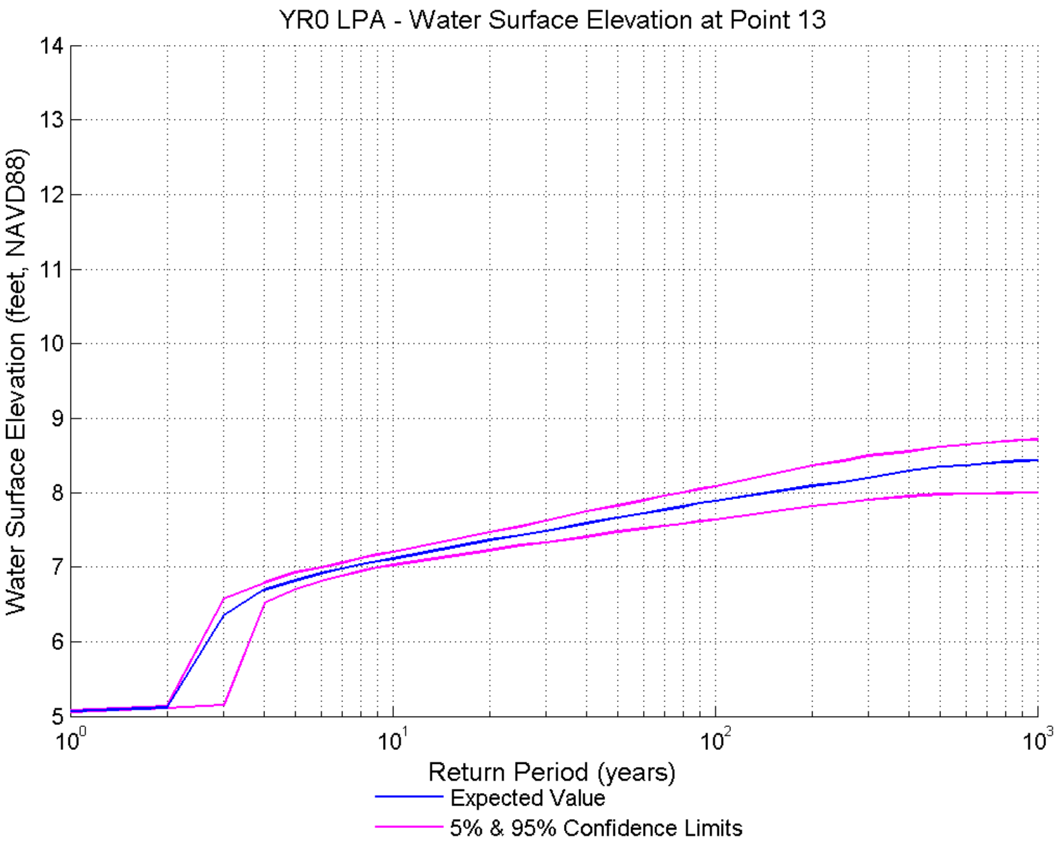


Figure 3-14. Flood stage frequency at point 13 of Yr-0 LPA alignment

Table 3-14. Flood stage frequency at point 13 of Yr-0 LPA alignment

Return Period (yrs)	5% (feet, NAVD88)	50% (feet, NAVD88)	95% (feet, NAVD88)
1	5.06	5.07	5.08
2	5.11	5.12	5.13
5	6.70	6.82	6.93
10	7.03	7.11	7.20
25	7.29	7.43	7.54
50	7.47	7.66	7.82
100	7.63	7.88	8.08
250	7.86	8.13	8.42
500	7.97	8.34	8.61
1000	8.00	8.43	8.71

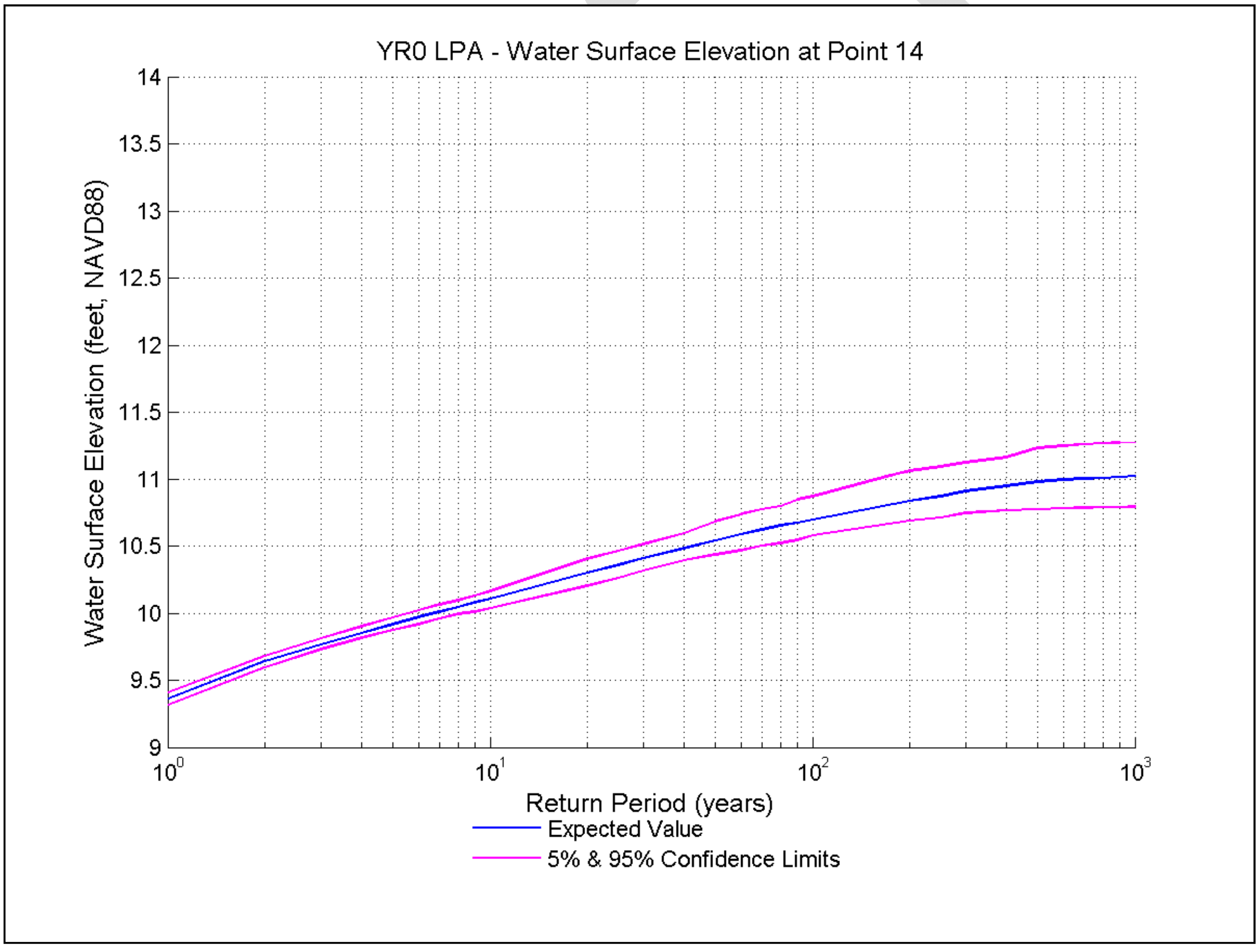


Figure 3-15. Flood stage frequency at Point 14 of Yr-0 LPA alignment

Table 3-15. Flood stage frequency at Point 14 of Yr-0 LPA alignment

Return Period (yrs)	5% (feet, NAVD88)	50% (feet, NAVD88)	95% (feet, NAVD88)
1	9.31	9.36	9.41
2	9.60	9.64	9.68
5	9.88	9.92	9.97
10	10.04	10.11	10.16
25	10.26	10.36	10.47
50	10.44	10.54	10.68
100	10.58	10.70	10.87
250	10.71	10.87	11.09
500	10.78	10.98	11.23
1000	10.79	11.02	11.28

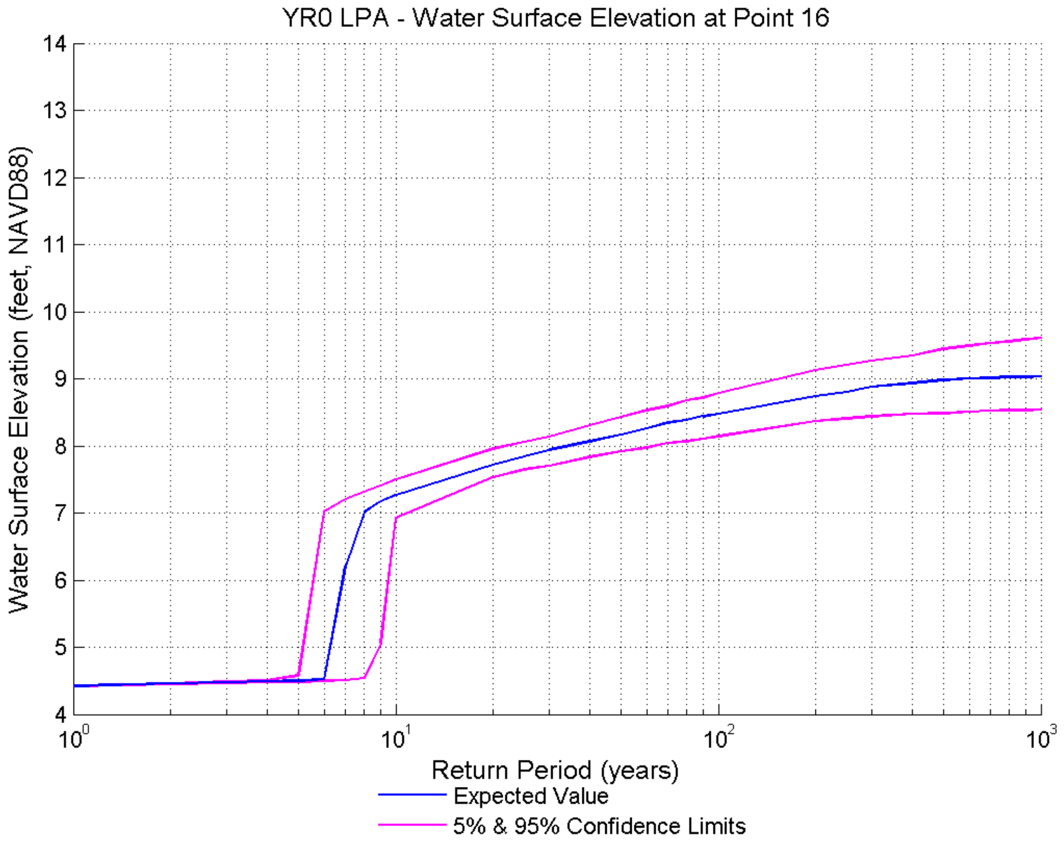


Figure 3-16. Flood stage frequency at Point 16 of Yr-0 LPA alignment

Table 3-16. Flood stage frequency at point 16 of Yr-0 LPA alignment

Return Period (yrs)	5% (feet, NAVD88)	50% (feet, NAVD88)	95% (feet, NAVD88)
1	4.42	4.42	4.43
2	4.45	4.46	4.46
5	4.49	4.50	4.58
10	6.93	7.26	7.50
25	7.65	7.84	8.06
50	7.92	8.16	8.43
100	8.14	8.48	8.79
250	8.41	8.80	9.20
500	8.48	8.98	9.44
1000	8.54	9.04	9.61

