NOAA Technical Report NOS CS 24

IMPLEMENTATION OF MODEL SKILL ASSESSMENT SOFTWARE FOR OPERATIONAL HYDRODYNAMIC FORECAST SYSTEMS

UPDATE VERSION

Aijun Zhang
Kurt W. Hess
Eugene Wei
Edward Myers

Silver Spring, Maryland
March 2013

noaa National Oceanic and Atmospheric Administration

U.S. DEPARTMENT OF COMMERCE
National Ocean Service
Office of Coast Survey
Coast Survey Development Laboratory

TABLE OF CONTENTS

LIST OF FIGURES v

LIST OF TABLES v

EXECUTIVE SUMMARY vii

1. INTRODUCTION 1

2. DATA REQUIREMENTS..3
2.1. Definition of Model Run Scenarios 3
2.1.1. Astronomical Tide Simulation Only..3
2.1.2. Hindcast 3
2.1.3. Semi-Operational Nowcast 3
2.1.4. Semi-Operational Forecast...4
2.1.5. Persistence Forecast 4
2.2. Definition of Time Series Variables 5
2.3. Definition of Standard Statistics and Error Criteria...5

3. DATA ANALYSIS TECHNIQUES...7
3.1. Gap Filling and Time Interval Conversion 7
3.2. Filtering...7
3.3. Tidal Prediction and Harmonic Analysis..7
3.4. Concatenation 8
3.5. Extrema Extraction 9

4. OVERVIEW OF THE SOFTWARE SYSTEM 11
4.1. Steps to Execute Skill Assessment 11
4.2. Control Files...14
4.2.1. my_parameters.ctl...14
4.2.2. Control File of Station Information 16
4.3. Installation...16

5. SCRIPT AND FORTRAN PROGRAMS...19
5.1. Shell Scripts 19
5.1.1. STEPS_SETUP.sh 19
5.1.2. STEP2.sh...19
5.1.3. tide_prediction.sh..20
5.1.4. concatenate_hindcast.sh..20
5.1.5. concatenate_nowcast.sh 20
5.1.6. concatenate_forecast.sh 21
5.1.7. harmonic_analysis.sh..21

5.2. Main Fortran Programs...23

5.2.1. skill.f 23
5.2.2. harm29d.f 24
5.2.3. lsqha.f 24
5.2.4. pred.f 25
5.2.5. read_netcdf_modeltides.f 25
5.2.6. read_netcdf_now.f 26
5.2.7. read_netcdf_fcst.f 27
5.2.8. persistence.f 27

5.3. Fortran Subroutines 27
5.3.1. equal_interval 28
5.3.2. foufil 28
5.3.3. prcmp 29
5.3.4. extremes 29
5.3.5. slack 31
5.3.6. spline 32
5.3.7. svd 32

6. FUTURE DEVELOPMENTS 33

ACKNOWLEDGMENTS 33

REFERENCES 35

APPENDIX A. EXAMPLES OF CONTROL FILES 37

APPENDIX B. LIST OF SHELL SCRIPTS 41

APPENDIX C. LIST OF 37 TIDAL CONSTITUENTS 53

APPENDIX D. EXAMPLES OF WATER LEVEL SKILL ASSESSMENT TABLES 55

APPENDIX E. EXAMPLES OF WATER CURRENT SKILL ASSESSMENT TABLES 57

LIST OF TABLES

Table 1. Shell script and Fortran programs included in the skill assessment software package viii
Table 2. Data series Groups and the variables in each group 4
Table 3. Skill Assessment Statistics 6
Table 4. Standard Suite and Standard Criteria for Skill Assessment 6

LIST OF FIGURES

Figure 1. Software directory structure 11
Figure 2. System flowchart of the skill assessment software 13

EXECUTIVE SUMMARY

The National Ocean Service (NOS) is developing and implementing oceanographic nowcast and forecast modeling systems to support navigational and environmental applications in U.S. coastal waters. These prediction systems provide NOAA users with nowcasts (i.e. analysis) and forecast guidance of water levels, currents, water temperature, and salinity for the next 24 to 36 hours. The primary variables are water levels and currents.

NOS requires these modeling systems, whether developed within or outside NOS be assessed for skill in adherence to NOS standards (Hess et al., 2003). Skill assessment is an objective measurement of how well the model nowcast or forecast guidance does when compared to observations. The approach here is to measure the performance of the model in: (1) simulating astronomical tidal variability, (2) simulating total (tide and non-tidal effects) variability, and (3) giving a more accurate forecast than the tide tables and/or persistence. The skill assessment scores are, admittedly, difficult to describe and compute. Therefore, NOS' Coast Survey Development Laboratory has developed a software package that computes the scores automatically using data files containing observed, nowcast, and forecast variables. These data are processed and the skill assessment results are displayed in tables which can be incorporated into model evaluation reports.

This report focuses on the water levels and current assessment software according to the procedures for the evaluation of NOS' nowcast/forecast models for navigation as discussed in the standards document (Hess et al., 2003). The software package computes the skill assessment scores automatically using data files containing observed, nowcast, and forecast variables. The observations, such as verified water levels, currents at NOS Physical Oceanographic Real Time System (PORTS) stations and tidal constituents can be directly acquired via the Internet from database of the NOS' Center for Operational Oceanographic Products and Services (CO-OPS). Different types of data are processed and the skill assessment results are listed in tables valid at the selected validation stations. The package's processing routines include tidal prediction, harmonic analysis, gap filling, filtering (or singular value decomposition), and other methods. The routines also include ways of concatenating nowcast and forecast guidance, and in extracting extrema. All programs (including shell scripts and Fortran) are listed in Table 1. This package can be run in Unix or Linux environments. All Fortran programs can be compiled using Fortran compilers, version 77, or above.

This report is designed to be a stand-alone user's guide for each of the programs, giving a detailed explanation of how the calculations are carried out, options to be set by the user, and sample input and output files.

Key words: oceanographic predictions, nowcast, forecast guidance, skill assessment, water levels, currents, tides

Table 1. Shell Script and Fortran programs included in the skill assessment software package.

Program Name	Function
Shell Script	
STEPS_SETUP.sh	set up all parameters and directories
SKILLSTEPS.sh	execute all steps together
STEP2.sh-STEP10.sh	execute each step separately
get_WL_historic_TIDES.sh	acquire CO-OPS verified 6 minutes or hourly water levels from NWLON and PORTS databases
get_WL_historic_GLAKES.sh	acquire CO-OPS verified 6 minutes or hourly water levels from Great Lakes databases
get_obs_PORTS.sh	acquire observations of water levels (6 minutes or hourly), currents, surface temperature and salinity at PORTS stations
tide_prediction.sh	make water level and current predictions
concatenate_hindcast.sh	concatenate model hindcast station files
concatenate_nowcast.sh	concatenate model nowcast station files
concatenate_forecast.sh	concatenate model forecast station files
harmonic_analysis.sh	conduct harmonic analysis of water levels and currents
harmonic_analysis_obs.sh	conduct harmonic analysis for observed water levels and currents
Main Fortran Programs	
skill.f	read in all required time series, compute statistical variables, and generate skill assessment tables.
harm29.f	Fourier harmonic analysis for 29 day water level and current time series
harm15.f	Fourier harmonic analysis for 15 day water level and current time series
lsqha.f	least squares harmonic analysis
pred.f	make water level and current predictions
read_netcdf_modeltides.f	read a single netCDF file of model simulations
read_netcdf_now.f	read model nowcast files in netCDF format
read_netcdf_fcst.f	read model forecast files in netCDF format
persistence.f	make persistence forecasts
Fortran Subroutines	
equal_interval	convert a time series with interval delt0 to a continuous equally spaced time series with interval of delt
foufil	Fourier low pass filter
prcmp	compute principal current direction
extremes	extract extreme values for a given time series
slack	compute variables associated with slack water

1. INTRODUCTION

In order to meet its operational oceanographic mission responsibilities, the National Ocean Service (NOS) is developing and implementing nowcast and forecast modeling systems to support NOS' Physical Oceanographic Real Time Systems (PORTS) and other navigational and environmental applications in U.S. coastal waters. These modeling systems are designed to enhance the navigational guidance supplied by NOS' real-time observations by providing guidance regarding both the present (nowcast) and future (forecast) ocean conditions at many locations within an estuary, bay, lake, or the coastal ocean. The primary forecast variables are water levels and currents.

NOS must ensure that the modeling systems that produce nowcasts and forecasts in support of safe navigation, whether developed within or outside NOS, will be assessed for skill in adherence to NOS standards (Hess et al., 2003). Skill assessment is an objective measurement of how well the model nowcast or forecast does when compared to observations. The approach here is to measure the performance of the model in: (1) simulating astronomical tidal variability, (2) simulating total (tide and non-tidal effects) variability, and (3) producing a more accurate forecast than the tide tables and/or persistence.

This report discusses the specific procedures for the evaluation of NOS' nowcast/forecast modeling systems for navigation as discussed in the standards document (Hess et al., 2003). The skill assessment scores are, admittedly, difficult to describe and compute. Therefore, we have undertaken to develop a software package that will compute the scores automatically using data files containing observed, nowcast, and forecast variables. These data are processed and the skill assessment results are displayed in tables which may be incorporated into model evaluation reports. The processing routines include harmonic analysis, gap filling, filtering (or singular value decomposition), and other methods. They also include ways of concatenating forecasts and in extracting water level and current extrema.

NOAA's Coast Survey Development Laboratory (CSDL) presently develops and uses several different modeling systems. Each is different in various ways but all have a unique standard netCDF output file format. NetCDF is probably the most popular in the oceanographic community and is also used by the atmospheric modeling community outside of the national meteorological and oceanographic operational forecast centers. In theory netCDF is "self describing", which means that programs written with the netCDF library may read these files, find the names and descriptions of the included variables and retrieve the essential information. A core set of water variables, such as velocity, water level and optional salinity and temperature are specified and saved in the netCDF files. The skill assessment software can take this type of netCDF files as its input. Therefore, this software can easily be applied to all modeling systems using the NOS's standard netCDF output file formats.

Sections two and three of this report focus on data requirements and data analysis techniques. The subsequent sections provide an overview of the software system (Section 4) and a discussion of the individual computer programs that comprise the system (Section 5). Future developments are explained in Section 6. The five appendices provide samples of tabular output for water levels and currents, examples of control files, and list of shell script.

Conventions used in this report is as follows:

- Commands, path names, file names, and program names are in italic Courier font. Bold font is used when they appear for the first time in the text and are sometimes used to emphasize important points.
- Actual script, Fortran codes, and examples of control file in the text are in Courier font.

2. DATA REQUIREMENTS

Three basic types of time series data are required to assess the skill of an oceanographic forecast modeling system at a specific location (i.e. validation station): observed, tidally predicted (for tidal regions), and model simulated. A uniform time interval of 6 minutes is required for each series, but 1-hr intervals are suitable for water levels. The length of each time series is ideally 365 days in order to capture all expected seasonal conditions. However, it is sometimes difficult to get such a long time series. Therefore, the suggested minimum length of time is 6 months for water levels and 29 days for currents. All model output and observational data units are to conform to the international standard for units and time reference (UTC), although English units may occasionally appear for reference.

All observational data have to be quality-controlled and processed to final units (e.g., meters or m/s). It is expected that there will be occasional gaps that can be filled by some simple methods (see Section 3.3). Within NOS, CO-OPS is the standard source for water level and current data. CO-OPS' verified water level data are available from its web site.

Tidally predicted data are based on NOS' 37 standard constituents obtained either from NOS' Center for Operational Oceanographic Products and Services (CO-OPS) or derived from observational time series by harmonic analysis (see Section 3.1). The NOS standard prediction method (see Section 3.2) uses harmonic constants, lunar node factors, and equilibrium arguments.

Model output are generated by running the model under one of four scenarios: (1) astronomical tide only, (2) hindcast, (3) semi-operational nowcast, and (4) semi-operational forecast. The scenarios are described below.

2.1. Definition of Model Run Scenarios

2.1.1. Astronomical Tide Simulation Only

For regions where there are significant tidal variations, the model is run in the astronomical tide only scenario as tidal variations may account for a significant part of the error. In this scenario, the model is forced with only harmonically-predicted astronomical tides for the ocean boundary water levels. There is no surface forcing (wind, pressure, etc.). The temperature and salinity should be set as constant and there are no (or constant) river flow inputs. The model time series can be compared with tidal predictions, and be harmonically analyzed to produce constituent amplitudes and phases for comparison with accepted values. The model time series for this scenario should be demeaned because the mean value of tidal prediction is normally zero.

2.1.2. Hindcast

In this scenario, model forcing is based on historical, best available gap-filled observational data for open boundary water levels, surface winds, temperature, salinity, and river flows. The model time series can be compared with the available observations.

2.1.3. Semi-Operational Nowcast

In this scenario, the model forcing is based on real time observed values. The real-time observation may be incomplete and have gaps. The operational model will be restarted often (for instance, four times daily). The ability of the model to correctly work in the restarting mode will be tested. This run tests the ability of the model in an operational environment.

2.1.4. Semi-Operational Forecast

In this scenario, the model forcing is based on recent forecast guidance from other models (e.g. weather prediction, coastal ocean, river), even though some data could be missing. Initial conditions are generated from observed data or the output from a nowcast. This run tests the ability of the model to produce forecast guidance in an operational environment.

2.1.5. Persistence Forecast

A persistence forecast is constructed by adding an offset value, which is based on an observed offset at one station during some time period before the forecast is made (subtracting the tidal prediction from observation produces the non-tidal component), to the tidal prediction for the duration of the 24 hour forecast. For currents, the offset may be a mean current. This procedure synthesizes the information available to a mariner under normal condition with real-time observations and tide tables.

Table 2. Data series groups and the variables in each. Note that upper case letters indicate a prediction series (e.g., H), and lower case letters (e.g., h) indicate a reference series (observation or astronomical prediction). Slack water is defined as a current speed less than ½ knot. The direction is computed only for current speeds greater than ½ knot (from Hess et al., 2003).

Group	Variable	Symbol
Group 1	Water level	H, h
(Time Series)	Current speed	U, u
	Current direction	D,d
	Salinity	S, s
	Water temperature	T,t
Group 2	Amplitude of high water	AHW,ahw
(Values at a Tidal Stage)	Amplitude of low water	ALW,ahw
	Time of high water	THW,thw
	Time of low water	TLW,tlw
	Amplitude of maximum flood current	AFC,afc
	Amplitude of maximum ebb current	AEC,aec
	Time of maximum flood current	TFC,tfc
	Time of maximum ebb current	TEC,tec
	Direction of current at maximum flood	DFC,dfc
	Direction of current at maximum ebb	DEC,dec
	Time of start of current slack before flood	TSF,tsf
	Time of end of current slack before flood	TEF, tef
	Time of start of current slack before ebb	TSE, tse
	Time of end of current slack before ebb	TEE, tee
Group 3	Water level at forecast projection time of nn hrs	Hnn, hnn
(Values from a Forecast)	Current speed at forecast projection time of nn hrs	Unn, unn
	Current direction at forecast projection time of nn hrs	Dnn, dnn
	Salinity at forecast projection time of nn hrs	Snn, snn
	Water temperature at forecast projection time of nn hrs	Tnn, tnn

2.2. Definition of Time Series Variables by Groups

The following time series are required for skill assessment computations. The definitions are summarized in Table 2.

For Group 1, the data can be either (1) a time series of values (such as observations at a location) or (2) a series of values from concatenated segments (such as a set of 24-hr nowcasts or forecasts starting at one time in the day). For currents, the time series will need to have speed and direction; the direction error is computed only for current speeds greater than ½ knot.

For Group 2, values are created from a Group 1 series by selecting a sub-set of values such as the time and amplitude of high water or the time of the start and end of slack water (defined as having a current speed less than ½ knot).

For Group 3, values of the forecast variable valid at a fixed interval into the forecast (e.g., 0 hr, 6 hr, 12 hr, etc). The comparison series is then the observed variable at the time the forecast is valid. If there are, for example, two forecasts per day, then there will be two 6-hr projection values, separated by 12 hours in time.