3.2.4 Yr-50 LPA Results

Representative flood stage frequency results were presented at the outer levees, Point 3 and 7, in Figures and Tables 3-17 through 3-18 for the Yr-50 condition. Flood stage frequency results were also derived Points 13, 14 and 16 along the inner levees for the Yr-50 condition as presented in Figures and Tables 3-19 through 3-21. All the statistical results presented include 5%, 50%, and 95% confidence limits.

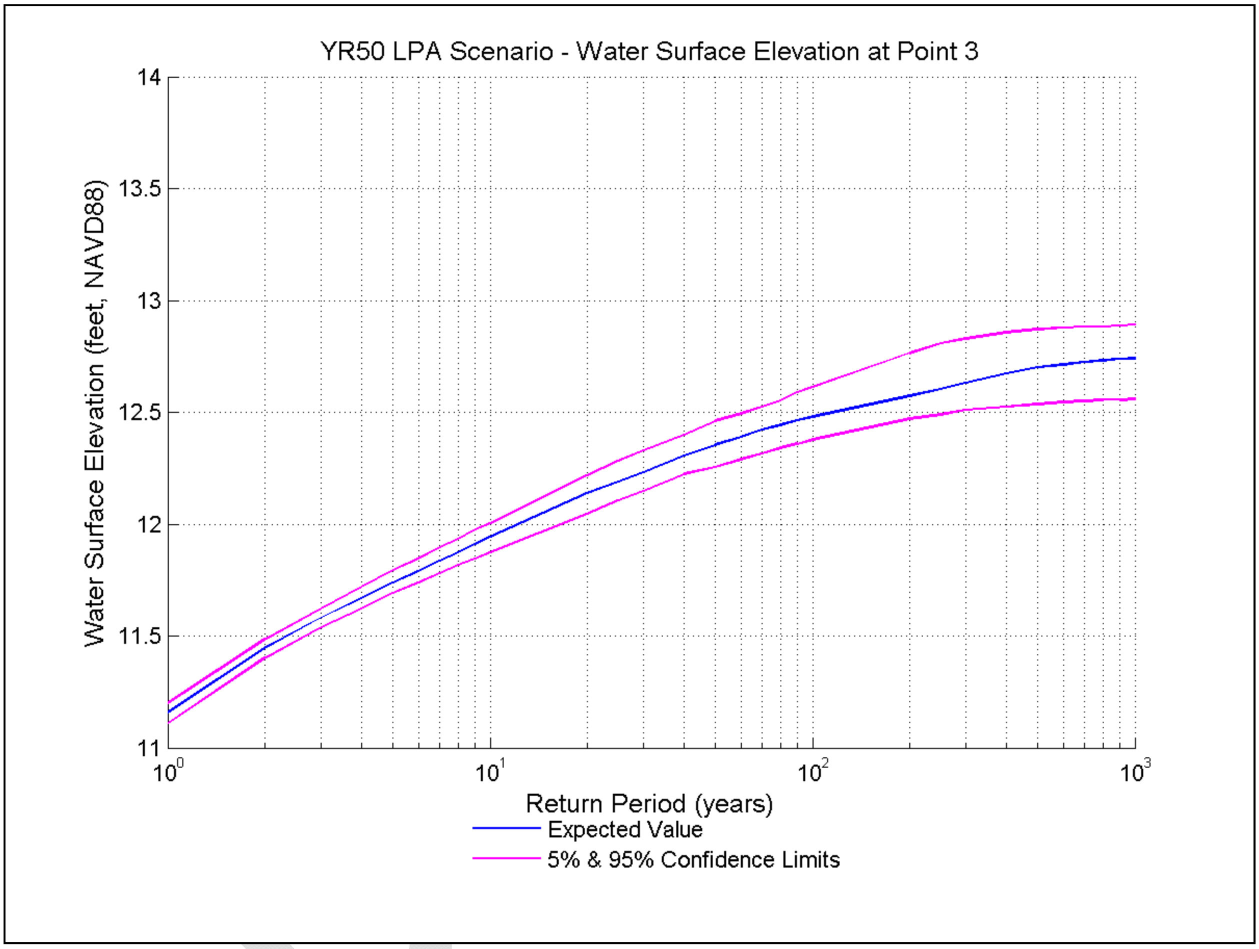


Figure 3-17. Flood stage frequency at Point 3 of Yr-50 LPA alignment

Table 3-17. Flood stage frequency at Point 3 of Yr-50 LPA alignment

Return Period (yrs)	5% (feet, NAVD88)	50% (feet, NAVD88)	95% (feet, NAVD88)
1	11.11	11.16	11.20
2	11.40	11.45	11.48
5	11.69	11.74	11.80
10	11.88	11.94	12.00
25	12.11	12.19	12.29
50	12.26	12.36	12.46
100	12.38	12.48	12.62
250	12.49	12.60	12.81
500	12.54	12.70	12.87
1000	12.56	12.74	12.89

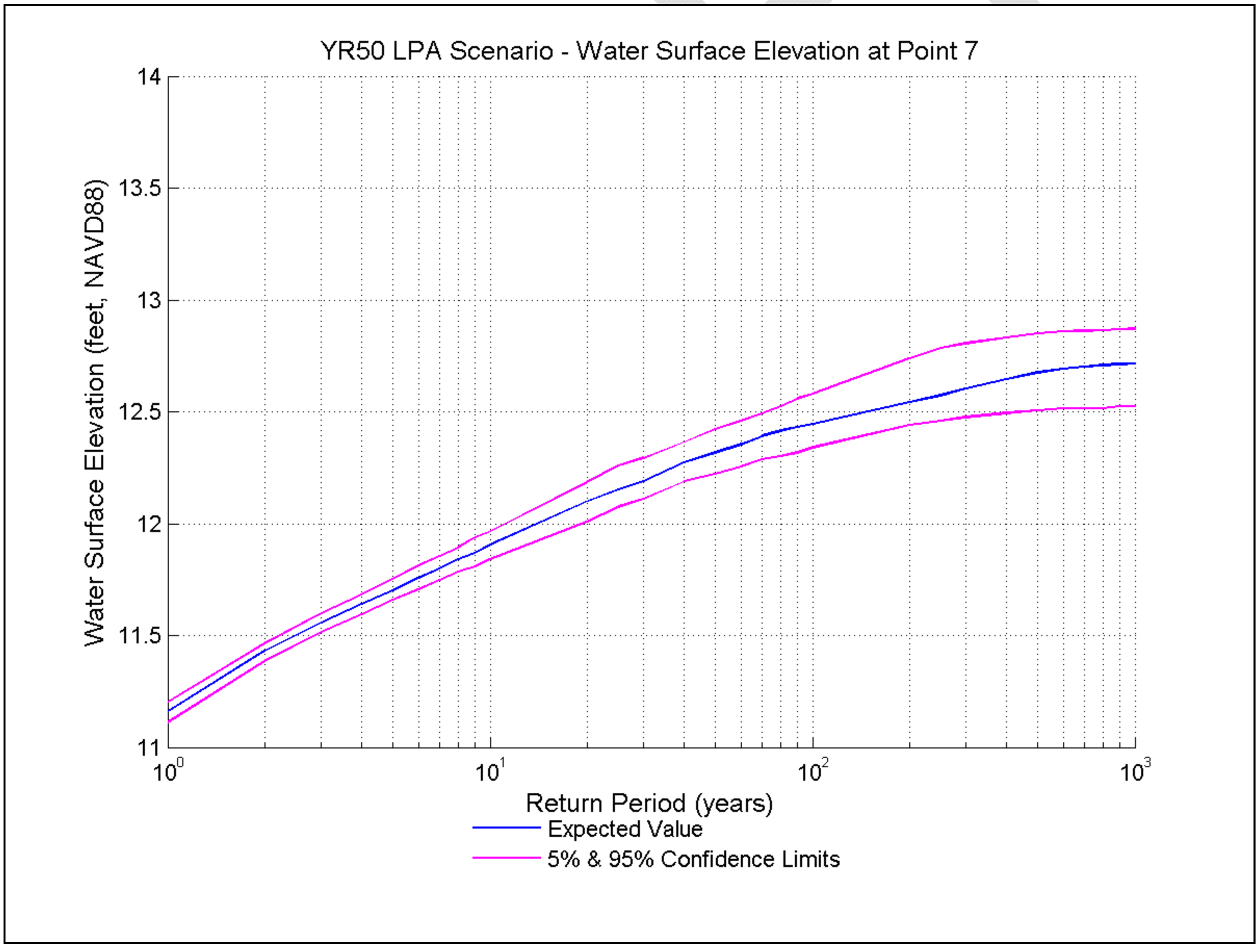


Figure 3-18. Flood stage frequency at Point 7 of Yr-50 LPA alignment

Table 3-18. Flood stage frequency at Point 7 of Yr-50 LPA alignment

Return Period (yrs)	5% (feet, NAVD88)	50% (feet, NAVD88)	95% (feet, NAVD88)
1	11.11	11.16	11.20
2	11.39	11.43	11.47
5	11.66	11.70	11.75
10	11.84	11.91	11.97
25	12.07	12.15	12.26
50	12.22	12.32	12.42
100	12.34	12.45	12.58
250	12.46	12.57	12.79
500	12.50	12.68	12.85
1000	12.52	12.72	12.87