2.3. Definition of Standard Statistics and Error Criteria

The following statistical variables are defined and computed in the skill assessment (see Table 3). Most of the statistics have an associated target frequency of occurrence. For example,

$$S(X) \leq P$$

where S is the statistic, X is the acceptable error magnitude (defined by the user), and P is the target frequency (or percentage).

$$CF(X) \geq 90\%, \quad POF(2X) \leq 1, \; NOF(2X) \leq 1$$

Other statistics are expressed as limits on the duration of errors, such as

$$S(X) \leq L$$

where L is the time limit or maximum allowable duration

$$MDPO(2X) \leq L, \qquad MDNO(2X) \leq L$$

The standard criteria for skill assessment are listed in Table 4.

Table 3. Skill Assessment Statistics (from Hess et al., 2003)

Variable	Explanation
Error	The error is defined as the predicted value, p, minus the reference (observed or astronomical tide value, r : $e_i = p_i - r_i$.
SM	Series Mean. The mean value of a series y. Calculated as $\bar{y} = \frac{1}{N}\sum_{i=1}^{N} y_i$.
RMSE	Root Mean Square Error. Calculated as $RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N} e_i^2}$.
SD	Standard Deviation. Calculated as $SD = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N} (e_i - \bar{e})^2}$
CF(X)	Central Frequency. Fraction (percentage) of errors that lie within the limits $\pm$X.
POF(X)	Positive Outlier Frequency. Fraction (percentage) of errors that are greater than X.
NOF(X)	Negative Outlier Frequency. Fraction (percentage) of errors that are less than -X.
MDPO(X)	Maximum Duration of Positive Outliers. A positive outlier event is two or more consecutive occurrences of an error greater than X. MDPO is the length of time (based on the number of consecutive occurrences) of the longest event.
MDNO(X)	Maximum Duration of Negative Outliers. A negative outlier event is two or more consecutive occurrences of an error less than -X. MDNO is the length of time (based on the number of consecutive occurrences) of the longest event.
WOF(X)	Worst Case Outlier Frequency. Fraction (percentage) of errors that, given an error of magnitude exceeding X, either (1) the simulated value of water level is greater than the astronomical tide and the observed value is less than the astronomical tide, or (2) the simulated value of water level is less than the astronomical tide and the observed value is greater than the astronomical tide.
SKILL	Index of Agreement (defined by Willmott and Wicks, 1980, Willmott, 1981) $SKILL = 1 - \frac{\sum_{i=1}^{N}(M_i - O_i)^2}{\sum_{i=1}^{N}(\lvert M_i - \overline{M}\rvert + \lvert O_i - \overline{O}\rvert)^2}$

Table 4. Standard Suite of Statistics and Standard Criteria (from Hess et al., 2003)

Variable	SM	RMSE	SD	NOF(2x)	CF(X)	POF(2X)	MDPO(2X)	MDNO(2X)	WOF(2X)
Criterion	none	none	none	≤1%	≥90%	≤1%	≤ L	≤ L	≤ 0.5%

3. DATA ANALYSIS TECHNIQUES

Observational data and model output are processed and analyzed using several techniques. Observed time series that have gaps in the data are filled in one of three possible ways. Model-generated series, which are usually produced from numerous individual runs, must be concatenated to form a continuous series. For each type, the entire series is analyzed for harmonic constants and extrema (e.g., high water, maximum flood current) values. Specific methods for each process are discussed below.

3.1. Gap Filling and Time Interval Conversion

Data gaps often exist in observations, and the extraction of extrema cannot be accomplished in a time series with gaps. Data gaps can be filled using different interpolation methods. Three methods, (linear interpolation, cubic spline interpolation, and singular value decomposition [SVD]) are adopted in the gap filling program. As an option, the user can choose any method according to his experience and data simulation. If a gap is small enough, simple linear interpolation is appropriate. If a gap is large, a cubic spline or SVD interpolation should be used. The cubic spline interpolation is smooth in the first derivative, and continuous in the second derivative, both within an interval and at its boundaries. SVD produces a solution that is the best approximation in the least-squares sense in the case of an overdetermined system (i.e., where the number of data points is greater than number of parameters), and SVD also produces a solution whose values are smallest in the least-squares sense in the case of an underdetermined system (i.e., where the number of data points is less than number of parameters, or if ambiguous combinations of parameters exist). SVD's disadvantage is that it requires more memory space and can be significantly slower than solving the normal equations. However, its great advantage is that it (theoretically) cannot fail, and this more than makes up for the speed disadvantage.

The time intervals of observation and modeled time series might be different. The package will convert all time series with different time intervals into equally-spaced time series with the same unique desired time interval.

3.2. Filtering

Because of short period variations and noise, filtering of values in a time series is sometimes necessary to select accurately the extrema (i.e., maximum and minimum) values and times. A Fourier filter is used in this software as it computes the amplitudes of the components of the signal at various frequencies and reduces the amplitudes at selected frequencies. Simple smoothing is to be avoided because it reduces extrema amplitudes.

3.3. Tidal Prediction and Harmonic Analysis

Tidal prediction of water level and current is required for skill assessment in tidal regions. Tidal harmonic constants can be obtained either from the CO-OPS or can be derived from observations or model output using a Fourier harmonic analysis program or a least squares harmonic analysis program. Astronomical tidal water level and current time series will be predicted from 37 tidal constituents for any time period. The program pred (Zervas, 1999) is used to do such tidal predictions and was modified to overcome the multiyear problem.

In tidal regions, a comparison of tidal harmonic constants is necessary for the evaluation of water levels and currents. For this comparison, the NOS harmonic constants (37 amplitudes and phases) are analyzed from tide-only model simulation and observed data. Two analytical techniques, least squares harmonic analysis and Fourier harmonic analysis, are used in terms of the length of the data time series. The least squares method (Zervas, 1999) is a method for deriving the tidal constituents from a water level or current time series by creating a matrix of covariance between each individual constituent time series and the observed time series. The matrix is inverted to solve for the amplitudes and phases of the harmonic constituents. The constituent with the highest correlation is then subtracted from the observed time series, and the matrix is recalculated with a residual time series in place of the observed. This method has the capability of solving for the 175 tidal constituents, but will not analyze less than 29 days of data. The Fourier harmonic analysis method (Dennis and Long, 1971) uses Fourier series summations to obtain the tidal constituents of water level or current data. This method has been programmed for data periods of either 15 or 29 days of continuous data time series.

3.4. Concatenation

Model outputs of one scenario run sometimes are saved in multiple (NetCDF) files. For example, in the case of operational nowcasts and forecasts, model outputs are normally stored in different (netCDF) files for model runs on different days and on different cycles in the day. Therefore, it is necessary to concatenate certain of these files to construct several continuous time series for further analysis. In the discussion below, we consider the example of a model that is run four times a day (i.e., with four cycles per day) and, for each run, produces a 6-hr nowcast time series and a 36-hr forecast time series, each with a time interval of 0.1 hr.

To concatenate the nowcasts, the output from each cycle of each day is simply appended to the end of the previous cycles' output. This series will be continuous because each nowcast is initialized with the model output for the end of the previous cycle's nowcast. In the example of four cycles per day, each 6-hr nowcast is appended to the previous nowcast. Thus, the 6-hr to 12-hr nowcast is appended to the 0-hr to 6-hr nowcast, and so on.

The forecasts can be concatenated in two ways. In the first method, the value at a single projection time in each forecast is selected. For example, the forecasted value at hour 3 from the second cycle is appended to the forecasted value at hour 3 of the first cycle, and so on. The time interval is 6 hours and the time associated with each value in any one series is the time that the projection is valid. With this method, a unique series can be constructed for each of the 36 hours of the forecast, and individual values can be compared to observations at the same time. In the second method, the first 6 hours of each cycle is appended to the first 6 hours of the previous cycle. This method produces a time series with the time interval of 0.1 hours, although there may be a discontinuity of values every 6 hours, corresponding to the joining of two distinct segments. This series can be used to find outliers and extrema.

3.5. Extrema Extraction

For skill assessment, the amplitudes and times of high and low waters and the amplitudes and times of maximum flood and ebb currents are required. The time series needs to be filtered if there is noise before extracting extrema. The extrema are extracted by searching for the largest and smallest values within a given time period in a series by the following method. First, the time series values within each 0.5-hour segment are averaged to obtain a new series with a time interval of 0.5 hours. Second, preliminary extrema in the new time series are identified from the maxima and minima. Third, using SVD, a 6-th order polynomial is fit through the original, unaveraged data points within 3 hours of the time of each preliminary extrema point. From this polynomial, a refined extrema is determined. Finally, consecutive maxima and consecutive minima, or a maxima-minima pair that are too close in time and/or amplitude, are eliminated, using the specified criteria of *DELHR* and *DELAMP*. While *DELHR* and *DELAMP* are maximum allowed time and amplitude difference between high and low extrema. This method might not be appropriate for a non-tidal time series since consecutive maxima and consecutive minima are eliminated. Therefore, the final step is not applied for non-tidal time series.

4. OVERVIEW OF THE SOFTWARE SYSTEM

The skill assessment software package is designed to conduct skill assessment of water level, current velocities, water temperature, and salinity from different model systems in both tidally-dominated and non-tidal regions. For generic purpose, all time series (observations, tidal predictions, and model outputs) required by skill assessment are processed and reformatted into same ASCII format (see the sample files in Section 5.1).

The directory structure of this software is shown in Figure 1. The directory called ***skill*** (it can be different name) is the root directory of this software package. The executable programs are located in ***bin***. All programs in ***sorc*** have to be compiled by running ***COMPILE.sh*** script in directory ***sorc***, and all executable files are saved in ***bin***. Therefore, this system can be easily installed on different platforms such as Unix environment (Unix commands and executable Fortran programs compiled using F90) and Linux environment (Linux commands and executable Fortran programs compiled using LF95). The control files are stored in ***control_files***. The observations, harmonic constants, and tidal predictions are stored in ***data***. All Fortran source codes are stored in ***sorc***. All shell scripts are stored in ***scripts***. All intermediate files generated during the software execution are stored in ***work***. The data and work directories will be created if they do not exist while this software is executed.

4.1. Steps to Execute Skill Assessment

For running the software at the user's local directory, first copy the skill assessment software to your platform, which includes all scripts, source codes, and control files using the following Unix/Linux command for copying (for instance for OC-OPS ofsdev Linux server) :

```
cp –rp  /home/net/azhang/SKILL_OFS ./skill
```

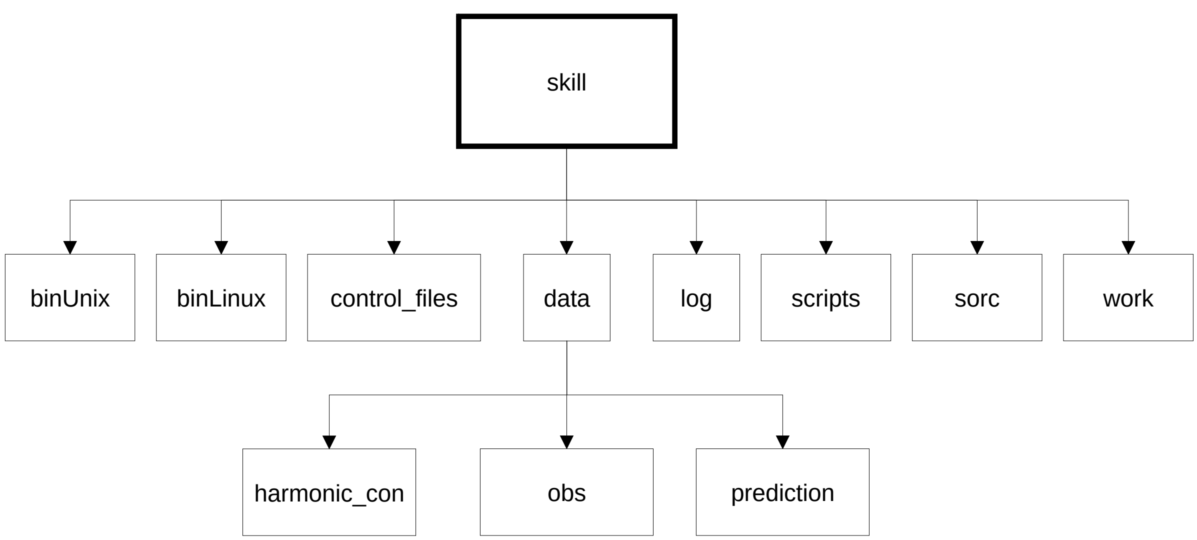

Figure 1. Directory structure of the skill assessment software

Go into directory ***sorc*** to modify ***COMPILE.sh*** to specify correct Fortran compiler (***FC***), C compiler (***CC***), and NetCDF path (***NETCDF_ROOT***) of the platform where software is installed, and then run ***COMPILE.sh*** to compile all needed programs in ***sorc***. Skill assessment software can be run anywhere of user's available hard disks, so it is not necessary to run it inside the directory where skill assessment software is installed (This is updated for this new version). It means that one software package can be used for different projects and by different users. Two directories "***data***" and "***work***" are created under the current directory. Before running skill assessment, set system environment variable HOME1 to be path name where skill assessment is installed as,

For bash shell, use command,

export HOME1= path name (e.g. HOME1=/home/net/azhang/SKILL_OFS)

For C shell, use command,

setenv HOME1 /home/net/azhang/SKILL_OFS/sorc

The following steps are involved in running this package after copying the software to the user's local directory (i.e., /user_home/skill), while steps 2 to 10 are automatically initiated by entering the command "/user_home/skill/***SKILLSTEPS.sh"*** in the directory where ever user is. Or after completing step 1, Steps 2 to 10 can be run separately by entering commands /user_home/skill/***STEP2.sh***, /user_home/skill/***STEP3.sh***, etc.

Step 1: The user needs to provide information such as directory path names, the parameters and station location information described in the two control files, ***`my_parameters.ctl`*** and ***`stationdata.ctl`*** in the subdirectory ***control_files***. Copy these two control files in ***control_files*** into your current directory where user will run skill assessment software. All parameters in these two control files have to be set properly before entering step 2. More detailed explanation for each parameter will be described in Section 4.2.

Step 2: The following types of observational data can be acquired via the Internet by running a shell script program, ***STEP2.sh***.

- Verified water levels with 6 minute and an hour time intervals in NOS' National Water Level Observing Network (NWLON), PORTS, and Great Lakes databases.
- Water velocity in PORTS database.
- Surface water temperature in NWLON, PORTS, and USGS databases.
- Surface salinity in PORTS and USGS databases.

For the stations, at which observational data are not available by running STEP2.sh, the user needs to prepare observational data, and store them in directory ./data/obs. The user can prepare observations (including water levels and currents) from other data sources as well, but the observational data must have the same formats as the sample files in Section 5.1.

Step 3: tidal predictions of water levels and currents are made by running ***STEP3.sh*** using the observed tidal constituents. The accepted water level tidal constituents at stations of NWLON will be acquired from a CO-OPS database via the Internet if they exist, or the user can provide his own tidal constituents by harmonically analyzing observations using ***harmonic_analysis_obs.sh*** (There are no current harmonic constants available from the CO-OPS web site currently). The tidal constituents are in a standard prediction format which can be directly used by the tidal prediction program.

Step 4: ***STEP4.sh*** is used to process model outputs for tidal simulations. The output file name from ***STEP4.sh*** is like "***STATIONNAME_***modeltides.dat" and stored in ***work*** directory. If ***NCYCLE_T*** =0, one single model output file (***MODELTIDES*** specified in ***my_parameters.ctl***) is read in. While ***NCYCLE_T*** >0, multiple model output files (determined from variables of ***ARCHIVE_DIR_T*** and ***NAME_TIDECAST*** specified in ***my_parameters.ctl***) are concatenated using a shell script "*concatenate_hindcast.sh*" to produce continuous monotonous time series for tidal simulation scenario at each station.

Step 5: ***STEP5.sh*** is used to process model outputs for hindcast simulations. The output file name from ***STEP5.sh*** is like "***STATIONNAME_***hindcast.dat" and stored in ***work*** directory. If ***NCYCLE_H*** =0, one single model output file (***HINDCAST*** specified in ***my_parameters.ctl***) is read in. While ***NCYCLE_H*** >0, multiple model output files (determined from variables of ***ARCHIVE_DIR_H*** and ***NAME_HINDCAST*** specified in ***my_parameters.ctl***) are concatenated using a shell script "*concatenate_hindcast.sh*" to produce continuous monotonous time series for hindcast simulation scenario at each station.

Step 6: ***STEP6.sh*** is used to process model outputs for operational or semi-operational nowcast simulations. The output file name from ***STEP6.sh*** is like "***STATIONNAME_***nowcast.dat" and stored in ***work*** directory. Multiple model output files (determined from variables of ***ARCHIVE_DIR*** and ***NAME_NOWCAST*** specified in ***my_parameters.ctl***) are concatenated using a shell script "*concatenate_nowcast.sh*" to produce continuous monotonous time series for nowcast simulation scenario at each station (details are described in section 5.1.5 and 5.2.6).

Step 7: ***STEP7.sh*** is used to process model outputs for operational or semi-operational forecast simulations. The output file name from ***STEP7.sh*** is like "***STATIONNAME_***forecast.dat" and stored in ***work*** directory. Multiple model output files (determined from variables of ***ARCHIVE_DIR*** and ***NAME_FORECAST*** specified in ***my_parameters.ctl***) are concatenated using a shell script "*concatenate_forecast.sh*" to produce time series for forecast simulation scenario at each station (details are described in section 5.1.6 and 5.2.7).

Step 8: persistence forecasts are made from observations and tidal predictions by running ***STEP8.sh*** for forecasting methods comparison.

Step 9: after completing the above processes, all input time series required by the skill assessment program are available with the same ASCII format. Skill assessment is performed by running ***STEP9.sh***. A Fortran program, ***skills.f***, is used to produce skill assessment tables for each station. In `skills.f`, all input time series are processed for low-pass filtering, gap-filling, and extrema extracting. Statistics computation will then be performed to produce all skill assessment score tables.

Step 10: for a tidal region, tidal simulation time series are harmonically analyzed to obtain modeled tidal constituents, which are then compared with the observed tidal constituents. Tables containing tidal harmonic constant comparison are generated by running ***STEP10.sh.***

The system flowchart of the skill assessment software is shown in Figure 2.

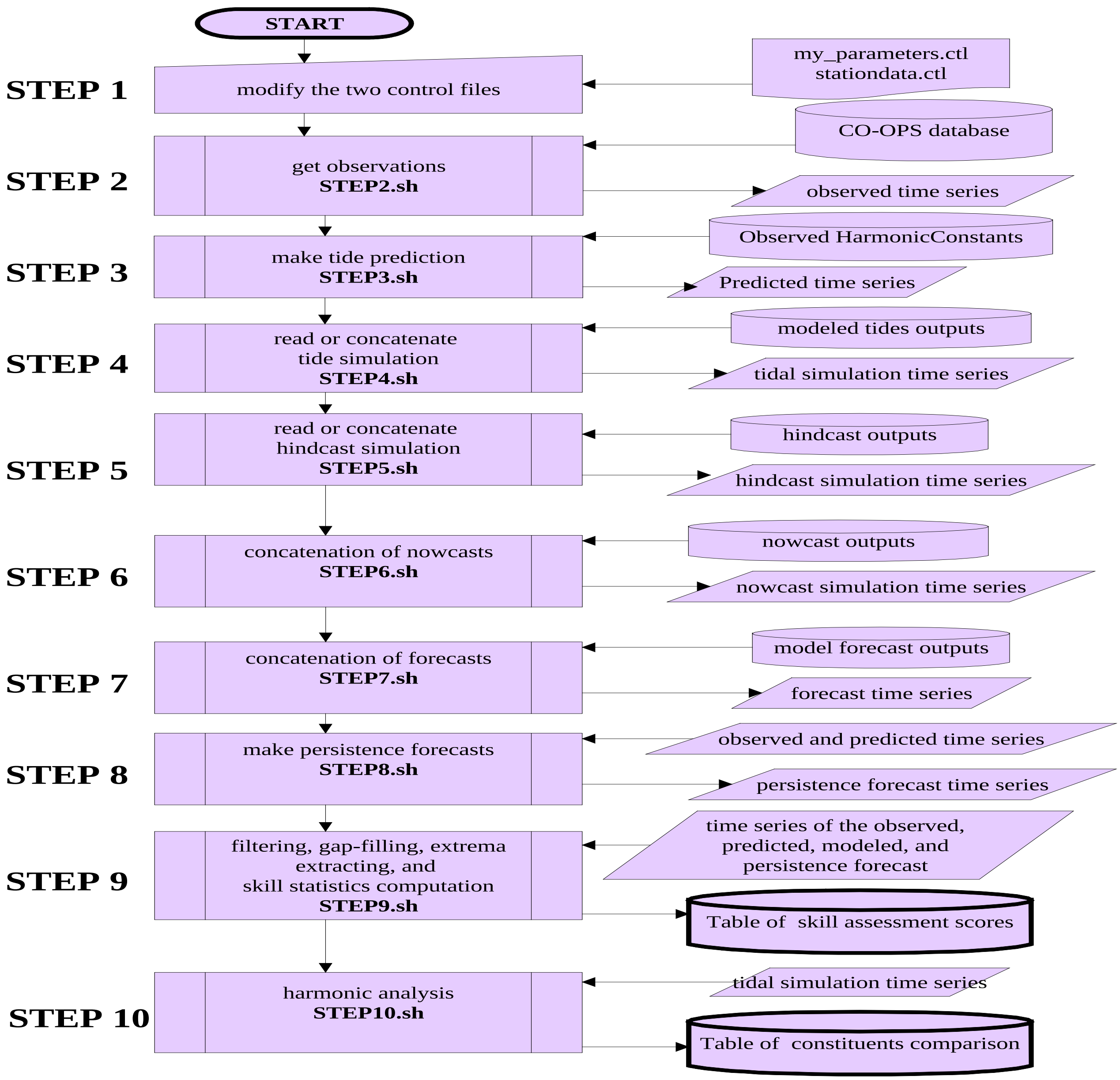

Figure 2. Flowchart of the skill assessment software system.

This software is designed to run in UNIX or LINUX operating system environments, and the programs were written using both shell scripts and the Fortran language. Utility commands, ***datemath*** and ***dateformat***, are used in some shell script programs, and the netCDF Fortran library is required as well for compiling some Fortran programs if steps 4 to 7 are needed.

Several sources of data are needed to run the skill assessment software. External sources are: (1) the CO-OPS web-server observations of water levels, currents, and harmonic constants; and (2) the model simulations (tidal simulations, hindcasts, nowcasts, and forecasts) and persistence forecasts. The interval sources are the two control files *my_parameters.ctl* and *stationdata.ctl*. The two control files are supplied by the user, and are discussed in the following sections.

4.2. Control Files

Two control files are needed for this software, and are described as follows.

4.2.1. my_parameters.ctl

A template file of ***my_parameters.ctl*** is located in the directory ***control_files***. User needs to copy it into user's current directory where skill assessment will be launched. This control file includes parameters that should be provided by the user and are required by the skill assessment software. It is called by the main shell script ***STEPS_SETUP.sh*** as an include file. ***STEPS_SETUP.sh*** is executed in the beginning of each shell script. The parameters include path names, file names, and parameter values. The parameters become script variables which are used by all shell scripts. The user needs to modify this control file before running any scripts. The parameters are explained as follows, and a sample of this control file is listed in Appendix A.1.

HOME1:	root directory name where skill assessment software is located. All sub-directories are named under this directory automatically
STATIONDATA:	station control file name in which station information is included
MODELTIDES:	file name of a netCDF file which includes tidal simulation results This file is processed if ***NCYCLE_T***=0
ARCHIVE_DIR_T:	directory name where model tidal simulation output files are located. It is used to locate tidal simulation file names
NAME_TIDECAST:	a common portion of model output file names for tidal simulation. If ***NCYCLE_T*** > 0, all files determined by ***ARCHIVE_DIR_T*** and ***NAME_TIDECAST*** are concatenated.
HINDCAST:	file name of the netCDF file which includes model hindcast simulations This file is processed if ***NCYCLE_H***=0
ARCHIVE_DIR_H:	directory name where model hindcast simulation output files are located. It is used to locate hindcast simulation file names
NAME_HINDCAST:	a common portion of model output file names for hindcast simulation. If ***NCYCLE_H*** > 0, all file names determined by ***ARCHIVE_DIR_H*** and ***NAME_HINDCAST*** are concatenated.

ARCHIVE_DIR: directory name where model nowcast and forecast output files are located. It is used to determine the nowcast and forecast file names

NAME_NOWCAST: name of model nowcast netCDF file

NAME_FORECAST: name of model forecast netCDF file

BEGINDATE: the beginning date of all data sources used in skill assessment: "yyyy mm dd hh mn", for example, “2009 01 01 00 00”

ENDDATE: the end date of all data sources used in skill assessment: "yyyy mm dd hh mn", for example, “2009 08 01 23 00”.

OS: =0, run in Linux environment, Fortran codes are compiled using LF95
=1, run in Unix environment, Fortran codes are compiled using F90

NTYPE: =0, for non-tidal regions, for example, Great Lakes
=1, for tidal regions, for example, Chesapeake Bay.

FACTOR: used only for event selection in non-tidal regions.
< 0.0: use constant criteria to pick extreme events
upper limit hupper=-factor,and lower limit hlower=factor
>0.0 use dynamic criteria to pick extreme events,
hupper=mean + factor * SD ,hlower=mean -factor * SD.
mean is mean value and SD is standard deviation of the time serial within 7 day period.

DBASE: CO-OPS data base name for grab water level observations
=NWLON, PORTS, or GLAKES

KINDAT: =1 for vector data (current speed and direction);
=2 for water level;
=3 for temperature;
=4 for salinity

TZ_MODEL: transfer time meridian of model outputs from local to GMT/UTC if necessary.
positive for west longitude, and negative for east longitude.
=0 if model output is in GMT/UTCl
=75 if model output is in standard eastern time of US

NFDURATION: forecast duration hours for forecast skill assessment (in hours)

NCYCLE_T: number of tidal simulation cycles per day, =0 read from a single file

NCYCLE_H: number of hindcast cycles per day, =0 read from a single file

NCYCLE_N: number of nowcast cycles per day, >=1

NCYCLE_F: number of forecast cycles per day, >=1

DELT: desired time interval of observation, tide prediction and model outputs, in minutes (e.g., =6 for 6 minute data). All time series used in skill assessment are converted to equal-spaced time series with interval of DELT while running STEP9.sh.

DELT_O: actual time interval (in minutes) of observation data

DELT_T: actual time interval (in minutes) of tidal predictions

DELT_M: actual time interval (in minutes) of model simulation data

CUTOFF: cutoff period (in hours) for Fourier filtering. =30 for 30-hour low-pass filtering, =0 no filtering

IGAPFILL: control switch of gap filling
0: filling missing value with -999.0;

1: filling missing value with interpolation value

METHOD: index of interpolation method if `IGAPFILL=1.`
0: cubic spline;
1: Singular Value Decomposition (SVD)

CRITERIA1: (in hours) Interpolation uses linear or cubic spline method when gap is less than criteria1. The suggested value is 2 hours

CRITERIA2: (in hours) Interpolation uses cubic spline or SVD method when criteria1 < gap < criteria2, and uses missing value -999.0 to fill gaps while gap> criteria2. The suggested value is 6 hours

IS: control model run scenarios. =0, do not run skill assessment for the scenario; =1, run skill assessment for the scenario.
IS(1): tidal simulation only
IS(2): hindcast
IS(3): nowcast
IS(4): forecast
IS(5): forecast method comparison: persistence forecast
IS(6): forecast method comparison: tidal prediction

IPRT: print switch. =0, no screen output; =1 screen output

DELHR: maximum allowed time difference between high and low tidal extrema. For tidal regions, if time difference between high and low tidal extrema is smaller than DELHR, eliminate both high and low tidal extrema

DELAMP: maximum allowed amplitude difference between high and low tidal extrema. For tidal regions, if amplitude difference between high and low tidal extrema is smaller, eliminate both

DELPCT: maximum allowed percentage of amplitude difference between high and low tidal extrema, if less, eliminate both

IOPTA: option for selecting amplitude criterion. If IOPTA=2 or 3, DELAMP is calculated from time series, and DELAMP specified above is overwritten.
=1, DELAMP=DELAMP
=2, DELAMP=DELPCT*(maximum amplitude-minimum amplitude)
=3, DELAMP=DELPCT*(average maximum amplitudes)

X1: accepted error criteria for skill assessment. > 0 use constant criteria for all stations; < 0 (X1=-10 for 10 percent of tidal range) compute X1 by percentage of varying/tidal range of each station as
X1 = -X1 * Trange/100
accepted error criteria for water level (0.15 m), current (0.26 m/s), Temperature (7.7 degrees), salinity (3.5 ppt)

X2: accepted error criteria for time (in hours), =0.5 for NOS standards

X11: accepted error criteria for phase (in degrees), =22.5 for NOS standards

NCON: number of tidal constituents to be analyzed by harmonic analysis program

4.2.2. Control File of Station Information

The name of the control file for station information is provided in the control file of *my_parameters.ctl* as a variable of ***STATIONDATA***. Therefore, a consistent file name has to be provided for station information. There are two lines of record for each station, first line contains "*STATIONID STATIONNAME LONGNAME*"; second line contains "*LATITUDE LONGITUDE FLOODDIR SDEPTH WATER_DEPTH*". The parameters included in this control file are explained as follows, and a sample is listed in Appendix A.2.

STATIONID NOS or USGS Station Identification Number which is used to grab observational data from CO-OPS or USGS web site. It is available from CO-OPS and USGS web sites. For example, STATIONID=8638863 for PORTS station of Chesapeake Bay Bridge Tunnel, STATIONID=01646500 for USGS station of Potomac River near Washington

STATIONNAME short name of the station which is used as a portion of the file names for all type of all time series included observations, tidal predictions, and model simulations. For example, if *STATIONNAME=CBBT for 8638863*
file name of observations is CBBT.obs
file name of tidal prediction is CBBT.prd
file name of tidal simulation is CBBT_modeltides.dat
file name of model hindcast is CBBT_hindcast.dat
file name of model nowcast is CBBT_nowcast.dat
file name of model forecast is CBBT_forecast.dat
file name of persistence forecast is CBBT_persistence.dat

LONGNAME full name of the station, a string less than 80 characters.

LATITUDE latitude of the station (decimal format), south hemisphere is in negative.

LONGITUDE longitude of the station (decimal format). West longitude is negative.

FLOODDIR flood current direction of the station, in degrees clockwise from north. It can be computed by a Fortran subroutine, ***prcmp***, This variable is used only for current data.
For scalar variable such as water level, temperature, and salinity, this parameter is used for offset/datum adjustment to take into account of the differences between observations and model simulations. In this case,

FLOODDIR = mean of observations – mean of model simulations
Adjusted model value = original model simulation + ***FLOODDIR***

FLOODDIR has to be zero for scalar variable if no adjustment is needed

SDEPTH observed (standard or vertical) depth from surface in meters, at which model results are compared with the observations. In step 4-7, the model results at that depth are extracted from the vertical profiles using either linear or cubic spline interpolation method. Its value is ignored for water levels.

WATER_DEPTH total water depth of the observed station in meters (not really used)

4.3. Installation

The skill assessment software is a stand-alone package, and it is designed to be as computer system independent as possible. To install this software, the user just copies the files in the subdirectories of *control_files*, *sorc* and *scripts*, and compiles all programs with a built-in compiling script, *COMPILE.sh*, as necessary.

This software package had been committed to a Concurrent Versioning System (CVS) as well, so MMAP users can install it on a user's local computer using the CVS. For instance, if a user likes to run the skill assessment at his/her local directory:

```
/disks/NASWORK/user/
```

The following commands are used to install the skill assessment software in the user's local directory from the CVS repository:

```
bash
export CVSROOT=dsofs1.nos-tcn.noaa.gov:/comf/CVSPROJECTS
export CVS_RSH=ssh
cd /disks/NASWORK/user/
cvs co SKILL_TEST
```

After execution of the above commands, a directory called *SKILL_TEST* is created, and all of the required programs are saved in different subdirectories under *SKILL_TEST*. ***COMPILE.sh*** in *sorc* can be executed for compiling all fortran programs if the executable files in *binLinux* or *binUnix* do not work in the user's local operating system.

5. SCRIPT AND FORTRAN PROGRAMS

The software package consists of several shell scripts and Fortran programs. It can be run in either a Unix or Linux environment by specifying the value of the parameter *`OS`* in the control file of *`my_parameter.ctl`*. The all steps described in Section 4.1 can be done together by running *`SKILLSTEPS.sh`*. Or any step can be run separately as needed without an impact on the execution of other steps if user does not need to run some of them (provided the user supplies the data that the script or program generated). However, the time series provided by the user should be in the same format as the samples shown in Section 5.1.1. As shown in Figure 1, the user has to modify the two control files, *`my_parameters.ctl`* and *`stationdata.ctl`* to provide correct parameter values associated with a specific project before running any shell scripts.