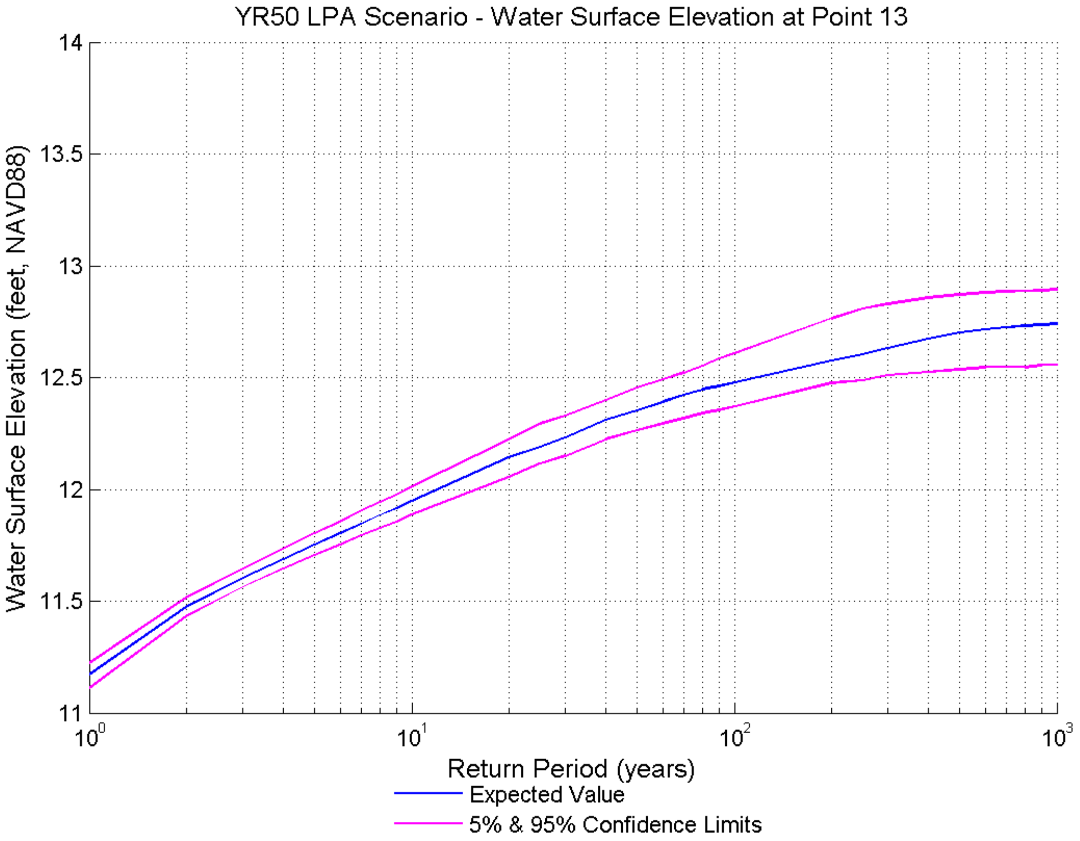


Figure 3-19. Flood stage frequency at Point 13 of Yr-50 LPA alignment

Table 3-19. Flood stage frequency at point 13 of Yr-50 LPA alignment

Return Period (yrs)	5% (feet, NAVD88)	50% (feet, NAVD88)	95% (feet, NAVD88)
1	11.11	11.17	11.22
2	11.43	11.48	11.52
5	11.71	11.75	11.80
10	11.89	11.95	12.02
25	12.12	12.19	12.30
50	12.27	12.35	12.46
100	12.37	12.48	12.61
250	12.49	12.60	12.81
500	12.54	12.70	12.87
1000	12.56	12.74	12.89

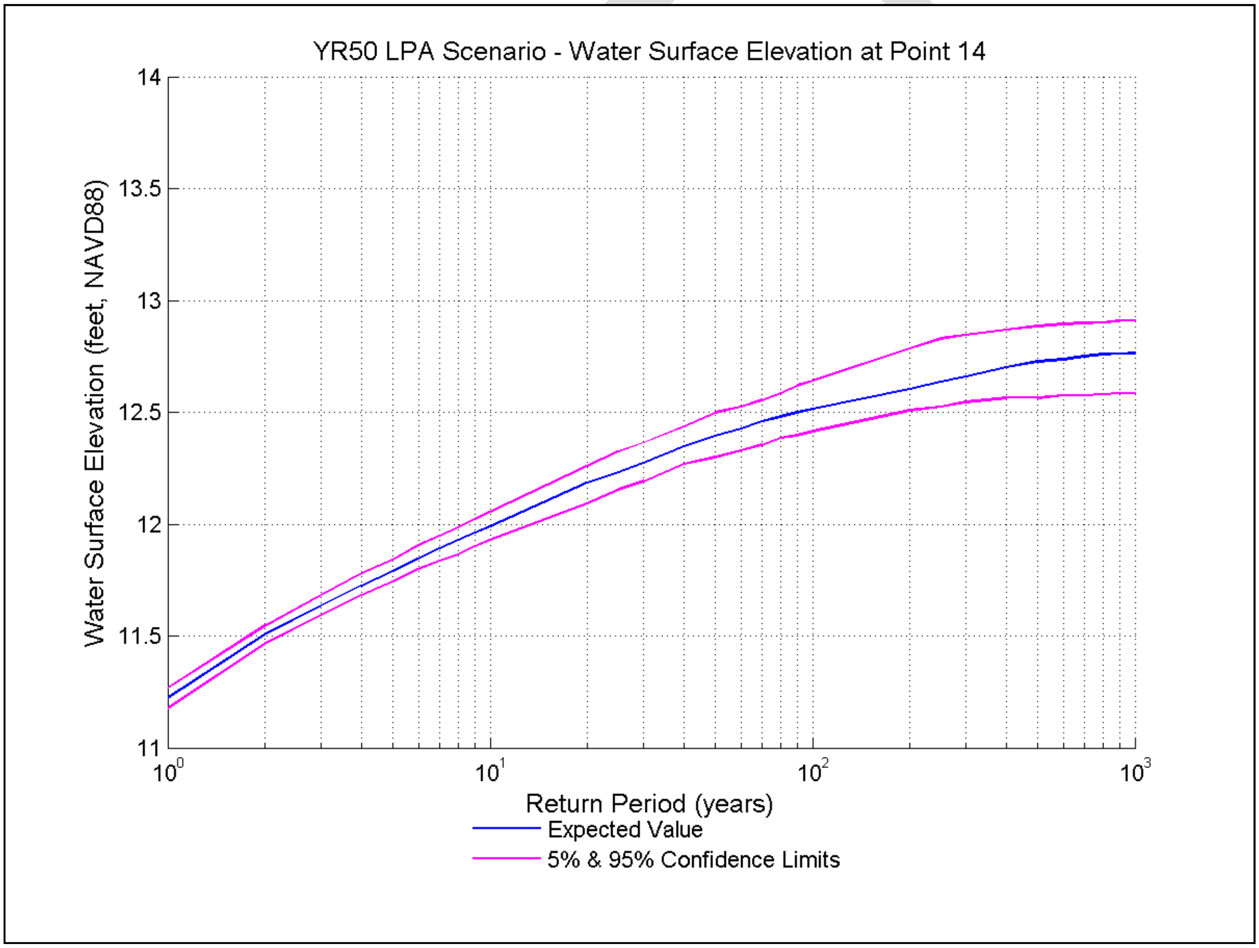


Figure 3-20. Flood stage frequency at Point 14 of Yr-50 LPA alignment

Table 3-20. Flood stage frequency at Point 14 of Yr-50 LPA alignment

Return Period (yrs)	5% (feet, NAVD88)	50% (feet, NAVD88)	95% (feet, NAVD88)
1	11.18	11.22	11.27
2	11.46	11.51	11.55
5	11.75	11.79	11.84
10	11.93	11.99	12.06
25	12.16	12.23	12.32
50	12.30	12.40	12.50
100	12.41	12.52	12.64
250	12.52	12.64	12.83
500	12.57	12.73	12.89
1000	12.59	12.77	12.91