The main processes performed in *`SKILLSTEPS.sh`* are discussed in the order they occur: (1) parameter setup, (2) acquisition of verified water level observations from CO-OPS database via the Internet; (3) tide prediction and tidal constituents acquisition from CO-OPS database via the Internet if necessary; (4) concatenation of model hindcasts, nowcasts, and forecasts to form continuous time series; (5) creation of a persistence forecast based on the tidal prediction and observation; (6) computation of standard statistics variables to produce skill assessment table; and (7) harmonic analysis and tide constituents comparison. Note that *`SKILLSTEPS.sh`* creates additional temporary control files to be read by fortran programs which generate data for the skill assessment. The shell scripts will be discussed in Section 5.1. Section 5.2 will describe major main Fortran programs, and Section 5.3 will explain the Fortran subroutines. All shell scripts are listed in Appendix B.

5.1. Shell Scripts

5.1.1. STEPS_SETUP.sh

In this script, all parameters are set up by calling control file, *`my_parameters.ctl`*. The required path names are specified and created if they do not exist.

5.1.2. STEP2.sh

This shell script is executed to acquire observational data from a CO-OPS database using the Unix/Linux command ***`wget`*** via the Internet. An ASCII file with Fortran format of "***(f10.5,I5,4I3,4f10.4)***" is generated for each station.

The file name of each station is automatically created as "*`stationname".`****`obs`*** and all observations are stored in ***./data/obs***. The variable *`stationname`* is read in from the control file for station information of *`stationdata.ctl`*. A sample file of water level time series is the following:

```
Julianday YYYY MM DD HH MIN   wl(in meters), this line is not included in
159.00000 1998  6  8  0  0    0.6420
159.00417 1998  6  8  0  6    0.6355
159.00833 1998  6  8  0 12    0.6291
```

```
 159.01250 1998  6  8  0 18    0.6225
```

A sample file of the current time series is the following:

```
 Julianday YYYY MM DD HH MIN  spd(m/s) dir(deg.)  u(east)  v(north)
 159.00000 1998  6  8  0  0   0.68200 246.50000  -0.62500  -0.27200
 159.00417 1998  6  8  0  6   0.66500 251.20000  -0.63000  -0.21400
 159.00833 1998  6  8  0 12   0.57700 251.10001  -0.54600  -0.18700
 159.01250 1998  6  8  0 18   0.41300 244.00000  -0.37100  -0.18100
 159.01666 1998  6  8  0 24   0.47800 263.50000  -0.47500  -0.05400
 159.02083 1998  6  8  0 30   0.51500 270.10001  -0.51500   0.00100
 159.02499 1998  6  8  0 36   0.41200 263.00000  -0.40900  -0.05000
 159.02916 1998  6  8  0 42   0.49000 247.39999  -0.45200  -0.18800
 159.03334 1998  6  8  0 48   0.50500 235.80000  -0.41800  -0.28400
 159.03751 1998  6  8  0 54   0.52200 243.60001  -0.46800  -0.23200
 159.04167 1998  6  8  1  0   0.45300 239.00000  -0.38800  -0.23300
```

The scripts, ***get_WL_verified.sh*** and ***get_obs_PORTS.sh*** are called to acquire different type of observations from different databases in this script. This program can be skipped if the user's observations are not from the CO-OPS databases. However, observational data of water level and current from other sources have to be converted to the same formats described above.

5.1.3. tide_prediction.sh

This shell script is executed in ***STEPS3.sh*** to make tidal elevation and tidal current predictions for any specified time period using the observed harmonic constants (either CO-OPS accepted harmonic constants, or the harmonic constants derived from observations using harmonic analysis programs) for 37 tidal constituents (see Appendix C) that are consistent with those used by CO-OPS to make tidal prediction tables. The CO-OPS accepted elevation harmonic constants of the 37 tidal constituents will be automatically obtained in this script if the harmonic constants file does not exist in the directory of *CONSTANTS_DIR*. The phase epoch obtained in this program is relative to Greenwich Mean Time (GMT). Therefore the predicted time is in GMT.

The file name for each station is automatically created as *"stationname"**.prd***, and all predictions are stored in *data/prediction*. The output file format is the same as the observation data file.

5.1.4. concatenate_hindcast.sh

This shell script is executed in ***STEP4.sh*** and ***STEP5.sh*** to concatenate model tidal simulation and hindcast netCDF station files. In general for a long period simulations, model outputs might be archived into multiple netCDF files (assuming that one netCDF file is generated for each cycle hindcast), then this script program finds all netCDF file names within any specified time period (from *BEGINDATE* to *ENDDATE*). The shell script can automatically locate all netCDF files by the provided parameters of *ARCHIVE_DIR_H* and *NAME_HINDCAST* given in my*_parameters.ctl* as,

ARCHIVE_DIR_H="/nos_ofs/archive/netcdf"

NAME_HINDCAST=_"$ARCHIVE_DIR_H/*${NAME_HINDCAST}*.nc"

A Fortran program, ***read_netcdf_now.f***, loops through and reads each of these netCDF files, and picks the data within the corresponding time period. An ASCII file that includes model results (water level, current, temperature, and salinity time series) from the time period of *BEGINDATE* to *ENDDATE* is generated for each station. There might be gaps if model running failed for some cycles. The output data format is the same as that for the observations.

5.1.5. concatenate_nowcast.sh

This shell script is executed in ***STEP6.sh*** to concatenate model nowcast netCDF station files. In general, an operational model system archives model outputs into a single netCDF file for each nowcast cycle (assuming that one netCDF file is generated for each cycle nowcast), then this script program finds all netCDF file names within any specified time period (from *BEGINDATE* to *ENDDATE*). For example, NOS' Chesapeake Bay Operational Forecast System (CBOFS) outputs the 1200 GMT nowcast of 05/10/2004 using the following file name,

"/ngofs/oqcs/cbofs/archive/netcdf/200405/200405101200_CBOFS_stationsnow.nc"

The model nowcasts from 06Z to 12Z at the selected stations are stored in this file. The shell script can automatically locate nowcast netCDF files of each nowcast cycle by the provided parameters of *ARCHIVE_DIR* and *NAME_NOWCAST* given in my*_parameters.ctl* as,

ARCHIVE_DIR="/ngofs/oqcs/cbofs/archive/netcdf"

NAME_NOWCAST=_"$ARCHIVE_DIR/%Y%m/%Y%m%d%H00_CBOFS_stationsnow.nc"

For current velocity, temperature, and salinity, skill assessment is conducted only at NOS prediction depths that are 15 feet below mean lower low water (MLLW) or one-half the MLLW depth, whichever is smaller, or at the measured depth. Therefore, the user has to specify the vertical depth (i.e., ***SDEPTH*** in station control file) for the model simulation at each station at which the model results are compared with the predictions and observations. This script can extract model results at the specified depth by vertically linear or cubic spline interpolation. Sigma vertical coordinates are converted to z-coordinate using the formulation used by Princeton Ocean Model (POM) by default. The corresponding station of the specified observed station is selected based on the minimum distance between the observed station and model output locations.
A Fortran program, ***read_netcdf_now.f***, loops through and reads each of these netCDF files, and picks the data within the corresponding time period (24/*NCYCLE* hours for hindcasts and nowcasts). An ASCII file that includes model results (water level, current, temperature, and salinity time series) from the time period of *BEGINDATE* to *ENDDATE* is generated for each station. There might be gaps if model nowcast running failed for some cycles. The output data format is the same as that for the observations.

5.1.6. concatenate_forecast.sh

This shell script is executed in ***STEP7.sh*** to concatenate model forecast netCDF station files in a way very similar to the model nowcast concatenation. First the script selects all netCDF file names by the provided parameters of *ARCHIVE_DIR* and *NAME_FORECAST* given in *my_parameters.ctl* in the specific time period from *BEGINDATE* to *ENDDATE* while all cycle forecasts are available within a day. A Fortran program, ***read_netcdf_fcst.f***, loops through and reads each of these netCDF files, and picks the data within the corresponding time period of *NFDURATION* hours long. An ASCII file that includes *NFDURATION* hours forecast time series of all cycles of each day from the time period of *BEGINDATE* to *ENDDATE* is generated for each station. The output data format is the same as that for the observations. The user has to provide his/her own naming convention such as the output archiving directory and the file names, the station index and the vertical depth in the two control files.

5.1.7. harmonic_analysis.sh

This shell script is executed in ***STEP10.sh*** to conduct harmonic analysis of water level and current time series that contain observations or model simulation outputs. Three methods, least squares, 29-day, and 15-day harmonic analysis techniques, are provided, one of them is chosen in terms of the time length of the analyzed time series. A least-squares harmonic analysis technique is chosen for a time series at least 40 day long, and 29 day Fourier harmonic method is used if a time series is shorter than 40 days but longer than 29 days, 15 day Fourier harmonic method is used if a time series is shorter than 29 days but longer than 15 days. The longest continuous segment will be picked for harmonic analysis if there are gaps in the time series. The principal current direction is calculated in the Fortran program for current harmonic analysis. The harmonic constants of 37 tidal constituents are saved in an ASCII file that can be directly used by the tidal prediction program and for tidal harmonic constants comparison.

The output harmonic constant file name of each station is automatically created as *"shortname".std*. The Fortran statements for reading the harmonic constant outputs (in a standard prediction format) are,

```
       READ (LIN,550)  HEAD(1),HEAD(2)
      READ (LIN,532)DATUM,ISTA(1),NO(1),(AMP(J),EPOC(J),J=1,7),
     1 ISTA(2),NO(2),
     2 (AMP(J),EPOC(J),J=8,14),ISTA(3),NO(3),(AMP(J),EPOC(J),J=15,21),
     3 ISTA(4),NO(4),(AMP(J),EPOC(J),J=22,28),ISTA(5),NO(5),(AMP(J),
     4 EPOC(J),J=29,35),ISTA(6),NO(6),(AMP(J),EPOC(J),J=36,37)
  550 FORMAT (A80)
  532 FORMAT (F6.3,6(/2I4,7(F5.3,F4.1)))
```

A sample of output file, ***cons.out***, for water level harmonic analysis is,

```
Harmonic Analysis of mayp_modeltides.dat                      R= 0.000
Least Squares H.A.  Beginning  1- 1-1998  at Hour  0.00
    -9
       1  642 334  104 579  142 163   772078   321887   552165   112356
       2   11 362    0   0   131792   32 118    0   0   15 457   16 133
       3    31998   12 367    91673    42139    52048   282299   81 552
       4  1201904   432043    0   0    22167   112056    6 598    1 722
       5    22262   261997    0   0    1 989   41 329    4 156   31 569
```

```
       6    2 948   141995
```

A sample of output file, ***cons.out***, for current harmonic analysis is,

```
Harmonic Analysis ofj2b07_modeltides.dat                    R= 0.002
Least Squares H.A.  Beginning  1- 1-1998  at Hour  0.00     along  95 degrees
   182
       1  7191950   872137  1381751   64 135   75 803   48 210   28 632
       2   202477    22227   19 699   351707    0   0   10 158   131515
       3    0   0   232233    73311    21924    62426    5  89   433314
       4   433231    41969    0   0    41212    32133   171845   14 990
       5    32559   213513    4 866    43525   602197   182478   292064
       6   133193    91047
Harmonic Analysis ofj2b07_modeltides.dat                    R= 0.002
Least Squares H.A.  Beginning  1- 1-1998  at Hour  0.00     along 185 degrees
    12
       1    3 964    22373    31748    11837   11 432    32107    22874
       2    11769    0 407    5 278    1 101    01692    21794    21769
       3    0  35    1 671    1 360    03100    1 600    5 130    31433
       4    31813    6 279    2  85    12288    03507    11003    0 677
       5    0 137    1 592    0 746    01821    5 464    02363    12308
       6    23077    3 639
```

5.2. Main Fortran Programs

In this software package, most data processing and statistical computation are implemented by Fortran programs. All Fortran source codes are in directory ***sorc***, and a script ***COMPILE.sh*** is used to compile all needed Fortran and C programs. In COMPILE.sh, parameters of ***FC*** (Fortran Compiler) and ***NETCDF_ROOT*** (path of NetCDF library) have to be provided correctly. In this section several main Fortran programs are discussed in detail.

5.2.1. skill.f

This is the core program of the skill assessment software package. After a series of data preparation and processing steps, the time series of observation, tidal prediction, and model simulation of several model scenarios are available for skill assessment computations. Each of these time series is in the same ASCII format and is located in a specific directory. From these modeled and observed time series, *skill.f* will compute the standard statistics variables listed in Table 3 using the associated error criteria in Table 4. A skill assessment score table for each station will be generated in the format shown in Appendix D and E. Low-pass filtering and gap-filling might be performed depending on the parameters the user provided in ***my_parameters.ctl***. For current assessment, current directions are computed only for speeds not less than 0.26 m/s (0.5 knot/s).

This program is run with the command:

```
skill.x < skill.ctl
```

where *skill.ctl* is the control file automatically created in the script, *STEP9.sh*, from the parameters provided in the control file *my_parameters.ctl*. A sample of *skill.ctl* is shown as,

```
2003 01 02 00 00     :BEGINDATE (YYYY MM DD HH MN)
2003 12 30 00 00     :ENDDATE (YYYY MM DD HH MN)
1 0.1 2 0.0          :NTYPE DELT, DELT_O DELT_M NCYCLE_F, CUTOFF
1 1 1 1 1 1          :IS, control scenario on/off switch:
                       0: assessment for the scenario will not
                       be performed; 1: assessment for the
                       scenario will be performed.
1 2 6 1              :IGAPFILL CRITERIA1 CRITERIA2 METHOD
1                    :IPRT, print switches, =0 no screen output
2.0 0.030 0.03 3     :DELHR DELAMP DELPCT IOPTA
0.15 0.5 22.5        :X1, X2, X11
2                    :KINDAT
ECDAstation.input    :the file name of station information
```

A detailed explanation for the above parameters can be found in Section 4.2. The file name for the output score tables of skill assessment for each station is ***"stationname"_table.out*** and ***"stationname"phase_table.out*** (for current only). The examples of skill assessment score tables from the St. Johns River forecast system are listed in Appendix D and E.

5.2.2. harm29d.f

This Fortran program uses Fourier series summations (Dennis and Long, 1971) to obtain the tidal constituents of 29-day continuous, evenly spaced water level or current data. None of the long-term constituents (Mf, MSF, Mm, Sa, and Ssa) are computed. And none of the compound tidal constituents (MK_3, $2MK_3$, etc.), which can be important in shallow water level areas, are solved for. This program solves for ten tidal constituents (M_2, S_2, N_2, O_1, K_1, M_4, M_6, M_8, S_4, and S_6). Once preliminary values for the amplitude and phase epoch of these ten constituents are obtained, fourteen other constituents are inferred using astronomically-determined amplitude ratios and phase shifts. The inferred constituents are: $2Q_1$, Q_1, ρ_1, M_1, P_1, J_1, OO_1, $2N_2$, γ_2, λ_2, L_2, T_2, R_2, and K_2). Following these computations, the elimination of perturbations between closely-spaced constituents is then carried out. The input timer series file is an ASCII file with the same format as that of the observations in Section 5.1. If there are gaps in the input time series, this program will pick the longest continuous equally spaced segment from the time series for analysis. The principal current direction is automatically calculated inside this program from the input time series for current data analysis.

This program is run in a Unix or Linux environment using the command:

```
harm29d.x KINDAT NCON DELT LONGITUDE FILEIN
```

A file named as ***cons.out*** is created with the 37 constituents in the standard predictions format. And this output file can be directly used by the tidal prediction program ***pred.x***. The constituent epochs will be Greenwich epochs. The input argument parameter is defined as,

KINDAT:	=1 for vector data (current speed and direction); =2 for scalar data (water level)
NCON:	the number of tidal constituents to be analyzed. NCON is not relevant for this program, but must be given since same control file is used for both harm29d.x and lsqha.x.
DELT:	time interval of the input time series (in hours).
LONGITUDE	longitude of the location.
FILEIN:	the name of the input data file.

5.2.3. lsqha.f

This Fortran program uses a least squares method of harmonic analysis (see Zervas, 1999) to derive the tidal constituents from water level or current time series. This is done by creating a matrix of covariance (or correlation coefficients) between each individual constituent time series and the observed time series (Harris, et al., 1965). The matrix is inverted to solve for the amplitudes and phases of the harmonic constituents. The constituent with the highest correlation is then subtracted from the observed time series and the matrix is recalculated with the residual time series in place of the observed (an option, *ITYPE*, exists for solving for the constituents in a specified order). This program has the capability of solving for the 175 tidal constituents, But will not analyze the time series if it is less than 29 days long. The formats of input data file are the same as that for harm29d.x. If there are gaps in the input time series, this program will pick the longest continuous equally spaced segment from the time series for analysis. The principal current direction is automatically calculated inside this program from the input time series for the current data analysis.

The program is run in a Unix or Linux environment using the command:

lsqha.x *KINDAT NCON DELT LONGITUDE FILEIN*

A file named *cons.out* is created with the 37 constituents in the same format as the output from *harm29d.x*.

5.2.4. pred.f

This Fortran program is used to predict tidal water levels or currents for any specified time period using the 37 tidal constituents listed in Appendix C. The original codes were from Zervas (1999), but some modifications were made for overcoming multi year problems. This program can be used for multiple year predictions with a maximum array dimension of 200,000.

The program is run in a Unix or Linux environment using the command:

pred.x *“BEGINDATE” “ENDDATE” KINDAT DELT XMAJOR FILEIN FILEOUT*

The following is a description of the command input arguments

BEGINDATE	start time of prediction as "YYYY MM DD HH MN"
ENDDATE	end time of prediction as "YYYY MM DD HH MN"
KINDAT	=1 for vector data (current speed and direction); =2 for scalar data (water level)
DELT:	time interval of the input time series (in hours).
XMAJOR	the axis of the first set of tidal constituents. The second set of tidal constituents should be along XMAJOR+90°. For scalar predictions, the parameter XMAJOR is not relevant, but must be given.
FILEIN	input file which contains tide constituents
FILEOUT	output file containing predicted time series

5.2.5. read_netcdf_modeltides.f

This Fortran program is used to read model simulation (tidal simulation or hindcasts) from a station netCDF format output file generated by using NOS standard netCDF model output program ***write_netcdf_Hydro_station***. A continuous time series will be produced, and saved as an ASCII file for each station by specifying the station index and vertical depth.

This program is run as

read_netcdf_modeltides.x *"BEGINDATE" FILEIN STATIONDATA KINDAT*

where the command input parameters are defined as:

BEGINDATE	start time as "YYYY MM DD HH MN"
FILEIN:	the name of the input data file.
STATIONDATA	control file name in which station information is included
KINDAT	1 for vector data (current speed and direction); 2 for scalar data (water level)

5.2.6. read_netcdf_now.f

This Fortran program is used to read model nowcasts from the station netCDF format output files generated by NOS standard netCDF model output program. This program reads each of the netCDF files specified in the control file and produce a monotonous time series for each station. Corresponding model grid index is automatically chosen according to the minimum distance between the location (latitude and longitude provided in station control file) and model output locations (latitude and longitude of model output). Linear or cubic spline vertical interpolation is conducted for currents, temperature, and salinity to interpolate model value onto the specified measured depth (***SDEPTH*** in station control file). Data from following files are discarded if there is overlap in time.

This program is run using the command:

read_netcdf_now.x < *read_netcdf.ctl*

where the control file is in the following format:

```
2009 01 02 00 00
DELT  NCYCLE_N KINDAT N
STATIONDATA.CTL
/ngofs/oqcs/cbofs/archive/netcdf/200405/200405100000_CBOFS_stationsnow.nc
/ngofs/oqcs/cbofs/archive/netcdf/200405/200405100600_CBOFS_stationsnow.nc
/ngofs/oqcs/cbofs/archive/netcdf/200405/200405101200_CBOFS_stationsnow.nc
/ngofs/oqcs/cbofs/archive/netcdf/200405/200405101800_CBOFS_stationsnow.nc
```

This control file is automatically generated by the script, `concatenate_nowcast.sh`, with the values of *DELT, NCYCLE_N*, and *KINDAT* from *my_parameters.ctl*. *N* is total number of netCDF files to be read. The following N lines contain netCDF file names.

5.2.7. read_netcdf_fcst.f

This Fortran program is used to read model forecasts from the station netCDF format output files. It is run in a way similar to *read_netcdf_now.x*. The difference is that this program picks 24 hours forecasts from each cycle's forecast file, and the time following the file name in the control file is the start time of that cycle's forecasts. The start time might be different for different model system since the model output netCDF file naming convention might be different; the user needs to verify the time periods of his model system's outputs to make sure that the model outputs are correctly picked up. Otherwise, the user needs to modify *concatenate_forecast.sh*.

5.2.8. persistence.f

This Fortran program create persisted forecast time series for one station from its tidal prediction and observation data. For each forecast cycle, an offset between the observation and the tidal prediction at forecast time=0 is calculated. This offset value is then superimposed to the next 24 hour tidal predictions (the offset stays constant for 24 hours) to generate a 24 hour persisted forecast of the cycle. In this software, persisted forecast is defined as the tidal prediction plus an offset, where the offset is equal to observation minus tide prediction at forecast time=0. The user can employ alternative techniques to generate persistence forecasts with the same data format as the model forecasts.

The program is run with the command:

persistence.x < *persistence.ctl*

where the control file, *persistence.ctl*, is automatically generated in the script *STEP8.sh* with the following format,

"stationname".obs:	water level observation file name
"stationname".prd:	tide prediction file name
"stationname"_persistence.dat:	output file name
KINDAT	1 for current; 2 for WL
BEGINDATE	begin date as "2003 01 02 00 00"
ENDDATE	end date as "2003 12 31 00 00"
DELT	desired time interval
NCYCLE_F	number of forecast cycle per day

5.3. Fortran Subroutines

There are some Fortran subroutines included in this software package to carry out different tasks. These subroutines are called by main Fortran programs. The functionality and usage of the subroutines are explained as follows.

5.3.1. equal_interval

This Fortran subroutine is used to convert a time series with time interval *DELT0* to a continuous equally spaced time series with time interval *DELT* for the period from the beginning time to the end time. The data gaps in the original time series are filled using an interpolation method specified by the values of *IGAPFILL, METHOD, CRITERIA1* and *CRITERIA2*. If the data gaps are less than *CRITERIA1*, they are filled with linearly and cubic spline interpolated values. If the data gaps are greater than *CRITERIA1* and less than *CRITERIA2*, they are filled with cubic spline (if *METHOD*=0) or SVD (if *METHOD*=1) interpolated values. If the data gaps are greater than *CRITERIA2*, they are then filled with -999.0. This subroutine is called with the statement,

```
call  equal_interval (DAY_BEGIN, DAY_END, DELT, DELT0, METHOD,
    CRITERIA1,CRITERIA2, TIME, WL,TIME_NEW, WL_NEW, NUM, M_NEW)
```

where the arguments are described as,

input

DAY_BEGIN:	start time in days
DAY_END:	end time in days
DELT:	time interval of output time series in hours
DELT0:	time interval of original time series in hours
METHOD:	index of interpolation method =0, use cubic spline interpolation method =1, use Singular Value Decomposition (SVD)
CRITERIA1:	in hours, if gap < criteria1, fill gaps with linear interpolated values.
CRITERIA2:	in hours, criteria1 < gap < criteria2, fill gaps with cubic spline or SVD interpolated values. gap > criteria2, fill gaps with -999.0.

TIME: an array of original input time.
WL: an array of original input data
NUM: number of original data

output
TIME_NEW: an array of gap-filled output time
WL_NEW: an array of gap-filled output data
NUM_NEW: number of gap-filled data

5.3.2. foufil

This Fortran subroutine is used as a low-pass Fourier filter for a time series using Fast Fourier Transforms (FFT). The call statement is,

```
Call foufil (LENGTH, DELMIN, TCUT, U, AU)
```

where the arguments are described as,

inputs:
LENGTH total data points.
DELMIN data time interval in minutes.
TCUT low-pass filter cutoff period in hours.
U input data-unfiltered series.

outputs
AU output data- filtered series.

5.3.3. prcmp

Before a harmonic analysis of current data can be carried out, the principal current direction must be determined. Tidal constituents can then determined for components parallel and perpendicular to the principal current direction. A Fortran subroutine, ***prcmp***, was designed to calculate principal current direction (see Zervas, 1999). The principal current direction is in degrees clockwise from north. This may be either the flood or the ebb direction. The principal current direction may be strongly affected by the nontidal currents. This subroutine is called with the statement,

```
call prcmp(N,U,V,PCD,RUV,RATIO)
```

where the arguments are
inputs:
N the number of total data points.
U an array containing east (u) velocity component.
V an array containing north (v) velocity component.

outputs:
PCD returned value of the principal current direction.
RUV correlation coefficient between U and V.
RATIO the ratio of minor axis variance to major axis variance.

5.3.4. extremes

This Fortran subroutine extracts the extreme values of a time series. In Table 2, the time series of Group 2 are derived from extreme values of Group 1 by selecting amplitudes and times of high and low water, as well as amplitudes and times of maximum flood and ebb current. These extreme values are extracted using the following procedure: First, the time series are averaged to obtain a new time series with half hour time interval. Second, all peaks in the new time series are found. Third, an SVD procedure is used to fit a curve through the data points (within a 3 hour time window) in the original series around the each peak. Fourth, refined extremum and corresponding time are found from the fitting curve if an extremum exists. Fifth, the extrema which are too close in time and magnitude, and one of the two consecutive highs or lows are eliminated. The final step is not applied for a non-tidal time series. However, upper and lower criteria are applied to pick up higher and lower events. By default, 2-sigma rule (twice of standard deviation of the time series) is applied for specifying event criteria (hupper and hlower). All extreme values and the corresponding time are saved in the arrays of hhighs and thighs. It may be better to filter the observed current data before extracting extreme values since current measurements are normally too noisy. This subroutine is called with the statement,

```
call  extremes(t,h,N,IPRT,DELHR,DELAMP,DELPCT,IOPTA,
      thighs,hhighs,idx,nsmax,CUTOFF,DELT,zhall,thall,num_h,
      zlall,tlall,num_L,hupper,hlower,NTYPE)
```

the following is a description of the parameters in the control file,

inputs:

```
 t      the array of input time.
 h      the array of water level, current speed, or direction.
 N      number of input data.
IPRT    print switch. =0, no screen output; =1 screen output
DELHR   maximum allowed time difference between high and low,
        if smaller than DELHR, eliminate both high and low.
DELAMP  maximum allowed amplitude difference between high and low,
        if smaller, eliminate both
DELPCT  maximum allowed percentage of amplitude difference between
        high and low

IOPTA   option for selecting amplitude criterion. =1, delamp=delamp
       =2, delamp=delpct*(hmax-hmin), =3, delamp=delpct*(average hmax)
CUTOFF  cutoff period (in hours) for Fourier filtering;
        =0 no filtering
DELT    time interval of input time series (in hours, e.g., =0.1
        for 6 minutes data).
hupper  high limit for selecting high events. It is ignored for
        tidal regions.
hlower  lower limit for selecting low events.
NTYPE   =0, for non-tidal regions; =1, for tidal regions.
```

outputs:

thighs	a returned array containing the times of extrema.
hhighs	a returned array containing the extreme values.
idx	a returned array containing index of the extreme values. idx=1 for maximum, and –1 for minimum.
Nsmax	the total number of extreme values.
zhall	the returned array containing all maximum values.
thall	the returned array containing the times of all maximum values.
num_h	total number of maximum values.
zlall	the returned array containing all minimum values.
tlall	the returned array containing the times of all minimum values.
num_l	total number of minimum values.

5.3.5. slack

This Fortran subroutine is used to calculate variables associated with the slack water. Slack water is defined as having a current speed less than 0.26 m/s. Maximum flood and ebb speed values can be extracted by calling subroutine *extremes***.** This Fortran subroutine is then used to calculate the time series for slack water from the user provided flood directions. Time of start and end of current slack (before flood and ebb) is defined by the time when current speed is equal to 0.26 m/s. This program is called with the statement,

```
call slack (t, speed, dirr, thighs, hhighs, idx, NSMAX, nmax,
     DIRFLOOD, DELT, AFC, AEC, TFC, TEC, DFC, DEC, TSF, TEF,
     TSE, TEE, NAFC, NAEC, NTSF, NTSE, nmx, nmx2)
```

The following is a description of the arguments,

inputs:

t	time array.
speed	the array containing current speeds.
dirr	the array containing current directions.
nmax	total input data points.
thighs	array containing time of maximum speed, derived using subroutine *extremes*.
hhighs	array containing maximum speed, derived using subroutine *extremes*.
idx	array containing index of the extreme values derived using subroutine *extremes*. idx=1 for maximum, and –1 for minimum.
NSMAX	total extreme values.
nmax	total number of input time series.
DIRFLOOD	direction of flood currents at a station (principal current direction, or opposite direction), provided by the user.
DELT	time interval of input time series.

outputs:

```
 AFC     amplitude of maximum flood current
 AEC     amplitude of maximum ebb current
 TFC     time of maximum flood current
 TEC     time of maximum ebb current
 DFC     direction of current at maximum flood
 DEC     direction of current at maximum ebb
 TSF     time of start of current slack before flood
 TEF     time of end of current slack before flood
 TSE     time of start of current slack before ebb
 TEE     time of end of current slack before ebb
 NAFC    total number of maximum flood currents.
 NAEC    total number of maximum ebb currents.
 NTSF    total number of time of start and end of current slack
         before flood.
 NTSE    total number of time of start and end of current slack
         before ebb.
```

5.3.6. spline

Subroutine *spline* implements cubic spline interpolation. This fortran subroutine is based on that in Numerical Recipes in Fortran (Press et al., 1992) and made some changes to apply it in data gap filling and interpolation in this software package. It is called with the statement,

```
call spline (n,x,y,xa,ya)

inputs:
     n    todal data point of input data array x and y
     x    an array containing input data of x
     y    an array containing input data of y
    xa    a given value of x to interpolate
outputs:
     ya   a cubic-spline interpolated value at xa.
```

5.3.7. svd

Subroutine svd implements Singular Value Decomposition, or SVD, to fit a given set of data. In the case of an overdetermined system,SVD produces a solution that is the best approximation in the least-squares sense. In the case of an underdetermined system, SVD produces a solution whose values are smallest in the least-squares sense. Therefore, as an option, SVD is applied in gap filling, interpolation, and extrema extraction. This fortran subroutine is based on that in Numerical Recipes in Fortran (Press et al.,1992). It is called with the statement,

```
call svd (n,m,x,y,xa,ya)
```

inputs:

n	number of data points in input array x and y
m	number of coefficients in fitting function
x	an array containing input data of x
y	an array containing input data of y
xa	a given value of x to interpolate

outputs:

ya	a value derived from the SVD fitting function at xa.

6. FUTURE DEVELOPMENTS

This version of skill assessment software has been tested to assess the skill of water level and current predictions from several NOS modeling forecast systems, including the St. Johns River Forecast System (SJROFS), models used for the CSDL Delaware Bay Model Evaluation Environment, and the East Coast Data Assimilation System (ECDA). The software package will be expanded by CSDL to skill assess water temperature and salinity nowcasts and forecast guidance from NOS modeling forecast systems. In addition, further standardization of the software package is being planned. Finally, a skill assessment software package for water levels and currents in non-tidal regions (e.g. the Great Lakes) is being developed as well.

ACKNOWLEDEGMENTS

The authors would like to acknowledge the support from the chief of CSDL's Marine Modeling and Analysis Programs, Dr. Frank Aikman. The authors respectfully thank all those who participated in developing the NOS standards and approaches to conduct model skill assessment, including the Coast Survey Development Laboratory's Tom Gross, John Kelley, and Richard Schmalz.

REFERENCES

Dennis, R.E., and E.E. Long, 1971. A User's guide to a computer program for harmonic analysis of data at tidal frequencies. NOAA Technical Report, NOS 41, 31pp.

Harris, D.L., A. Pore, and R.A. Cummings, 1965. Tide and tidal current prediction by high speed digital computer, International Hydrographic Review, Vol. XLII, No. 1, 95-103.

Hess, K. W., T. F. Gross, R. A. Schmalz, J. G. W. Kelley, F. Aikman III, E. Wei, and M. S. Vincent, 2003. NOS Standards for Evaluating Operational Nowcast and Forecast Hydrodynamic Model Systems. **NOAA Technical Report** NOS CS 17, 48 pp.