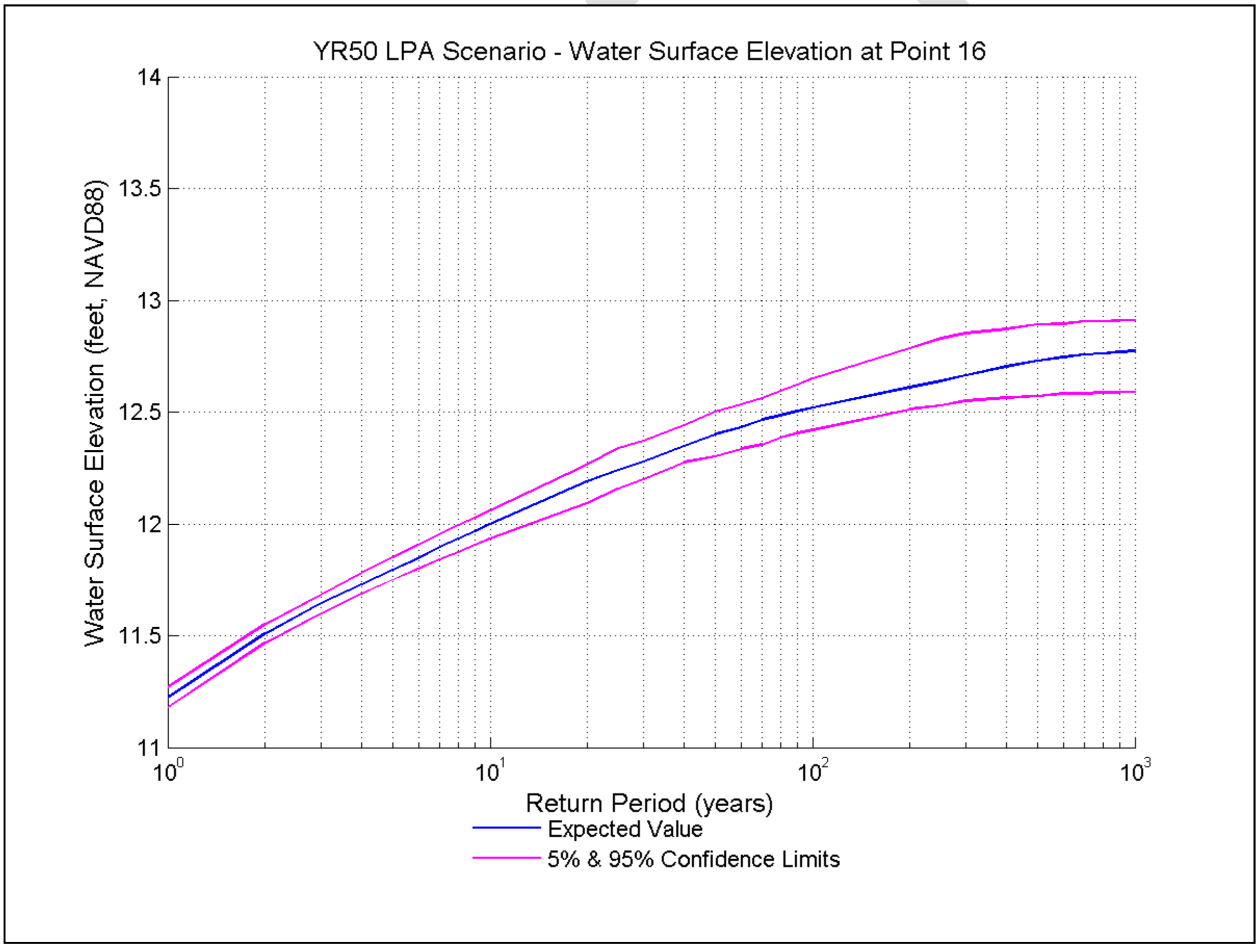


Figure 3-21. Flood stage frequency at Point 16 of Yr-50 LPA alignment

Table 3-21. Flood stage frequency at point 16 of Yr-50 LPA alignment

Return Period (yrs)	5% (feet, NAVD88)	50% (feet, NAVD88)	95% (feet, NAVD88)
1	11.18	11.22	11.27
2	11.47	11.51	11.55
5	11.75	11.80	11.85
10	11.93	12.00	12.06
25	12.16	12.24	12.34
50	12.30	12.40	12.50
100	12.42	12.52	12.65
250	12.53	12.64	12.83
500	12.57	12.73	12.89
1000	12.59	12.77	12.91

4.0 Conclusions

Reasonable flood stage frequency curves with uncertainties estimated were obtained by MCS method under all the levee layouts studied. Flood stage frequencies for both the LPA and NED levee layouts are very similar. It can be concluded that the technical approaches developed, using hydrodynamic and Monte Carlo simulations, provide a reasonable way for the establishment of coastal flood stage frequency at the project site.

5.0 References

Andes L. & Wu F., 2012,"Satistical Analysis Report – South San Francisco Bay Shoreline Study", prepared by the Corps of Engineers, San Francisco District, March 2012.

Dean, R.G. et al., 2010. "Erosional Equivalences of Levees: Steady and Intermittent Wave Overtopping", Ocean Engineering 37 Pages 104-113.

Dean, R.G. & Ledden M. V., 2010. "Accounting for Levee Overtopping Duration: A test with Hurricane Katrina Conditions", Coastal Engineering 2010.

Delta Modeling Associates, 2012. "South San Francisco Bay Long Wave Modeling Report", May 8, 2012.

Hubel B., 2012. "Memorandum of Record: Summary of Static Levee Failure Logic", dated April 10, 2012.

Hughes, S. A. & Nadal N.C., 2009. "Laboratory Study of Combined Wave Overtopping and Storm Surge Overflow of A Levee", Coastal Engineering 56, pp 244-259.

Hughes, S. A., 2007. “Estimation of Overtopping Flow Velocities on Earth Levees Due To Irregular Waves’, Coastal and Hydraulics Laboratory Engineering Technical Notes ERDC/CHL

CHETN-III-78 Vicksburg, MS: U.S. Army Engineering Research and Development Center.

Lee, L. T., jr., 2009a. “Reliability Assessment of San Francisco South Bay Salt Pond Outboard Levees’, prepared by U.S. Army, Engineering Research and Development Center.

Lee, L. T., jr., 2009b. “Reliability Assessment of San Francisco South Bay Salt Pond Inboard Levees’, prepared by U.S. Army, Engineering Research and Development Center.

Noble Consultants Inc. (NCI), 2012a, “Project Memorandum- Comparison of ERDC and Dean Methods for Levee Failure Assessment”, January 28, 2012.

Noble Consultants Inc. (NCI), 2012b, “Project Memorandum- Computation Procedure of Monte Carlo Simulation for South San Francisco Bay Shoreline Study”, May 1, 2012.

Snyder, B., 2012. Personal communication. US Army Corps of Engineers, San Francisco District.

APPENDIX A –COMPUTER CODE OF MONTE CARLO SIMULATION

```
C ****************************************************************************
C
C   A MCS program to perform F4 analysis for South Bay Shoreline Study
c   from IMSL (RNUN) and Intrinsic function (Ramdom_number).
C   It generates five sets of random numbers respectively corresponding to
C   Nubmer of storms per yr, preodicted and residual tides, and wind
c   direction and speed.
c   It determines flooding conditions in basins via various lookup tables
C   long waves, short waves, riverine discharge, etc.
C   date: 5/25/12 updated         By: CCL
c   For year 0 to check outer levee failure and link within various ponds
C ****************************************************************************

       PROGRAM Monte Carlo Simulations
      common/levee/ucrit,ework,xmout
      integer nr,nreal,nyr
      integer nstm,nstminc,npre,nres,nlo,nwdir,ndirm1,nwspd
      integer ncase,nstao,nstai,nprel,nresl,nspdl,nbin,nbreak
      integer nfree,nriv,nswl,nq,nbinc,nwse,ncstb
      Parameter (nr=2000, nreal=400, nyr=500)
      parameter (nstminc=11,npre=16,nres=15)
      parameter (nlo=2,nwdir=3,ndirm1=2,nwspd=9,nspdl=4)
      parameter (nprel=4,nresl=3,nstao=6,ncase=2,nstai=6,nbin=4)
      parameter (ncstb=3,nfree=4,nriv=2,nbreak=1,nswl=2,nq=8)
      Parameter (nwse=7,nbinc=21)
      real rstm(nr),rpre(nr),rres(nr),rwdir(nr),rwspd(nr),rcwse(nr)
      REAL cdfstm(nstminc),cdfpre(npre),zprey(npre),wspd(nlo),rcsf(nr)
      reaL cdfres(nres),zresy(nres),cdfwdir(nlo,nwdir),wdir1d(nwdir)
      real cdfwspd(nlo,ndirm1,nwspd),wspdy(nwspd),cdfwd1d(nwspd)
      real
zprel(nprel),zresl(nresl),volb(nbinc),elvb(nbinc),spdl(nspdl)
      real WSE4Dout(nstao,nprel,nresl)
      real WSE4Din(nstai,ncase,nprel,nresl)
      real WSExyo(nstao,nwdir,nspdl,nbin),fldelv(nr)
      real wsexyi(nstai,ncase,nwdir,nspdl,nbin),wtsout(ndirm1,nspdl)
      real wseout(nstao),wsein(nstai),whsout(ndirm1,nspdl)
      real whsin(nstai,ndirm1,nspdl),wtsin(nstai,ndirm1,nspdl)
      real cstout(nstao),freeb(nstao,nfree),probf(nstao,nfree)
      real stwse(nfree),probf1d(nfree),volw(nstai),RC(nwse),qwave(nwse)
      real cstin(nstai),cstlen(nstai),cwse(nstao),cstinb(ncstb)
      real WSE2d(nprel,nresl),whs1d(nspdl),wts1d(nspdl)
      real WSEnorm(nwse),twse(nwse),Hs(nwse),Ts(nwse),thour(nwse)
      real qriv(nriv,nq),zswl(nwse),Qbreak(nriv,nbreak,nswl,nq)
      real qriv1d(nq),qbreak2d(nswl,nq),volbk(nriv),volsum(nr)
      real prelow,preup,reslow,resup,xmout,xmin,cstoutm,volwave,volriv
      real zpre,zres,wsewo,wsewi,zc,ucrit,ework,vol

      INTEGER iseed,nstmy(nstminc),Iwdir(nwdir),idwdir(nlo),nweir(nriv)
```

```
      INTEGER IK,IL,IS,IWD,ISP,itid,ifail,ifailo(nstao),icase(nstai)
      INTEGER MM,J,K,MK,NN,KC,ikw,ikwm1,ir,icount,ictspd
      Data Wsenorm/0.37,0.68,0.91,1.0,0.96,0.77,0.56/
       OPEN (5, FILE='CDF.dat')
       OPEN (6, FILE='WSE_Wave.dat')
       OPEN (7,FILE='river_basin.dat')
      open (8,file='input_file.dat')
      open (9,file='output_check.dat')
      open (10, file='simulation.dat')
      open (11,file='WSEout.dat')
      open (12,file='wsein.dat')
110    format(f8.2,i5)
120   format(2f8.4)
130   format(8f9.2)
140    format(f6.2,8e10.2)
150    format(6i5)
160   format(i4,7f8.2)

!     ***   read cdf functions for 5 parameters ***
!      read CDF function for number of storm per year
!

       WRITE(8,*) 'CDF FOR Numbe of storms'
      DO J=1,nstminc
      READ (5,*) cdfstm(J),nstmy(J)
      WRITE (8,110) cdfstm(J),nstmy(J)
      END DO

!      read CDF function for predicted tides

      WRITE(8,*) 'CDF FOR predicted TIDES'
      DO J=1,Npre
      READ (5,*) cdfpre(J),zprey(J)
      WRITE (8,120) cdfpre(J),zprey(J)
      END DO

!     read CDF function for residual tides

      WRITE(8,*) 'CDF FOR residual TIDES'
      DO J=1,Nres
      READ (5,*) cdfres(J),zresy(J)
      WRITE (8,120) cdfres(J),zresy(J)
      END DO

!     read CDF function for wind direction
!     it has two locations, SFo (1)& MFF (2)
!     SFO data used for wind setup, MFF for short waves
!     wind data locations & wind direction
      WRITE(8,*) 'CDF FOR wind directions (2D)'
      Do i=1,nlo
        DO J=1,Nwdir
```

```
        READ (5,*) cdfwdir(i,J),Iwdir(J)
        WRITE (8,*) cdfwdir(i,J),Iwdir(J)
        END DO
      end do

!     read CDF function for wind speed
!     wind data locations (nlo=2), wind dirtion, & wind speed
      WRITE(8,*) 'CDF FOR wind speeds (3D)'
      do k=1,nlo
       Do i=1,ndirm1 ! two directions-the nonfactor 3rd
        DO J=1,Nwspd
        READ (5,*) cdfwspd(k,I,J),wspdy(J)
        WRITE (8,120) cdfwspd(k,I,J),wspdy(J)
        END DO
       END DO
      end do
      Write(*,*) 'end of CDFs reading'

!    ******************outer levee **********************
!     read lookup table for long waves
!     cases,locations,predicted and residual tides
!     two cases for levee non-breach and breached conditions
      WRITE(8,*) 'LOOKUP TABLE FOR LONG WAVES (4D)'

      read(6,*)(zprel(im),im=1,nprel)
      write(8,130)(zprel(im),im=1,nprel)
      read(6,*) (zresl(im),im=1,nresl)
       write(8,130) (zresl(im),im=1,nresl)
      write(8,*) 'hydrodynamics WSE'
      do ik=1,nstao
      write(8,*)'outer station ID',ik
        do im=1,nprel
        read(6,*)(wse4Dout(ik,im,in), in=1,nresl)
        write(8,130) (wse4Dout(ik,im,in), in=1,nresl)
        end do
      end do

!     read lookup table for wind setup
!     location,predicted, residual & wind direction, speed
      WRITE(*,*) 'LOOKUP TABLE FOR WIND-INDUCED SETUP-out (4D)'
!     read values of predicted & residual tides used for interpolation
!     only 4 combined (pre and res) bins for interpolation
      read(6,*) prelow,preup,reslow,resup
      write(8,*) 'prelow', prelow,preup,reslow,resup
      read(6,*) (spdl(i),i=1,nspdl)
      write(8,*) 'wind setup component'
      do il=1,nstao
      write(8,*)'outer station ID',il
       DO IWD=1, ndirm1  ! only two directions
        do isp=1,nspdl    ! 4 wind speeds
       read(6,*) (wsexyo(il,iwd,isp,j),j=1,nbin)
```

```
       write(8,130) (wsexyo(il,iwd,isp,j),j=1,nbin)
        end do
       END DO
      end do
      write(8,*) 'wind wave component only for station 1'
!     read lookup table for wind waves @ outer levee
!     only one general location exposed to local waves
      Write(*,*) 'Lookup Table for wind waves @outer levee (2D)'
!     read wave height only at one outer levee, no contribution for
others
      DO IWD=1,ndirm1   ! two directions
        do isp=1,nspdl
        read (6,*) Whsout(iwd,isp),wtsout(iwd,isp)
      write(8,120)Whsout(iwd,isp),wtsout(iwd,isp)
        end do
      End Do

!      ******** inner levee *********
!     read lookup table for long waves (only one station @ inner levee)
      WRITE(8,*) 'LOOKUP TABLE FOR LONG WAVES @ inner levee (4D)'

      do ik=1,nstai
      write(8,*)'inner station ID',ik
       DO IL=1,ncase
      write(8,*) 'case ID',il
        do im=1,nprel
        read(6,*)(wse4din(ik,il,im,in),in=1,nresl)
       write(8,130)(wse4din(ik,il,im,in),in=1,nresl)
        end do
       END DO
      end do

!     read lookup table for wind setup (multiple points)
      WRITE(8,*) 'LOOKUP TABLE FOR WIND-INDUCED SETUP- in (5D)'
       do ik=1,nstai
      write(8,*)'inner station ID',ik
       do im=1,ncase
      write(8,*) 'case ID',im
        DO IWD=1, ndirm1
           do isp=1,nspdl
         read(6,*) (wsexyi(ik,im,iwd,isp,in),in=1,nbin)
         write(8,130) (wsexyi(ik,im,iwd,isp,in),in=1,nbin)
           end do
      END DO
       end do
      end do

!     read wave height @inner levee (3D)
      write(8,*)'wind wave component'
      do iw=1,nstai
      write(8,*)'inner station ID',iw
```

```
       DO IWD=1,ndirm1    ! only two directions
        do isp=1,nspdl
        read (6,*) wHsin(iw,iwd,isp),WTsin(iw,iwd,isp)
        write (8,120) wHsin(iw,iwd,isp),WTsin(iw,iwd,isp)
        end do
       END DO
      end do
      write(*,*)'end of long & short wave reading'

!     *************Levee info **********
!     input outer levee information
      read (7,*)xmout,(cstout(i),i=1,nstao),ucrit,ework
      write(8,*) 'xmout,ucrit,ework',xmout,ucrit,ework
      write(8,130)(cstout(i),i=1,nstao)
!     read freeboard and probability of failure
      write(8,*) 'probability of static failure'
      do i=1,nstao
      write(8,*) 'nstao ID',i
       do j=1,nfree
       read(7,*)freeb(i,j),probf(i,j)
       write(8,*) freeb(i,j),probf(i,j)
       end do
      end do