Press, W.H., S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery, 1992. Numerical Recipes in Fortran: the Art of Scientific Computing, 2nd ed. Cambridge University Press, 963 pp.

Zervas, C., 1999. Tidal Current Analysis Procedures and Associated Computer Programs. **NOAA Technical Report** NOS CO-OPS 21, 101 pp.

APPENDIX A. CONTROL FILES

Examples of the two control files used for water level skill assessment for the St. Johns River nowcast/forecast model system are listed with the line number at the beginning of each line

A.1. An Example of my_parameters.ctl

```
#################################################################################
#   BEGINDATE:   the begining date: "yyyy mm dd hh mn"
#   ENDDATE:     the end date:    "yyyy mm dd hh mn"
#   OS           =1, run in Unix, fortran codes are compiled using F90.
#                =0, run in Linux, Frotran codes are compiled using LF95.
#   NTYPE        =0 for non-tidal region; =1 for tidal regions.
#   FACTOR       used for event selection of non-tidal region.
#                =<0.0 hupper=-factor,hlower=factor
#                >=0.0 hupper=mean +factor * SD ,hlower=mean -factor * SD.
#   DBASE        Four options: "NWLON", "PORTS", "GLAKES", and "USGS"
#  KINDAT      =1 for vector data (current speed and direction);
#              =2 for scalar data (water level)
#              =3 for temperature
#              =4 for salinity
#   NCYCLE_T:    number of tide simulation cycles per day, =0 read from a single file
#   NCYCLE_H:    number of hindcast cycles per day, =0 read from a single file
#   NCYCLE_N:    number of nowcast cycles per day, >=1
#   NCYCLE_F:    number of forecast cycles per day, >=1
#   NFDURATION   Forecast duration hours for skill assessment (in hours)
#   DELT:       desired time interval (in minutes) of observation, tide prediction and
#               model outputs.
#   DELT_O:     actual time interval (in minutes) of observation, tide prediction
#   DELT_T:     actual time interval (in minutes) of tidal prediction.
#   DELT_M:     actual time interval (in minutes) of model outputs.
#   CUTOFF:     CUTOFF period (in hours) for Fourier filtering, =0 no filtering
#   IGAPFILL:   control switch of gap filling with interpolation
#               =0, filling with missing value -999.0;
#               =1, filling with interpolation value
#   METHOD:     index of interpolation method
#               0: cubic spline  1:Singular Value Decomposition(SVD);
#   CRITERIA1:  (in hours)means using linear and cubic spline interpolation
#                when gap is less than criteria1
#   CRITERIA2:  (in hours) means using cubic spline or SVD interpolation method
#                when criteria1 < gap < criteria2.
#               fill gaps using missing value -999.0 while gap > criteria2
#   IS:         control model run scenorios. =0,off; =1, on
#              IS(1): Tidal simulation only
#              IS(2): model hindcast
#              IS(3): semi-operational nowcast
#              IS(4): semi-operational forecast
#              IS(5): forecast method comparison:persistence forecast
#              IS(6): forecast method comparison:tidal prediction
#  IPRT:       print switch. =0, no screen output; =1 screen output
#  DELHR:      maximum allowed time difference between high and low (in hours),
#              if small than delhr, eliminate both high and low.
#  DELAMP:     maximum allowed amplitude difference between high and low (in meters),
#              if smaller, eliminate both
#  DELPCT:     maximum allowed fraction of amplitude difference between high and low
#  IOPTA:      option for selecting amplitude criterion
#              =1, delamp=delamp
#              =2, delamp=delpct*(hmax-hmin)
#              =3,delamp=delpct*(average hmax)
#  X1          accepted error criteria for water level (0.15 m),
```

```
#              current (0.26 m/s), temperature (7.5 c),salinity (3.5 psu)
#  X2          accepted error criteria for time (in hours)
#  X11         accepted error criteria for phase (in degrees)
# NCON        = number of constituents to be analyzed by H.A., maximum=37
#*****************************************************************
#          Specify project path names
#*****************************************************************
   HOME1=/disks/NASUSER/azhang/SKILL_OFS
#*****************************************************************
#         Specify required file names
#*****************************************************************

#  station control file name

    STATIONDATA=$HOME1/control_files/CBOFS2_WL_station.ctl

# Tide Simulations
   MODELTIDES=$HOME1/archive/ocean_sta_Nowd_ver18_tide_test_0.nc
# or NCYCLE_T >0
   ARCHIVE_DIR_T=$HOME1/archive
   NAME_TIDECAST="ocean_sta_constdensity"

# Hindcast Simulation
   HINDCAST=$HOME1/archive/ocean_sta_Nowd_ver18_hindcast_ADCIRC9_0.nc
# or NCYCLE_H >0
   ARCHIVE_DIR_H=$HOME1/archive
   NAME_HINDCAST="ocean_sta_Nowd_ver18_hindcast_EC2001_"

# Nowcast and Forecast Runs
   ARCHIVE_DIR=$HOME1/archive
#   NAME_NOWCAST="%Y%m/%Y%m%d%H%M_SJROFS_stationsnow.nc"
#   NAME_FORECAST="%Y%m/%Y%m%d%H%M_SJROFS_stationsfore.nc"

   NAME_NOWCAST="TBOFS_sta_nowcast_%Y%m%d%H.nc"
   NAME_FORECAST="TBOFS_sta_forecast_%Y%m%d%H.nc"

#*****************************************************************
#         Specify required parameters
#*****************************************************************

   BEGINDATE="2000 01 01 00 00"
   ENDDATE="2000 12 31 23 00"
   OS=0
   NTYPE=1
   FACTOR=2.0
   DBASE=NWLON
   KINDAT=2
   NCYCLE_T=1
   NCYCLE_H=1
   NCYCLE_N=4
   NCYCLE_F=4
   NFDURATION=24
   DELT=6
   DELT_O=6
   DELT_M=6
   DELT_T=6
   CUTOFF=0.0
   IGAPFILL=1
   METHOD=1
   CRITERIA1=2
   CRITERIA2=6
   IS[1]=1
   IS[2]=0
```

```
   IS[3]=0
   IS[4]=0
   IS[5]=0
   IS[6]=0
   IPRT=0
   DELHR=2.0
   DELAMP=0.2
   DELPCT=0.03
   IOPTA=3
   X1=0.15   #0.260     ## 0.15 for WL, 0.26 for Currents 3.0 for Temp.; 3.5 for S
   X2=0.50
   X11=22.5
   NCON=23
#*****************************************************************
```

A.2. An Example of Control File *stationdata.ctl*

This control file is used for skill assessment of St. Johns nowcast/forecast model system.

```
8571892 E_892 "Cambridge"
  38.57333 -76.06833 0.0  0.0  0.0
8573364 E_364 "Tolchester"
  39.21333 -76.24500 0.0  0.0  0.0
8574680 E_680 "Baltimore"
  39.26667 -76.57833 0.0  0.0  0.0
8575512 E_512 "Annapolis"
  38.98333 -76.48000 0.0  0.0  0.0
8638863 E_863 "CBBT"
  36.96667 -76.11333 0.0  0.0  0.0
```

APPENDIX B. LISTS OF SHELL SCRIPTS PROGRAMS

All shell scripts are listed as the follows with the line number at the beginning of each line.

B.1. STEPS_SETUP.sh

```
#!/bin/sh
#Name:                  STEPS_SETUP.sh
#purpose:               READ my_parameters.ctl and
#                       set environment variables for STEP*.sh scripts
#
#******************************************************************
#   STEP 1  modify two control files
# *****************************************************************
. my_parameters.ctl
WRK_DIR=$HOME1/work
SCRIPT_DIR=$HOME1/scripts
CTL=$HOME1/control_files
OBS=$HOME1/data/obs
PRD=$HOME1/data/prediction
CONSTANTS_DIR=$HOME1/data/harmonic_con
LOG=$HOME1/log
if [ $OS -eq 1 ]
then
  BIN=$HOME1/binUnix
else
  BIN=$HOME1/binLinux
fi
if test ! -r $WRK_DIR
then
   mkdir -p $WRK_DIR
fi
if test ! -r $LOG
then
   mkdir -p $LOG
fi
if test ! -r $OBS
then
   mkdir -p $OBS
fi
if test ! -r $PRD
then
   mkdir -p $PRD
fi
if test ! -r $CONSTANTS_DIR
then
   mkdir -p $CONSTANTS_DIR
fi
export HOME1 WRK_DIR LOG SCRIPT_DIR BIN SORC_DIR ARCHIVE_DIR CONSTANTS_DIR OBS PRD CTL
export BEGINDATE ENDDATE NCYCLE_T NCYCLE_H NCYCLE_N NCYCLE_F DELT DELT_O DELT_M
export CUTOFF IGAPFILL METHOD CRITERIA1 CRITERIA2
export DBASE KINDAT NCON NTYPE
export STATIONDATA MODELTIDES HINDCAST  NAME_NOWCAST NAME_FORECAST
BEGINDATE=`$BIN/dateformat $BEGINDATE "%Y %m %d 00 00"`
ENDDATE=`$BIN/dateformat $ENDDATE "%Y %m %d 00 00"`
cd $WRK_DIR
```

B.2. SKILLSTEPS.sh

```
#!/bin/sh
#Name:                 SKILLSTEPS.sh
#purpose:              Runs the STEP*.sh scripts in sequence
#
#Output:
#

source STEPS_SETUP.sh

# **********************************************************************
#    Run all the Steps
#************************************************************************
#************************************************************************
#     STEP 2     observation CO-OPS verified 6-minutes water level
#************************************************************************
    $HOME1/scripts/STEP2.sh

#************************************************************************
#     STEP 3     Make tidal predictions
#************************************************************************
    $HOME1/scripts/STEP3.sh

#************************************************************************
#     STEP 4    read model tidal simulation
#************************************************************************
    $HOME1/scripts/STEP4.sh

#************************************************************************
#     STEP 5    read model hindcast simulation
#************************************************************************
    $HOME1/scripts/STEP5.sh

#************************************************************************
#     STEP 6    concatenate model nowcast simulation
#************************************************************************
    $HOME1/scripts/STEP6.sh

#************************************************************************
#     STEP 7    concatenate model forecast simulation
#************************************************************************
    $HOME1/scripts/STEP7.sh

#************************************************************************
#     STEP 8    make persistence forecasts
#************************************************************************
    $HOME1/scripts/STEP8.sh

#************************************************************************
# STEP 9 statistics computation and generate skill assessment score tables
#************************************************************************
    $HOME1/scripts/STEP9.sh

#***********************************************************************
# STEP 10:  conduct harmonical constants comparison
#***********************************************************************
    $HOME1/scripts/STEP10.sh
```

B.3. STEP2.sh

```
#!/bin/sh
#Name:          STEP2.sh
#purpose:       Get Observations using station info in
#               $STATIONDATA
#
#               Output is ../data/obs/$stationname".obs"
#               $stationname is string read from $STATIONDATA
#
# Author:         Aijun Zhang
# Date:           11/20/2004
#Language:        Korn Shell Script
# *****************************************************************
#     STEP 2     observation CO-OPS verified 6-minutes water level
# *****************************************************************
source STEPS_SETUP.sh
  if [ $KINDAT -eq 1 ]
  then
     if [ $DBASE = "PORTS" ]
     then
         $SCRIPT_DIR/get_obs_PORTS.sh
     else
       echo there is no data reader for $DBASE database
     fi
  elif [ $KINDAT -eq 2 ]
  then
    $SCRIPT_DIR/get_WL_verified.sh

  elif [ $KINDAT -eq 3 ]
  then
     if [ $DBASE = "PORTS" ]
     then
         $SCRIPT_DIR/get_obs_PORTS.sh
     elif [ $DBASE = "USGS" ]
     then
         $SCRIPT_DIR/get_TS_USGS.sh
     else
         $SCRIPT_DIR/get_WT_NWLON.sh
     fi

  elif [ $KINDAT -eq 4 ]
  then
      if [ $DBASE = "PORTS" ]
     then
         $SCRIPT_DIR/get_obs_PORTS.sh
     elif [ $DBASE = "USGS" ]
     then
         $SCRIPT_DIR/get_TS_USGS.sh
     else
        echo there is no data reader for $DBASE database
     fi

  fi
```

B.4. get_wl_verified.sh

```
#!/bin/sh
#Name:      et_WL_verified.sh
#purpose:   gets CO-OPS verified water levels from CO-OPS databases
#                     within BEGINDATE and ENDDATE
# Author:             Aijun Zhang
# Date:               11/20/2004
#Language:            Korn Shell Script
#input parameters:    BEGINDATE,ENDDATE,STATIONDATA
# Programs Called:
#       Name           Location              Description
#     refwl.x           $BIN     FORTRAN program to reformat to a standard format
  cd $WRK_DIR
  echo get_WL_verified.sh $BEGINDATE " to " $ENDDATE
  BEGINDATE0=`$BIN/dateformat $BEGINDATE "%Y %m %d 00 00"`
  ENDDATE0=`$BIN/datemath $ENDDATE + 0 0 2 0 0`
  bdate=`$BIN/dateformat $BEGINDATE0 "%Y%m%d"`
  edate=`$BIN/dateformat $ENDDATE0 "%Y%m%d"`
  WGETOUT=`mktemp -q wgetout.XXXXXX`
#  Loop on lines in $STATIONDATA
  case "$DBASE"
  in

    "NWLON")
       TEMPLATE=$CTL/request.template_verified
       if [ $DELT_O = 1 -o $DELT_O = 1.0 ]
       then
         TEMPLATE=$CTL/request.template_verified_hourly
       fi
    ;;
    "PORTS")
       TEMPLATE=$CTL/request.template_verified
       if [ $DELT_O = 1 -o $DELT_O = 1.0 ]
       then
         TEMPLATE=$CTL/request.template_verified_hourly
       fi

    ;;

     "GLAKES")
       TEMPLATE=$CTL/request.template_greatlakes
   esac
  exec 5<&0 <$STATIONDATA
  while read stnid stationname longlabel
  do
      echo StationNames $stnid ":" $stationname ":" $longlabel
      read Latlon
      sed -e s/VSTNIDV/$stnid/ \
          -e s/VBDATEV/$bdate/ \
          -e s/VEDATEV/$edate/  $TEMPLATE > request.GET
      wget -o junk -O output.txt -i request.GET
      perl $HOME1/scripts/notbracket.pl output.txt |tr "/:" " " > junk

      awk ' $11 < 2 { print $2" "$3" "$4" "$5" "$6" "$7  }'  junk | sort -u > $WGETOUT
```

```
     $BIN/refwl.x "$BEGINDATE0" "$ENDDATE0" $WGETOUT $stationname"_msl.6min"

     echo    $stationname"_msl.6min"
     head -1 $stationname"_msl.6min"
     tail -1 $stationname"_msl.6min"

     mv $stationname"_msl.6min" $OBS/$stationname".obs"
     rm -f junk $WGETOUT
  done 3<&-
exit
```