!     input inner levee info
      read(7,*) xmin
      write(8,*)'xmin',xmin
       do i=1,nstai
       read(7,*)cstin(i),cstlen(i)
       write(8,130)cstin(i),cstlen(i)
       end do
!     read WSE overtop levee crest info
      read(7,*)(cstinb(i),i=1,ncstb)
      write(8,*)'cstinb',(cstinb(i),i=1,ncstb)

!     ******* River Q info ******
!     read river breakout flow (Guadalupe River & Coyote River)
      WRITE(*,*) 'Lookup table for river outbreak flow volume'
!     read number of break locations for each river

      read(7,*)(nweir(ii),ii=1,nriv)
      write(8,*)(nweir(ii),ii=1,nriv)

      DO idriv=1,nriv
!     read river q for 2 rivers each with different breakout locations
!     nq=8
      read(7,*) (qriv(idriv,iq),iq=1,nq)
      write(8,130)(qriv(idriv,iq),iq=1,nq)
!     note: if nweir()>1, nbreak=1 should be reassigned
       do iweir=1,nweir(idriv)
         DO IK=1,Nswl
```

```
        READ(7,*) zswl(ik),(Qbreak(idriv,iweir,ik,IM),IM=1,NQ)
        write(8,140) zswl(ik),(Qbreak(idriv,iweir,ik,IM),IM=1,NQ)
        END DO
       end do
      END DO
      write(*,*) 'end of river breakout reading'

!     ***** Volume vs Basin Elv *****
!      read lookup table for volume vs elevation
      write(*,*) 'read lookup table for volume vs elevation'
      Do ib=1,nbinc
      read(7,*) elvb(ib),volb(ib)
      write(8,140) elvb(ib),volb(ib)
      end do
      write(*,*) 'end of basin vol. vs elv. reading'
!     **************************************************

!     starting with a number of realization runs
      DO MK=1,nreal
      write(9,*)'realization= ',mk
      write(11,*)'realization= ',mk
      write(12,*)'realization= ',mk
!    Use the function listed in IMSL
!    It creates 7 sets of random number from a specified number (3000)
!

      CALL RNGET(ISEED)
      CALL RNSET(ISEED)
      CALL RNUN (nr, rstm)

      CALL RNGET(ISEED)
      CALL RNSET(ISEED)
      CALL RNUN(nr,rpre)

      CALL RNGET(ISEED)
      CALL RNSET(ISEED)
      CALL RNUN (nr, rres)

      CALL RNGET(ISEED)
      CALL RNSET(ISEED)
      CALL RNUN(nr,rwdir)

      CALL RNGET(ISEED)
      CALL RNSET(ISEED)
      CALL RNUN(nr,rwspd)

      CALL RNGET(ISEED)
      CALL RNSET(ISEED)
      CALL RNUN(nr,rcwse)

      CALL RNGET(ISEED)
```

```
      CALL RNSET(ISEED)
      CALL RNUN(nr,rcsf)

      write(*,*) 'end of random number generating'
!     ******************************************************
!
!     starting the "numbe of year" process, e.g., say 500yr
!
      icount=0
      ictspd=0
      Do NN=1,nyr

!     Determine the critical WSE in each year at stato for static
failure
      do i=1,nstao
!     convert 2D to 1D
       do j=1,nfree
       stwse(j)=cstout(i)-freeb(i,j)
       probf1d(j)=probf(i,j)
       write(9,*) j,stwse(j),probf1d(j)
       end do
      call interp(nfree,probf1d,stwse,rcwse(nn),cwse(i))
      write(9,*)'nstao',i,rcwse(nn),cwse(i)
      end do

!     Determine number of storm per year
      call interp_step(nstminc,cdfstm,nstmy,rstm(nn),nstm)
      write(9,*) 'nstm=',nstm,rstm(nn),'year=',nn

!     do loop of number of storms per year
      DO kc=1,nstm

      do il=1,nstao
         ifailo(il)=1
         end do

! no breach condition to begin
      Icount=icount+1
      write(*,*) 'icount=',icount

!     randomly select a predicted tide
      CALL INTERP(npre,cdfpre,zprey,rpre(icount),zpre)
!     randomly select a residual tide
      CALL INTERP(nres,cdfres,zresy,rres(icount),zres)
      write(9,*)'icount=',icount,zpre,zres
!     determine wind conditions for each storm event
!     randomly select a wind direction @ SFO & MFF
      do i=1,nlo
      wspd(i)=0.
      do j=1,nwdir
      wdir1d(j)=cdfwdir(i,j)
```

```
      end do
       CALL INTERP_step(nwdir,wdir1d,Iwdir,rwdir(icount),IDwdir(i))
      write(9,*) 'nlo&idwdir',i,idwdir(i),rwdir(icount)
      end do

      if(idwdir(1).lt.3.or.idwdir(2).lt.3) then
      ictspd=ictspd+1
      end if

      do i=1,nlo
      if(idwdir(i).eq.3) then
      goto 11
      end if

!     determine randomly selected wind speed @ SFO & MFF
!     wspd(1)=SFO & wspd(2)=MFF
      idk=idwdir(i)
      do im=1,nwspd
      cdfwd1d(im)=cdfwspd(i,idk,im)
      end do
      CALL INTERP(nwspd,cdfwd1d,wspdy,rwspd(ictspd),wspd(i))
!     write(9,*) 'rwspd()',rwspd(ictspd)
11    continue
      end do
      write(9,*)'wind speed',wspd(1),wspd(2)

!     BEGINING OF OUTER LEVEE ANALYSIS
!     do loop of various stations @ outer levee
      do il=1,nstao

!     convert 4D to 2D variables
!     use ncase=1 (no breach)
      do im=1,nprel
      do in=1,nresl
      wse2d(im,in)=wse4dout(il,im,in)
      end do
      end do

!     check lookup table to determine WSE @ outer levee
!     write(*,*) zpre,(zprel(i),i=1,nprel)
!     write(*,*) zres,(zresl(i),i=1,nresl)
       CALL INTERP_2D
(nprel,zprel,nresl,zresl,wse2d,zpre,zres,wsetideo)

      if(idwdir(1).eq.3) then
      write(*,*) 'no contribution from wind setup', Icount
      wseout(il)=wsetideo
      goto 12
      end if

!     determine wind setup from lookup tables
```

```
!     determine the wind speed range @SFO
!     write(9,*)'spdl',(spdl(i),i=1,nspdl)
      do i=2,nspdl
      if (wspd(1).le.spdl(i)) then
      ikw=i
      ikwm1=i-1
      goto 19
      end if
      end do
19    continue
      write(9,*)'ikw',wspd(1),spdl(ikw)
!     interpolate wind setup from 1st wind speed @SFO
      z11a=wsexyo(il,idwdir(1),ikwm1,1)
      z12a=wsexyo(il,idwdir(1),ikwm1,2)
      z21a=wsexyo(il,idwdir(1),ikwm1,3)
      z22a=wsexyo(il,idwdir(1),ikwm1,4)
      Z1a =((zpre-prelow)*z21a+(preup-zpre)*z11a)/(preup-prelow)
      Z2a =((zpre-prelow)*z22a+(preup-zpre)*z12a)/(preup-prelow)
      Zouta=((zres-reslow)*z2a+(resup-zres)*z1a)/(resup-reslow)
!     write(9,160)ikwm1,z11a,z12a,z21a,z22a,z1a,z2a,zouta
!     interpolate wind setup for 2nd wind speed @ SFO
      z11b=wsexyo(il,idwdir(1),ikw,1)
      z12b=wsexyo(il,idwdir(1),ikw,2)
      z21b=wsexyo(il,idwdir(1),ikw,3)
      z22b=wsexyo(il,idwdir(1),ikw,4)
      Z1b =((zpre-prelow)*z21b+(preup-zpre)*z11b)/(preup-prelow)
      Z2b =((zpre-prelow)*z22b+(preup-zpre)*z12b)/(preup-prelow)
      Zoutb=((zres-reslow)*z2b+(resup-zres)*z1b)/(resup-reslow)
!     write(9,160)ikw,z11b,z12b,z21b,z22b,z1b,z2b,zoutb
!     interpolate between 2 wind speeds @ SFO
      Wsewo=zouta+(zoutb-zouta)*(wspd(1)-spdl(ikwm1))/
     +(spdl(ikw)-spdl(ikwm1))
      WSEout(il)=wsetideo+wsewo ! the combined WSE

!     determine short wave @ outer levee
!     Convert Hs and Ts from 2D to 1D for only one outer station
!     use data @ Moffatfield
      if(idwdir(2).eq.3) goto 12 ! no wave contribution
      DO id=1,nspdl
      Whs1D(id)=Whsout(idwdir(2),id)
      WTs1D(id)=wtsout(idwdir(2),id)
      end do
      CALL INTERP(nspdl,spdl,whs1D,wspd(2),waveout)
      CALL INTERP(nspdl,spdl,wTs1d,wspd(2),Tsout)

12    continue
      write(9,*)'Wseout',wsetideo,wsewo,waveout,tsout
!     beginning of levee Static Failure Analysis
      if (cwse(il).eq.cstout(il).or.wseout(il).le.cwse(il))then
      ifailo(il)=1
      probbr=0.
```

```
      goto 13
      end if
      Probbr=(wseout(il)-cwse(il))/(cstout(il)-cwse(il))

      if (rcsf(icount).gt.probbr)then
      ifailo(il)=1
      else
      ifailo(il)=2
      write(*,*)'outer levee static failure'
      end if

13     continue
      write(9,*) 'cwse',cwse(il),cstout(il)
      write(9,*)'static failure',ifailo(il),probbr,rcsf(icount)
!     if(idwdir(2).eq.3) then
!     goto 14    ! no waves no dynamic failure
!     end if

!     Check levee failure, based on Dean's erosion work method
!     convert english unit to metric unit for analysis to determine
!     whether failure of outer levee occurs

      if(il.gt.1.or.idwdir(2).eq.3) goto 14 ! no wave conditions
      Do i=1,nwse
      twse(i)=wseout(il)*wsenorm(i)*0.3048
      Hs(i)=waveout*0.3048 ! covnert to the meteric system
      Ts(i)=Tsout
      thour(i)=real(i)
       end do
      cstoutm=cstout(il)*0.3048
      write(9,*)'dynamic failure check'
      CALL LeveeFail(cstoutm,nwse,thour,twse,Hs,Ts,Ifailo(il))

      if (ifailo(il).EQ.2) then
       write(*,*) 'outer Levee dynamic failure'
      end if
14    continue
      write(*,*) 'WSE level including wind setup determined',
wseout(il)
      end do ! end of number of outer stations (il)
      Write(9,*) 'complete levee analysis',(ifailo(i),i=1,nstao)

!     *******************************
!     BEGINING OF INNER LEVEE ANALYSIS

!     determine the correponding icase() for inner levee locations
!     there are 6 outer breaching locations correponding to 4 inner
stations
      do i=1,nstai
      icase(i)=1
      wsein(i)=0.
```

```
      qwave(i)=0.
      end do
!     any breach of Pt. 7, 6 & 5 corres to inner levee Pt.11&12
      do i=1,3
      if(ifailo(i).eq.2) then
      icase(1)=2
      icase(2)=2
      goto 18
      end if
      end do
18    continue

!     outer pt.4 corres to inner Pt.13
      if(ifailo(4).eq.2) then
      icase(3)=2
      end if

!     outer locations 4, 5 & 6 corres. to inner locations pt 14,16 & 23

      do i=4,6
      if (ifailo(i).eq.2) then
      icase(i)=2
      end if
      end do


!     ONLY CONSIDER WAVE OVERTOPPING WITHOUT LEVEE FAILURE
!     read inner data for either icase=1 (no breached)or 2 (breached)
      volwave=0.

      do il=1,nstai
      write(*,*) 'inner levee'
!     determine if WSE @ Pt 11x,12 & 13 higher than the levee crest
elev
!     if so, then icase(6)=2
      if(il.le.5) goto 15
      do i=1,ncstb
      write(9,*) 'cstinb',wsein(i),cstinb(i)
      if(il.eq.nstai.and.wsein(i).gt.cstinb(i)) then
      icase(nstai)=2
      go to 15
      end if
      end do
15    continue

!     convert 4D to 2D variables
      do im=1,nprel
      do in=1,nresl
      wse2d(im,in)=wse4din(il,icase(il),im,in)
      end do
      end do
```

```
!     check lookup table to determine WSE @ INNER levee

!     write(*,*) zpre,(zprel(i),i=1,nprel)
!     write(*,*) zres,(zresl(i),i=1,nresl)
      CALL INTERP_2D(nprel,zprel,nresl,zresl,wse2d,zpre,zres,wsetidei)
      write(9,*) 'Inner wse',zpre,zres,wsetidei,icase(il)

      if(idwdir(1).eq.3) then
      wsein(il)=wsetidei    ! no wind setup
      goto 16
      end if
      do i=2,nspdl
      if (wspd(1).le.spdl(i)) then
      ikw=i
      ikwm1=i-1
      goto 20
      end if
      end do
20    continue
!     determine wind setup from lookup tables (@SFO)
!     convert 5D wind speed (2 direction) into 1D
!     interpolate first wind speed @SFO
      z11a=wsexyi(il,icase(il),idwdir(1),ikwm1,1)
      z12a=wsexyi(il,icase(il),idwdir(1),ikwm1,2)
      z21a=wsexyi(il,icase(il),idwdir(1),ikwm1,3)
      z22a=wsexyi(il,icase(il),idwdir(1),ikwm1,4)
       Z1a =((zpre-prelow)*z21a+(preup-zpre)*z11a)/(preup-prelow)
       Z2a =((zpre-prelow)*z22a+(preup-zpre)*z12a)/(preup-prelow)
       Zina=((zres-reslow)*z2a+(resup-zres)*z1a)/(resup-reslow)
!     write(9,160)ikwm1,z11a,z12a,z21a,z22a,z1a,z2a,zina
!     interpolate second wind speed@SFO
      z11b=wsexyi(il,icase(il),idwdir(1),ikw,1)
      z12b=wsexyi(il,icase(il),idwdir(1),ikw,2)
      z21b=wsexyi(il,icase(il),idwdir(1),ikw,3)
      z22b=wsexyi(il,icase(il),idwdir(1),ikw,4)
       Z1b =((zpre-prelow)*z21b+(preup-zpre)*z11b)/(preup-prelow)
       Z2b =((zpre-prelow)*z22b+(preup-zpre)*z12b)/(preup-prelow)
       Zinb=((zres-reslow)*z2b+(resup-zres)*z1b)/(resup-reslow)
!     write(9,160)ikw,z11b,z12b,z21b,z22b,z1b,z2b,zinb
!     interpolate between 2 wind speeds
      Wsewi=zina+(zinb-zina)*(wspd(1)-spdl(ikwm1))/(spdl(ikw)-
spdl(ikwm1))
      WSEin(il)=wsetidei+wsewi
      WRITE(9,*) 'wind-setup',wsewi,wsein(il),WSPD(1)

!     no need to do computation for il=2 & 3 as it is not a inner levee
      if(il.eq.2.or.il.eq.3) then
      volw(il)=0.
      go to 21
      end if
```

```
!     determine short waves @ inner levee using data @ MFF
!     Convert Hs and Ts from 3D to 1D
      if(idwdir(2).eq.3) goto 16
      DO id=1,nspdl
      Whs1D(id)=Whsin(il,idwdir(2),id)
      WTs1D(id)=wtsin(il,idwdir(2),id)
      end do

      CALL INTERP(nspdl,spdl,whs1D,wspd(2),wavein)
      CALL INTERP(nspdl,spdl,wTs1d,wspd(2),Tsin)

!     Computer wave overtopping volume for three different conditions
!     wave only, surge only, and combined wave +surge

      DO i=1,nwse
      twse(i)=wsein(il)*wsenorm(i)
      Rc(i)= cstin(il)-twse(i)
!     write(*,*)'twes & rc',twse(i),rc(i)

!     compute irribarren number
      xirr=xmin/Sqrt(wavein/(5.12*Tsin**2))

!     F wave ovetopping only (Hughes, cf/sec/ft)
      IF (Rc(i).ge.0.0.and.xirr.le.2.0) then
      qwave(i)=0.06*Sqrt(32.2*wavein**3)*xirr/xmin*exp(-5.2*Rc(i)
     +     /(wavein*xirr))
      else If (Rc(i).ge.0.0.and.xirr.gt.2.0) then
      qwave(i)=0.2*Sqrt(32.2*wavein**3)*exp(-2.6*Rc(i)/wavein)
      end if

!     for the combined wave overtopping and surge flow from Hughes
      If(Rc(i).lt.0.) then
      qwave(i)=Sqrt(32.2*wavein**3)*(0.034+0.53*(-Rc(i)/wavein)**1.58)
      end if
      end do

      goto 17

16    continue

!     compute the surge overflow volume only (Hughes,cf/sec/ft)
      do j=1,nwse
      twse(j)=wsein(il)*wsenorm(j)
      Rc(j)= cstin(il)-twse(j)

      If(Rc(j).lt.0.) then
      qwave(j)=0.5443*sqrt(32.2)*ABS(rc(j))**(3./2.)
      else
      qwave(j)=0.
      end if
      end do
```

```
17    continue
!     compute the water volume over the entire length of cstlen
!     the time increment is one hour
      volw(il)=0.
      do i=1,nwse-1
      volw(il)=volw(il)+0.5*(qwave(i)+qwave(i+1))*3600.*cstlen(il)
      end do
21    volwave=volwave+volw(il)
      write(9,*) 'volwave',volwave,volw(il)
      end do ! end fo inner levee stations (il)

!     estimate river breakout volume
!     determine the river Q correlated to the residual tide @ bay end
!     1 for guadalupe River, 2 for Coyotee Creek

      volriv=0.
      write(11,160) Icount,(wseout(i),i=1,nstao)
      write(12,160) Icount,(wsein(i),i=1,nstai)
      do ir=1,nriv
      volbk(ir)=0.
      if (ir.eq.1) then
      qres=3160*zres
      else
      qres=1650.
      end if
      write(9,*) ir,qres
      DO j=1,nweir(ir)! check nweir(ir) = nbreak

       DO ij=1,nswl
      do k=1,nq
      qriv1d(k)=qriv(ir,k)
!     CONVERT 4D TO 2D
      qbreak2d(ij,k)=qbreak(ir,j,ij,k)
        end do
       end do

      call interp_2D(nswl,zswl,nq,qriv1d,qbreak2d,wseout(1),qres,vol)
      volbk(ir)=volbk(ir)+vol
      END DO     ! breakout do loop

!     WRITE(*,*) 'river break', nswl,(zswl(ii),ii=1,nswl)
!     write(*,*) nq,(qriv1d(ii),ii=1,nq)
      Volriv=volriv+volbk(ir)
      write(9,*) 'river=',ir,volbk(ir),volriv
      end do     ! river do loop

!     ************* Basin ********************
!     determine basin elevation due to water volume flow or overtop
into the basin
      Volsum(icount)=volriv+volwave
```

```
      Call interp (nbinc,volb,elvb,volsum(icount),flood)
      fldelv(icount)=flood
      write(9,*)'fldelv',icount,volriv,volwave,fldelv(icount)
      write(*,*)'complete anlysis for one inner station'

      write(*,*) 'obtain flood level for each event',fldelv(icount)
       END DO  ! number of storm do loop (kc)

      write(10,150) nn,nstm,icount
      Write(*,*)'end of individual year simulation',nn,icount
      end do ! end of number of year simulation (nn)

      write(10,150) mk,icount
      write(*,*) 'end of each realization',mk
      end do ! end of number of realization (mk)

       STOP
       END



!

C
************************************************************************
*****
C
C This subroutine does the interpretation to determine the randomly
selected
C variables such as SWE (storm water elv.), wind speed & wind
direction
C
C***********************************************************************
*******

      SUBROUTINE INTERP(Nxy,XSER,YSER,XIN,YOUT)
!     ************************************************
!    This subroutine interpretes the incremental values
!    from two discrete values
!     ***********************************************
!
       INTEGER Nxy, I
       REAL XSER(Nxy),YSER(Nxy),XIN,YOUT

       IF (XIN.LE.XSER(1))THEN
       YOUT=YSER(1)
       GOTO 20
       END IF

       IF(XIN.GT.XSER(Nxy)) THEN
       WRITE(9,*) 'INTERP XSER OUT OF RANGE'
```

```
        write(9,*) 'xin,Xser(nxy)',xin,xser(1),xser(nxy)
       YOUT=YSER(Nxy)
       GOTO 20
       END IF

       DO I=1,Nxy-1
       IF(XIN.GT.XSER(I).AND.XIN.LE.XSER(I+1)) THEN
       YOUT=YSER(I)+(XIN-XSER(I))/(XSER(I+1)-XSER(I))*(YSER(I+1)-
YSER(I))
       GOTO 20
      end if
       END DO

   20 CONTINUE
!       WRITE(*,*) YOUT
       RETURN
       END






!======================================================================
=====
!
       SUBROUTINE INTERP_2D(NX,XSER,NY,YSER,ZSER,XIN,YIN,ZOUT)
!
! This subroutine executeS a 2d interpolation process
! Two input series are (nx, xser) and (ny, yser)
! the interpolated series is zser (nx, ny)
! the parameter input are xin and yin
! the interpolated output is zout
! Use the weighted method

       INTEGER NX, NY,ISEL,JSEL,I,J
       REAL XIN,YIN,ZOUT