B.5. get_obs_PORTS.sh

```
#!/bin/sh
#Name:                get_obs_PORTS.sh
#purpose:  gets Historic 6-minutes  current, water Temperature, and Salinity at PORTS
#                      within BEGINDATE and ENDDATE
# Author:             Aijun Zhang
# Date:               11/20/2004
#Language:            Korn Shell Script
#input parameters:    BEGINDATE,ENDDATE,STATIONDATA
# Programs Called:
#          Name       Location              Description
#        refwl.x        $BIN   FORTRAN program to reformat to a standard format

#         <SELECT mtype>
#            mtype=7  Air Temperature
#            mtype=8  Barometric Pressure
#            mtype=5  Salinity/Gravity
#            mtype=4  Water Currents
#            mtype=9  Water Level
#            mtype=10 Water Temperature
#            mtype=6  Winds
#         </SELECT>
#  cd $WRK_DIR
  echo getnwlon.sh $BEGINDATE " to " $ENDDATE
  BEGINDATE0=`$BIN/dateformat $BEGINDATE "%Y %m %d 00 00"`
  ENDDATE0=`$BIN/datemath $ENDDATE + 0 0 2 0 0`
  WGETOUT=`mktemp -q wgetout.XXXXXX`
  echo $BEGINDATE0 " to " $ENDDATE0
#  Loop on lines in $STATIONDATA
  index=0
  if [ $KINDAT -eq 1 ]
  then
     mtype=4
  elif [ $KINDAT -eq 2 ]
  then
     mtype=9
  elif [ $KINDAT -eq 3 ]
  then
     mtype=10
  elif [ $KINDAT -eq 4 ]
  then
     mtype=5
  fi
```

```
exec 5<&0 < ../control_files/station_ports.info
N=0
while read stnid stationname
do
   PORTSID[N]=$stnid
   PORTSNAME[N]=$stationname
   (( N = N + 1 ))
done 3<&-
TEMPLATE=$CTL/request.template_PORTS
   exec 5<&0 <$STATIONDATA
   while read stnid stationname longlabel
   do
      echo StationNames $stnid ":" $stationname ":" $longlabel
      read Latlon
         index=0
         while (( index < $N ))
         do
           if [ $stnid =  ${PORTSID[index]} ]
           then
            longlabel=${PORTSNAME[index]}
            echo $stnid ${PORTSNAME[index]}
            break
           fi
           (( index = index + 1 ))
         done
      tbegin=$BEGINDATE0
      tbeginp30=`$BIN/datemath $tbegin + 0 0 30 0 0`

while [ `$BIN/dateformat $tbeginp30 "%Y%m%d%H"` -lt `$BIN/dateformat $ENDDATE0 "%Y%m%d%H"` ]
 do
         echo 'time from ' $tbegin to $tbeginp30
         hourb=`$BIN/dateformat $tbegin "%H"`
         dayb=`$BIN/dateformat $tbegin "%d"`
         monb=`$BIN/dateformat $tbegin "%m"`
         yearb=`$BIN/dateformat $tbegin "%Y"`
         houre=`$BIN/dateformat $tbeginp30 "%H"`
         daye=`$BIN/dateformat $tbeginp30 "%d"`
         mone=`$BIN/dateformat $tbeginp30 "%m"`
         yeare=`$BIN/dateformat $tbeginp30 "%Y"`

         sed -e s/STNAME/$longlabel/ \
             -e s/KINDAT/$mtype/ \
             -e s/monb/$monb/ \
             -e s/dayb/$dayb/ \
             -e s/yearb/$yearb/ \
             -e s/timeb/$hourb/ \
             -e s/mone/$mone/ \
             -e s/daye/$daye/ \
             -e s/yeare/$yeare/ \
             -e s/timee/$houre/ \
             -e s/VEDATEV/$edate/  $TEMPLATE > request.GET
             wget -o junk -O output.txt -i request.GET

           perl $HOME1/scripts/notbracket.pl output.txt |tr "/:" " " >> $WGETOUT
           tbegin=$tbeginp30
           tbeginp30=`$BIN/datemath $tbegin + 0 0 30 0 0`

```

```
         done
         hourb=`$BIN/dateformat $tbegin "%H"`
         dayb=`$BIN/dateformat $tbegin "%d"`
         monb=`$BIN/dateformat $tbegin "%m"`
         yearb=`$BIN/dateformat $tbegin "%Y"`
         houre=`$BIN/dateformat $ENDDATE0 "%H"`
         daye=`$BIN/dateformat $ENDDATE0 "%d"`
         mone=`$BIN/dateformat $ENDDATE0 "%m"`
         yeare=`$BIN/dateformat $ENDDATE0 "%Y"`

            sed -e s/STNAME/$longlabel/ \
                -e s/KINDAT/$mtype/ \
                -e s/monb/$monb/ \
                -e s/dayb/$dayb/ \
                -e s/yearb/$yearb/ \
                -e s/timeb/$hourb/ \
                -e s/mone/$mone/ \
                -e s/daye/$daye/ \
                -e s/yeare/$yeare/ \
                -e s/timee/$houre/ \
                -e s/VEDATEV/$edate/  $TEMPLATE > request.GET
                wget -o junk -O output.txt -i request.GET

         perl $HOME1/scripts/notbracket.pl output.txt |tr "/:" " " >> $WGETOUT

      awk '$1!="#" { print $3" "$1" "$2" "$4" "$5" "$7" "$8  }' $WGETOUT  | sort -u > junk0

     $BIN/reformat_PORTS.x "$BEGINDATE0" "$ENDDATE0" $KINDAT junk0 $stationname"_msl.6min"

      echo    $stationname"_msl.6min"
      head -1 $stationname"_msl.6min"
      tail -1 $stationname"_msl.6min"

      mv $stationname"_msl.6min" $OBS/$stationname".obs"
      rm -f junk junk0 $WGETOUT
   done 3<&-
exit
```

B.5. get_TS_USGS.sh

```
#!/bin/sh
#-------------------------------------------------------------------
#
# Script Name: get_TS_USGS.sh
#
# Abstract:
#  Gets Real-time temperature and salinity data from USGS web page:
#
#            http://waterdata.usgs.gov/md/nwis/uv?01578310
#
#         Request tab separated data and you will see the source file.
#
#         There is no choice about times on this web page, so this only
#         gives you the last SEVEN days of data.
#
#         The script decodes these files to grab the different data
```

```
#          types which might be available.  Not all stations have the
#          same data (or in the same order.)  Possible choices are:
#
#          TEMP        TEMPERATURE, WATER (DEG. C)
#          COND        SPECIFIC CONDUCTANCE (MICROSIEMENS/CM AT 25 DEG. C)
#          DISCHARGE   DISCHARGE, CUBIC FEET PER SECOND
#          GAGE        GAGE HEIGHT, FEET
#
#          The requested page is sent to  READUSGS.pl to parse out the data
#          type requested.
#            produces a ascii file
#            Capable of returning any data variables from any river station
#          Returns ascii like:
#          2003 04 17 00 00  0  6.390000 63700.000000
#          2003 04 17 00 30  0  6.380000 63600.000000
#
#
# Language:  Bourne Shell Script
#
# Scripts/Programs Called:
#    Name       Directory Location             Description
#   READUSGS.pl COMF/oqcs/scripts   Parses out the data from USGS river web page.
#   mktemp.c    /COMF/oqcs/sorc  Makes a temporary unique filename.
#
# ---------------------------------------------------------------------
echo get_TS_USGS.sh $BEGINDATE " to " $ENDDATE
BEGINDATE0=`$BIN/dateformat $BEGINDATE "%Y %m %d 00 00"`
ENDDATE0=`$BIN/datemath $ENDDATE + 0 0 2 0 0`
WGETOUT=`mktemp -q wgetout.XXXXXX`
WGETLOG=`mktemp -q wgetlog.XXXXXX`
RIVSCRATCH=`mktemp -q river1.dat.XXXXXX`
exec 5<&0 <$STATIONDATA
while read stnid stationname longlabel
do
     echo StationNames $stnid ":" $stationname ":" $longlabel
     read Latlon
REQUESTGET="http://waterdata.usgs.gov/nwis/uv?format=rdb&period=31&site_no=$stnid"

wget -o $WGETLOG -O $WGETOUT $REQUESTGET
$HOME1/scripts/READUSGS.pl $WGETOUT "COND TEMP" $RIVSCRATCH
# convert conductivity into specfic cond. 25C.
cat $RIVSCRATCH |   \
awk '{ print $1" "$2" "$3" "$4+5" "$5"  " $6 "  "$7*(1+0.02*($8-25)) "  " $8  }' | sort -u > tmp1
awk '{ print $1" "$2" "$3" "$4" "$5 "  " $8  }' tmp1 | sort -u > temperature.tmp
# Call the PERL script to convert conductivity and temperature to Salinity
#"SALINITY.pl written for conductivity from NWLON, USGS gives temperature"
$HOME1/scripts/SALINITY.pl tmp1 tmp2
awk '{printf("%4d %02d %02d %02d %02d   %10.5f \n",$1,$2,$3,$4,$5,$7)}' tmp2 > junk0
  if [ $KINDAT -eq 3 ]
  then
    $BIN/reformat_USGS.x "$BEGINDATE0" "$ENDDATE0" $KINDAT temperature.tmp tmp.out
  elif [ $KINDAT -eq 4 ]
  then
    $BIN/reformat_USGS.x "$BEGINDATE0" "$ENDDATE0" $KINDAT junk0  tmp.out
  fi

mv tmp.out $OBS/$stationname".obs"
```

```
done
rm junk* tmp*
rm $RIVSCRATCH &> /dev/null
rm $WGETOUT &> /dev/null
rm $WGETLOG &> /dev/null
exit
```

B.6. tide_prediction.sh

```
#!/bin/sh
#Name:                tide_prediction.sh
#purpose:             make tidal predictions between BEGINDATE and ENDDATE
# Author:             Aijun Zhang
#   Coast Survey Development LaboratorySDL, NOS of NOAA
# Date:               11/20/2004
#Language:            Korn Shell Script
#input parameters:    BEGINDATE,ENDDATE, DELT,KINDAT,STATIONDATA
# Programs Called:
#   Name               Location              Description
#  reformat_ha.x       $BIN     Fortran program to reformat harmonic constants
#                               to a standard format
#      pred.x          $BIN     Fortran Program to make tidal predictions
#****************************************************************************
  echo  "run pred.x from $BEGINDATE  to  $ENDDATE"
  ENDDATE0=`$BIN/datemath $ENDDATE + 0 0 2 0 0`
  bdate=`$BIN/dateformat $BEGINDATE "%Y%m%d"`
  edate=`$BIN/dateformat $ENDDATE0 "%Y%m%d"`

# Loop on lines in $stationdata
  exec 5<&0 <$STATIONDATA
 while read stnid stationname longlabel
 do
  read lat longitude XMAJOR ISTA LAYER
   if [ ! -s $CONSTANTS_DIR/$stationname'.std' ]
   then
##   get harmonic constants from CO-OPS

   sed -e s/VSTNIDV/$stnid/ \
       -e s/VBDATEV/$bdate/ \
       -e s/VEDATEV/$edate/  \
     $CTL/request.template_ha > request.GET_ha

    wget -o junk -O $stationname'.ha' -i request.GET_ha
    $BIN/reformat_ha.x  $stationname
    mv $stationname'.ha' $CONSTANTS_DIR
    mv $stationname'.std' $CONSTANTS_DIR
   fi
###   run pred.f
FILEIN=$CONSTANTS_DIR/$stationname'.std'
FILEOUT=$stationname.prd
  $BIN/pred.x "$BEGINDATE" "$ENDDATE0" $KINDAT $DELT $XMAJOR $FILEIN $FILEOUT > /dev/null
  mv $stationname.prd $PRD
 done 3<&-
exit
```

B.7. concatenate_nowcast.sh

```
#!/bin/sh
#Name:                concatenate_nowcast.sh
#purpose:concatenate all netCDF files of the nowcast cycles between BEGINDATE and ENDDATE
# Author:             Aijun Zhang
#   Coast Survey Development LaboratorySDL, NOS of NOAA
# Date:               11/20/2004
#Language:            Korn Shell Script
#inputparameters:BEGINDATE,ENDDATE,ARCHIVE_DIR,NAME_NOWCAST,NCYCLE_N,DELT,KINDAT,STATIONDATA
# Programs Called:
#    Name             Location                Description
#    nday.x           $BIN      FORTRAN program to compute Julian days
#  read_netcdf_now.x  $BIN FORTRAN program to read nowcasts form a netCDF file
#***********************************************************************
  BEGINDATE1=$BEGINDATE
  ENDDATE1=`$BIN/datemath $ENDDATE + 0 0 0 1 0`
  rm -f filename.ctl
  while [ `$BIN/dateformat $BEGINDATE1 "%Y%m%d%H"` -le `$BIN/dateformat $ENDDATE1 "%Y%m%d%H"` ]
  do
      filename=`$BIN/dateformat $BEGINDATE1 $NAME_NOWCAST`
      if [ -s $filename ]
      then
            echo $filename >> filename.ctl
          echo `$BIN/dateformat $BEGINDATE1 "%Y %m %d %H" `>> filename.ctl
      fi

      BEGINDATE1=`$BIN/datemath $BEGINDATE1 + 0 0 0 1 0`
   done

   wc -l filename.ctl > junk
   read N nn < junk
   (( N = N/ 2 ))
   echo $DELT_M $NCYCLE_N $KINDAT $N > file.ctl
   echo $STATIONDATA >> file.ctl
   cat file.ctl filename.ctl > tmp1
   cp tmp1 now_filename.ctl
   rm -f tmp1
   $BIN/read_netcdf_now.x < now_filename.ctl

exit
```

B.8. Concatenate_forecast.sh

```
#!/bin/sh
#Name:                concatenate_forecast.sh
#purpose:concatenate all netCDF files of the forecast cycles between BEGINDATE and ENDDATE
# Author:             Aijun Zhang
# Date:               11/20/2004
#Language:            Korn Shell Script
#input parameters: BEGINDATE,ENDDATE,ARCHIVE_DIR,SUBNAME_FORECAST,NCYCLE_F,DELT,KINDAT,STATIONDATA
# Programs Called:
#          Name         Location           Description
#  read_netcdf_fcst1.x    $BIN    FORTRAN program to read netCDF file and check whether
#                                  it contains 24-hours forecasts.
#  read_netcdf_fcst.x     $BIN FORTRAN program to read 24-hour forecasts form a netCDF file
#***********************************************************************
   ENDDATE1=`$BIN/datemath $ENDDATE + 0 0 0 1 0`
   rm -f filename.ctl
   index=0
while [ `$BIN/dateformat $BEGINDATE "%Y%m%d%H"` -le `$BIN/dateformat $ENDDATE1 "%Y%m%d%H"` ]
do
     BEGINDATE1=$BEGINDATE
     cycle=0
     del0=0
     while (( cycle < $NCYCLE_F))
     do
       BEGINDATE1=`$BIN/datemath $BEGINDATE1 + 0 0 0 $del0 0`
       filename[cycle]=`$BIN/dateformat $BEGINDATE1 $NAME_FORECAST`
      if [ ! -s ${filename[cycle]} ]
      then
         break
      fi
       echo $DELT_M $NCYCLE_F $KINDAT 1 > filetmp.ctl
       echo $STATIONDATA >> filetmp.ctl
       echo ${filename[cycle]} >> filetmp.ctl
       echo `$BIN/dateformat $BEGINDATE1 "%Y %m %d %H" ` >> filetmp.ctl
       $BIN/read_netcdf_fcst1.x < filetmp.ctl
       read dummy < fort.86
       if [ $dummy = 'F' ]
       then
         echo the file does not contain correct data
         break
       fi
       (( del0= 24 / NCYCLE_F ))
       (( cycle = cycle + 1 ))
     done
     if [ $cycle -eq $NCYCLE_F ]
     then
        BEGINDATE1=$BEGINDATE
        del0=0
        cycle=0
        while (( cycle < $NCYCLE_F))
        do
          BEGINDATE1=`$BIN/datemath $BEGINDATE1 + 0 0 0 $del0 0`
          echo ${filename[cycle]} >> filename.ctl
          echo `$BIN/dateformat $BEGINDATE1 "%Y %m %d %H" ` >> filename.ctl
          (( del0= 24 / NCYCLE_F ))
```

```
          (( cycle = cycle + 1 ))
        done
     fi
    (( index = index + 1 ))
   BEGINDATE=`$BIN/datemath $BEGINDATE + 0 0 1 0 0`
  done
  wc -l filename.ctl > junk
  read N nn < junk
  (( N = N / 2 ))
  echo $DELT_M $NCYCLE_F $KINDAT $N > filetmp.ctl
  echo $STATIONDATA >> filetmp.ctl
  cat filetmp.ctl filename.ctl > tmp1
  cp tmp1 fore_filename.ctl
  rm -f tmp1
  $BIN/read_netcdf_fcst.x < fore_filename.ctl
```

B.9. harmonic_analysis.sh

```
#!/bin/sh
#Name:      harmonic_analysis.sh
#purpose:   perform harmonic analysis for the model simulated tidal time series.
# Author:             Aijun Zhang
# Date:               11/20/2004
#Language:            Shell Script
#input parameters: BEGINDATE,ENDDATE,ARCHIVE_DIR, SUBNAME_NOWCAST,NCYCLE_N,DELT,KINDAT
# Programs Called:
#     Name    Location                        Description
#    lsqha.x     $BIN   FORTRAN program for least squares harmonic analysis
#    harm29d.x   $BIN   FORTRAN program for Fourier harmonic analysis for 29 days data
#  table_Harmonic_C.x  FORTRAN program to create constituents comparison tables
#                         between the observed and modeled values
cd $WRK_DIR
exec 5<&0 <$STATIONDATA
while read stnid stationname longlabel
do
    read LAT LONGITUDE XMAJOR ISTA LAYER
    FILEIN=$stationname'_modeltides.dat'
    if [ $IHA -eq 1 ]
    then
       $BIN/lsqha.x $KINDAT  $NCON $DELT_M $LONGITUDE $FILEIN
    elif [ $IHA -eq 29 ]
    then
        $BIN/harm29d.x $KINDAT  $NCON $DELT_M $LONGITUDE $FILEIN
    fi
    if [ -s cons.out ]
    then
       mv cons.out $stationname'_modeltides.std'
    fi
    $BIN/table_Harmonic_C.x $KINDAT  $stationname "$longlabel"
done 3<&-

exit
```

APPENDIX C. 37 TIDAL CONSTITUENTS

These tidal constituents are used for tidal prediction, harmonic analysis, and constituents comparison.