       REAL Z11, Z12,Z21,Z22,Z1,Z2
       REAL XSER(NX), YSER(NY),ZSER(NX,NY)

! FIND INCREMENTAL jsel FOR XSER
!     write(*,*)'xin=',xin
!     write(*,*)(xser(i),i=1,nx)
       IF (XIN.LT.XSER(1).OR.XIN.GT.XSER(NX)) THEN
       WRITE(9,*) 'INTERP_2d XSER OUT OF RANGE'
       STOP
       END IF

       DO I=1,NX-1
       IF (XIN.GE.XSER(I).AND.XIN.LT.XSER(I+1)) THEN
```

```
      ISEL=I
      GOTO 10
     end if
      END DO

! FIND INCREMENT jsel FOR YSER
10    CONTINUE

      IF (YIN.LT.YSER(1).OR.YIN.GT.YSER(NY)) THEN
      WRITE(9,*) 'INTER_2d YSER OUT OF RANGE'
      STOP
      END IF

      DO J=1,NY-1
      IF (YIN.GE.YSER(J).AND.YIN.LT.YSER(J+1)) THEN
      JSEL=J
      GOTO 20
     end if
      END DO
20    CONTINUE

! SELECT 4 VALUES OF isel, isel+1, jsel AND jsel+1 FOR INTERPOLATION

      z11=zser(isel,jsel)
      z12=zser(isel,jsel+1)
      z21=zser(isel+1,jsel)
      z22=zser(isel+1,jsel+1)

      Z1 =((Xin-xser(isel))*Z21+(Xser(isel+1)-Xin)*z11)/
     +     (xser(isel+1)-xser(isel))
      Z2 =((Xin-xser(isel))*z22+(xser(isel+1)-Xin)*Z12)/
     +     (xser(isel+1)-xser(isel))

      Zout=((Yin-yser(jsel))*z2+(yser(jsel+1)-yin)*z1)/
     +     (yser(jsel+1)-yser(jsel))
!    Write(9,*) isel,jsel,z11,z12,z21,z22, zout
      RETURN
      END


!
     SUBROUTINE InterP_step(Nxy,XSER,ISER,XIN,IOUT)
!    *********************************************
!    This subroutine select incremental values from
!    multiple steps function
!    *********************************************
      INTEGER Nxy, I,ISER(Nxy)
      REAL XSER(Nxy)

      IF (XIN.LE.XSER(1)) THEN
      IOUT=ISER(1)
```

```
      GOTO 20
      END IF

      IF(XIN.GT.XSER(Nxy)) THEN
      IOUT=ISER(Nxy)
      WRITE(9,*) "Interp_step YSER OUT OF RANGE"
      GOTO 20
      END IF

      DO I=1,Nxy-1
      IF(XIN.GT.XSER(I).AND.XIN.LE.XSER(I+1)) THEN
      IOUT=ISER(I+1)
      GOTO 20
     END IF
      END DO

   20 CONTINUE
      RETURN
      END



     Subroutine leveefail(zc,ITID,TTID,TID,HS,TS,IFAIL)

!  This program determines the erosion work due to wave overtopping
!  to determine whether a levee breched under storm wave attack
!  orignally programed by Dean on SEPTEMBER 11, 2010, modified by CCL
!  date: 2/15/12
!  This version for time varying conditions.
!  Finally, this version applies TAW methodology for calculating
!  overtopping rates. Metric Units
!
! zc = levee crest
!   UCRIT=1.8, 1.30 and 0.76 for good, fair. & poor grass coverage
!   ework=0.492,0.229 and 0.103 for good,fair & poor grass coverage
! XM=bay-side slope of levee
! ITID= number of input increments for oceanographic conditions
! TTID= time in hours
! TID= temporalwater surface elevation
! HS= temporal wave height
! TS= temporal wave period
!
     common/levee/ucrit,ework,xm
     REAL TTID(itid),TID(itid),HS(itid),TS(itid)
     REAL  TTIDV(45),TIDV(45),HSV(45),TSV(45)
     REAL PLV(20),RV(20),RBARV(20),TR2(20)
     REAL ZC,SUMTOT,ework
     INTEGER ITID,ITIMES,NN,IFAIL

   11 FORMAT(I6,5F8.3)
```

```
   12 FORMAT(/,'I= ',I3,' Time=',F6.2,' Tide=',F5.2,' Hs=',f5.2,
     +' Per=',F5.2/)
   13 FORMAT(I6,F7.3,F10.6,2E12.4)
   38 FORMAT(4F8.3)
! constant parameters
! NN=increment of Rayleigh distribution
! ITIMEs= number of discrete oceanographic conditions

      NN=15
      ITIMEs=40
      PI=4.*ATAN(1.)
      SPI=SQRT(PI)
      TWOPI=2.*PI
      GRAV=9.81
      FRIC=0.08
      SUMTOT=0.0
      QEMPMAX=0.0

!     converted time from hr to second by *100 and then *36
      DO 777 I=1,ITID
777   TTID(I)=100.0*TTID(I)
!
!     WRITE(6,38)(TTID(I),TID(I),HS(I),TS(I),I=1,ITID)
      DT=(TTID(ITID)-TTID(1))/real(ITIMES)
      DTS=36.0*DT
      TTIDV(1)=TTID(1)
!
C ESTABLISH TIDES AND WAVE HEIGHTS AT INTERPOLATED TIMES
!
       TTIDV(1)=TTID(1)
      TIDV(1)=TID(1)
      HSV(1)=HS(1)
      TSV(1)=TS(1)
      DO 60 I=2,ITIMES
       TTIDV(I)=TTIDV(I-1)+DT
      TC=TTIDV(I)
       CALL INTERP_D(ITID,ITIMES,TTID,TID,TC,TIDC)
      TIDV(I)=TIDC
      CALL INTERP_D(ITID,ITIMES,TTID,HS,TC,HSC)
      HSV(I)=HSC
      CALL INTERP_D(ITID,ITIMES,TTID,TS,TC,TSC)
      TSV(I)=TSC
   60 CONTINUE

! begining of analysis

      ZCE=ZC

!   TIME LOOP

       DO 600 I=1,ITIMES
```

```
      TIDC=TIDV(I)
      HSC=HSV(I)
      T=TSV(I)
!      CONVERT TIME STEP INTO HR UNIT
      THR=TTIDV(I)/100
!
!     WRITE(6,12)I,THR,TIDC,HSV(I),TSV(I)

      ZCC=ZCE-TIDC
      XNN=NN

C   CALCULATE 2% RUNUP
       XIRR=XM/SQRT(HSC/(1.56*T**2)) ! Irribarren Number
!      check irribarren number to determine which R2 formula to be used
       IF (XIRR.LT.1.8) THEN
      R2=1.75*HSC*XIRR ! 2% Runup
      ELSE
      R2=HSC*(4.3-1.6/SQRT(XIRR))
      END IF

      RBAR=R2/2.23 ! Average Rubup
      Rrms=RBAR/0.886 ! RMS Runup
!     WRITE(6,*)'Rrms=',Rrms,' Xirr+',XIRR
      PL=EXP(-(ZCC/Rrms)**2) ! Prob Runup > Levee Crest Elev
!      check which overtopping formula to be used

       IF (XIRR.LT.1.8) THEN
      QEMP=0.067*SQRT(GRAV*HSC**3/XM)*XIRR*EXP(-(4.3*ZCC/(HSC*XIRR)))
      ELSE
      QEMP=0.2*SQRT(GRAV*HSC**3)*EXP(-2.3*ZCC/HSC)
      END IF

      IF(QEMP.GT.QEMPMAX)QEMPMAX=QEMP
      TR1=1.0-PL ! This is a overtopping time reduction factor.
      TR1=PL
!
C IF PL< 0.01, ASSUMED THAT NO OVERTOPPING OCCURS
!
       IF(PL.LT.0.01)GO TO 598
       DP=PL/XNN

      PLV(1)=PL
      RV(1)=ZCC
      N=1
!     WRITE(6,*)N,PLV(1),RV(1)
!
C     Calculate runup values associated with equal probability
increements
!
      DO 100 N=2,NN+1
       PLV(N)=PLV(N-1)-DP
```

```
      IF(N.EQ.NN+1)PLV(N)=0.000001
      RV(N)=Rrms*SQRT(ALOG(1.0/PLV(N)))
!     WRITE(6,*)N,PLV(N),RV(N)
  100 CONTINUE
!
C    CALCULATE MEAN VALUES OF PROBABILITY INTERVALS
!
       DO 120 N=1,NN
      A1=EXP(-(RV(N)/RRMS)**2)
      A2=EXP(-(RV(N+1)/RRMS)**2)
       B1=RV(N)/RRMS
      B2=RV(N+1)/RRMS
      CALL ERFN(B1,BB1)
      CALL ERFN(B2,BB2)
      RBARV(N)=RV(N)*A1-RV(N+1)*A2
     1 +SPI/2.0*Rrms*(-BB1+BB2)
      RBARV(N)=1.0/DP*RBARV(N)
  120 CONTINUE
      XL1=GRAV*XM*T**2/4.0 ! This wave length came from Bessel
Functions
      XK=TWOPI/(XL1)

!       WRITE(6,25)(N,PLV(N),RV(N),RBARV(N),N=1,NN)
!
C     NEXT, CALCULATE OVERTOPPING RATES BASED ON INDIVIDUAL AVERAGE
RUNUP VALUES
!
       SUM=0.0
       DO 240 N=1,NN
      RC=RBARV(N)
      IF(RC-ZCC.LE.0.0)GO TO 240
      XC=1.0/XK*ACOS(ZCC/RC)
      Tr2(N)=1.0/PI*XC*XK
C     WRITE(6,37)N,RC,ZCC,TR2(N),TR1,TR1
      SUM=SUM+RC-ZCC ! Modified
  240 CONTINUE
       XNN=NN
       XKRUN=QEMP/(SUM*DP) ! ensures that average overtopping agrees
with TAW
!     WRITE(6,21)NN,QEMP,SUM*DP,XKRUN
       SUM=0.0
      SUM2=0.0
      SUMUP=0.0
      FRIC=0.164/QEMP**0.2 ! Added
       DO 280 N=1,NN
      UPREDC3=8.0*XKRUN*GRAV*XM*(RBARV(N)-ZCC)*DP/(TR1*TR2(N)*FRIC) !
Modified
      SUMUP=SUMUP+UPREDC3
C     SUM2=SUM2+UPREDC3*(FRIC/(8.0*GRAV*XM)) ! This was for checking
!     WRITE(6,11)N,DP,UPREDC3,sumup,UCRIT**3,TR1,TR2(N),RBARV(N),ZCC
  280 CONTINUE
```

```
C       WRITE(10,*)QEMP,SUM2,SUM2 ! This was for checking
      SUM=(SUMUP-UCRIT**3)*DTS
      IF(SUM.GT.0.0)SUMTOT=SUMTOT+SUM
  598  continue
!     WRITE(7,13)I,TTIDV(I),QEMP,SUM,SUMTOT*1.0E-06
  600 CONTINUE

! output whether the levee has failed
      sumtot= 1.0e-06*sumtot
      IF(sumtot.GE.ework)     Ifail=2
!
!      output of wave overtopping converted to liter/sec per meter (x
1000)
      WRITE(9,11)Ifail,ZC,SUMTOT,UCRIT,ework,QEMPMAX*1.0E03
 1000 CONTINUE
      RETURN
      END

      SUBROUTINE ERFN(ARG,R)
C     ****************************************
!    This subroutine compute cdf of a Rayleight distribution
!    It is an approximation of an error function
C     ****************************************
      ARGSAV=ARG
      ARG=ABS(ARG)
      A1=0.254829592
      A2=-0.284496736
      A3=1.421413741
      A4=-1.453152027
      A5=1.061405429
      P=0.3275911
      T=1.0/(1.0+P*ARG)
      R=1.0-(A1*T+A2*T**2+A3*T**3+A4*T**4+A5*T**5)*
     1 EXP(-ARG**2)
      IF(ARGSAV.LT.0.0)R=-R
      RETURN
      END

      SUBROUTINE INTERP_D(ITID,ITIMES,TTID,TID,TC,TIDC)
!     **********************************************
!    This subroutine interpretes the incremental values
!     *********************************************
      DIMENSION TTID(100),TID(100)
      DO 20 I=2,ITID
      IC=I
      ICM=I-1
      IF(TTID(IC).GE.TC.AND.TTID(ICM).LE.TC) GO TO 22
   20 CONTINUE
   22 DEN=(TTID(IC)-TTID(ICM))
      A1=(TC-TTID(ICM))/DEN
      A2=1.0-A1
```

```
      TIDC=A1*TID(IC)+A2*TID(ICM)
C      WRITE(*,*)IC,TC,TIDC,DEN,A1,A2,TID(IC),TID(ICM)
      RETURN
      END
```