Number	*Name*	*Speed (degrees/hour)*
1	M(2)	28.9841042
2	S(2)	30.0000000
3	N(2)	28.4397297
4	K(1)	15.0410690
5	M(4)	57.9682083
6	O(1)	13.9430351
7	M(6)	86.9523163
8	MK(3)	44.0251732
9	S(4)	60.0000000
10	MN(4)	57.4238319
11	NU(2)	28.5125828
12	S(6)	90.0000000
13	MU(2)	27.9682083
14	2N(2)	27.8953552
15	OO(1)	16.1391010
16	LAMDA(2)	29.4556255
17	S(1)	15.0000000
18	M(1)	14.4966936
19	J(1)	15.5854435
20	MM	0.5443747
21	SSA	0.0821373
22	SA	0.0410686
23	MSF	1.0158958
24	MF	1.0980331
25	RHO(1)	13.4715147
26	Q(1)	13.3986607
27	T(2)	29.9589329
28	R(2)	30.0410671
29	2Q(1)	12.8542862
30	P(1)	14.9589310
31	2SM(2)	31.0158958
32	M(3)	43.4761581
33	L(2)	29.5284786
34	2MK3(3)	42.9271393
35	K(2)	30.0821381
36	M(8)	115.9364166
37	MS(4)	58.9841042

APPENDIX D. EXAMPLES OF WATER LEVEL SKILL ASSESSMENT TABLES

D.1. Comparison of tidal constituent ampltitudes and epochs for water levels. The amplitudes are in meters and the epochs are in degrees.

Station: "Mayport:Bar Pilots, St.Johns River "
Observation:CO-OPS Accepted Harmonic Constants
Model: Least Squares H.A. Beginning 1- 1-1998 at Hour 0.00
Phase is in degrees (GMT)

		Observed		Modeled		Difference	
N	Constituent	Amplitude	Epoch	Amplitude	Epoch	Amplitude	Epoch
1	M(2)	0.676	25.3	0.642	33.4	-0.034	8.1
2	S(2)	0.105	48.3	0.104	57.9	-0.001	9.6
3	N(2)	0.157	7.3	0.142	16.3	-0.015	9.0
4	K(1)	0.084	202.5	0.077	207.8	-0.007	5.3
5	M(4)	0.033	159.4	0.032	188.7	-0.001	29.3
6	O(1)	0.058	210.9	0.055	216.5	-0.003	5.6
7	M(6)	0.009	196.0	0.011	235.6	0.002	39.6
8	MK(3)	0.008	20.4	0.011	36.2	0.003	15.8
9	S(4)	0.005	290.7	0.000	0.0	-0.005	-290.7
10	MN(4)	0.013	156.0	0.013	179.2	0.000	23.2
11	NU(2)	0.032	2.7	0.032	11.8	0.000	9.1
12	S(6)	0.000	0.0	0.000	0.0	0.000	0.0
13	MU(2)	0.012	31.2	0.015	45.7	0.003	14.5
14	2N(2)	0.019	354.6	0.016	13.3	-0.003	-341.3
15	OO(1)	0.004	212.6	0.003	199.8	-0.001	-12.8
16	LAMDA(2)	0.009	47.8	0.012	36.7	0.003	-11.1
17	S(1)	0.011	158.3	0.009	167.3	-0.002	9.0
18	M(1)	0.003	221.2	0.004	213.9	0.001	-7.3
19	J(1)	0.005	210.2	0.005	204.8	0.000	-5.4
20	MM	0.025	230.4	0.028	229.9	0.003	-0.5
21	SSA	0.077	55.4	0.081	55.2	0.004	-0.2
22	SA	0.115	190.2	0.120	190.4	0.005	0.2
23	MSF	0.039	202.7	0.043	204.3	0.004	1.6
24	MF	0.000	0.0	0.000	0.0	0.000	0.0
25	RHO(1)	0.002	214.5	0.002	216.7	0.000	2.2
26	Q(1)	0.011	209.5	0.011	205.6	0.000	-3.9
27	T(2)	0.010	22.1	0.006	59.8	-0.004	37.7
28	R(2)	0.005	291.8	0.001	72.2	-0.004	-219.6
29	2Q(1)	0.002	219.2	0.002	226.2	0.000	7.0
30	P(1)	0.029	202.2	0.026	199.7	-0.003	-2.5
31	2SM(2)	0.003	60.1	0.000	0.0	-0.003	-60.1
32	M(3)	0.006	186.4	0.001	98.9	-0.005	-87.5
33	L(2)	0.041	31.4	0.041	32.9	0.000	1.5
34	2MK3(3)	0.008	44.0	0.004	15.6	-0.004	-28.4
35	K(2)	0.028	48.2	0.031	56.9	0.003	8.7
36	M(8)	0.003	4.2	0.002	94.8	-0.001	90.6
37	MS(4)	0.013	175.8	0.014	199.5	0.001	23.7

D.2. Skill Assessment Table for Water Levels at Mayport

Station: Mayport:Bar Pilots, St.Johns River
Observed Data time period from: / 1/ 2/2003 to / 1/ 1/2004
Data gap is filled using SVD method
Data are filtered using 2.0 Hour Fourier Filter

Variable	X	N	Imax	SM	RMSE	SD	NOF	CF	POF	MDNO	MDPO	WOF
Criterion	-	-	-	-	-	-	<1%	>90%	<1%	<N	<N	<.5%
SCENARIO: TIDAL SIMULATION ONLY												
H			87361	-0.009								
h			87361	0.019								
H-h	15cm	24h	87361	-0.028	0.120	0.117	0.7	78.1	0.0	2.4	0.0	0.00
HHW-hhw	15cm	24h	703	-0.043	0.110	0.101	0.0	80.8	0.0	0.0	0.0	
HLW-hlw	15cm	24h	703	-0.046	0.113	0.103	0.0	78.8	0.0	0.0	0.0	
THW-thw	.5h	25h	703	0.249	0.350	0.247	0.0	82.1	0.0	0.0	0.0	
TLW-tlw	.5h	25h	703	0.169	0.319	0.271	0.0	84.2	0.0	0.0	0.0	
SCENARIO: HINDCAST												
H			87361	-0.141								
h			87361	-0.146								
H-h	15cm	24h	87361	0.005	0.064	0.063	0.0	98.2	0.0	0.0	0.0	0.00
HHW-hhw	15cm	24h	701	-0.023	0.050	0.044	0.0	99.6	0.0	0.0	0.0	
HLW-hlw	15cm	24h	701	0.025	0.055	0.049	0.0	99.3	0.0	0.0	0.0	
THW-thw	.5h	25h	701	-0.072	0.435	0.430	1.4	79.0	0.6	25.0	0.0	
TLW-tlw	.5h	25h	701	0.159	0.450	0.421	0.0	80.5	9.6	0.0	25.0	
SCENARIO: SEMI-OPERATIONAL NOWCAST												
H			87120	-0.001								
h			87120	-0.008								
H-h	15cm	24h	87120	0.007	0.051	0.051	0.0	99.8	0.0	0.1	0.0	0.00
HHW-hhw	15cm	24h	702	-0.031	0.045	0.033	0.0	99.9	0.0	0.0	0.0	
HLW-hlw	15cm	24h	701	0.022	0.026	0.014	0.0	100.0	0.0	0.0	0.0	
THW-thw	.5h	25h	702	0.215	0.265	0.155	0.0	97.2	0.0	0.0	0.0	
TLW-tlw	.5h	25h	701	0.150	0.184	0.108	0.1	99.9	0.0	0.0	0.0	
SCENARIO: SEMI-OPERATIONAL FORECAST												
H00-h00	15cm	24h	1452	0.008	0.057	0.056	0.0	99.9	0.0	0.0	0.0	0.00
H06-h06	15cm	24h	1452	0.009	0.100	0.100	0.1	87.1	0.1	0.0	0.0	0.00
H12-h12	15cm	24h	1452	0.005	0.082	0.082	0.2	93.5	0.2	0.0	0.0	0.28
H18-h18	15cm	24h	1452	0.001	0.110	0.110	0.4	84.2	0.5	0.0	0.0	0.34
H24-h24	15cm	24h	1452	0.002	0.105	0.105	0.5	88.6	0.9	0.0	6.0	0.69
HHW-hhw	15cm	24h	702	-0.021	0.067	0.064	0.0	98.4	0.4			
HLW-hlw	15cm	24h	703	0.012	0.058	0.057	0.0	98.3	0.0			
THW-thw	.5h	25h	702	0.235	0.352	0.262	0.0	83.0	0.1			
TLW-tlw	.5h	25h	703	0.191	0.317	0.254	0.0	87.1	0.3			
COMPARISON: PERSISTENCE FORECAST												
H00-h00	15cm	24h	1452	0.000	0.010	0.010	0.0	100.0	0.0	0.0	0.0	0.00
H06-h06	15cm	24h	1452	0.000	0.082	0.082	0.2	94.0	0.1	0.0	0.0	0.07
H12-h12	15cm	24h	1452	0.000	0.072	0.072	0.1	95.5	0.1	0.0	0.0	0.14
H18-h18	15cm	24h	1452	0.000	0.104	0.104	0.8	86.9	0.7	6.0	0.0	0.62
H24-h24	15cm	24h	1452	0.000	0.098	0.098	0.7	88.6	0.7	6.0	6.0	0.83
HHW-hhw	15cm	24h	703	-0.004	0.059	0.059	0.0	98.4	0.1			
HLW-hlw	15cm	24h	703	0.007	0.062	0.061	0.0	97.7	0.0			
THW-thw	.5h	25h	703	0.004	0.177	0.177	0.1	98.4	0.0			
TLW-tlw	.5h	25h	703	0.043	0.238	0.234	0.0	95.0	0.3			
COMPARISON: ASTRONOMICAL TIDE ONLY												
H-h	15cm	24h	87361	0.008	0.145	0.145	1.4	66.9	1.4	9.7	32.0	0.00
HHW-hhw	15cm	24h	704	0.004	0.124	0.124	0.0	76.6	1.0	0.0	24.7	
HLW-hlw	15cm	24h	703	0.016	0.156	0.155	2.0	62.0	1.3	37.7	37.2	
THW-thw	.5h	25h	704	0.010	0.159	0.159	0.0	99.1	0.0	0.0	0.0	
TLW-tlw	.5h	25h	703	0.036	0.223	0.221	0.0	96.0	0.1	0.0	0.0	

APPENDIX E. EXAMPLES OF CURRENT SKILL ASSESSMENT TABLES

E.1. Comparison of tidal constituent amplitudes and epochs for tidal currents. The amplitudes are in m/s and the epochs are in degrees.

Station: "Intracoastal Waterway Intersection "
Observation: 29-Day H.A. Beginning 4-15-1998 at Hour 17.30
Model: Least Squares H.A. Beginning 1- 1-1998 at Hour 0.00
Phase is in degrees (GMT)

		Observed(R= 0.05)		Modeled(R= 0.002)		Difference	
N	Constituent	Amplitude	Epoch	Amplitude	Epoch	Amplitude	Epoch
CURRENT ALONG PCD		DIR= 113		DIR= 121			
1	M(2)	1.054	187.9	0.942	207.4	-0.112	19.5
2	S(2)	0.143	213.6	0.128	227.2	-0.015	13.6
3	N(2)	0.196	164.1	0.200	185.9	0.004	21.8
4	K(1)	0.129	343.4	0.098	18.4	-0.031	-325.0
5	M(4)	0.050	43.6	0.063	118.9	0.013	75.3
6	O(1)	0.089	357.9	0.076	21.0	-0.013	-336.9
7	M(6)	0.052	17.0	0.006	128.0	-0.046	111.0
8	MK(3)	0.000	0.0	0.025	277.0	0.025	277.0
9	S(4)	0.007	90.0	0.000	0.0	-0.007	-90.0
10	MN(4)	0.000	0.0	0.026	94.2	0.026	94.2
11	NU(2)	0.038	167.3	0.041	181.9	0.003	14.6
12	S(6)	0.005	355.0	0.000	0.0	-0.005	-355.0
13	MU(2)	0.000	0.0	0.010	117.0	0.010	117.0
14	2N(2)	0.026	140.2	0.022	171.1	-0.004	30.9
15	OO(1)	0.004	328.9	0.003	41.8	-0.001	-287.1
16	LAMDA(2)	0.007	199.9	0.027	235.9	0.020	36.0
17	S(1)	0.000	0.0	0.010	326.4	0.010	326.4
18	M(1)	0.006	350.7	0.001	154.5	-0.005	-196.2
19	J(1)	0.007	336.2	0.008	232.7	0.001	-103.5
20	MM	0.000	0.0	0.017	3.5	0.017	3.5
21	SSA	0.000	0.0	0.044	325.4	0.044	325.4
22	SA	0.000	0.0	0.043	326.6	0.043	326.6
23	MSF	0.000	0.0	0.007	93.0	0.007	93.0
24	MF	0.000	0.0	0.010	126.9	0.010	126.9
25	RHO(1)	0.003	4.1	0.004	68.8	0.001	64.7
26	Q(1)	0.017	5.1	0.006	183.4	-0.011	178.3
27	T(2)	0.008	212.6	0.022	204.7	0.014	-7.9
28	R(2)	0.001	214.6	0.014	121.2	0.013	-93.4
29	2Q(1)	0.002	12.3	0.005	243.8	0.003	231.5
30	P(1)	0.043	344.5	0.031	1.4	-0.012	-343.1
31	2SM(2)	0.000	0.0	0.004	101.4	0.004	101.4
32	M(3)	0.000	0.0	0.002	321.4	0.002	321.4
33	L(2)	0.028	164.1	0.056	243.6	0.028	79.5
34	2MK3(3)	0.000	0.0	0.022	280.0	0.022	280.0
35	K(2)	0.039	215.7	0.045	216.8	0.006	1.1
36	M(8)	0.006	61.5	0.004	172.5	-0.002	111.0
37	MS(4)	0.000	0.0	0.018	144.0	0.018	144.0

CURRENT ACROSS PCD DIR= 203 DIR= 211

1	M(2)	0.153	283.4	0.019	301.5	-0.134	18.1
2	S(2)	0.026	323.5	0.004	286.4	-0.022	-37.1
3	N(2)	0.032	257.0	0.008	245.4	-0.024	-11.6
4	K(1)	0.018	53.9	0.003	90.8	-0.015	36.9
5	M(4)	0.107	95.5	0.010	43.9	-0.097	-51.6
6	O(1)	0.007	61.6	0.001	128.3	-0.006	66.7
7	M(6)	0.012	250.7	0.006	277.1	-0.006	26.4
8	MK(3)	0.000	0.0	0.002	151.8	0.002	151.8
9	S(4)	0.003	32.1	0.000	115.3	-0.003	83.2
10	MN(4)	0.000	0.0	0.006	19.0	0.006	19.0
11	NU(2)	0.006	260.5	0.002	310.6	-0.004	50.1
12	S(6)	0.004	281.5	0.000	83.6	-0.004	-197.9
13	MU(2)	0.000	0.0	0.003	200.7	0.003	200.7
14	2N(2)	0.004	230.6	0.002	193.0	-0.002	-37.6
15	OO(1)	0.000	46.2	0.000	271.9	0.000	225.7
16	LAMDA(2)	0.001	302.0	0.002	37.3	0.001	-264.7
17	S(1)	0.000	0.0	0.001	346.6	0.001	346.6
18	M(1)	0.000	57.8	0.000	310.6	0.000	252.8
19	J(1)	0.001	50.1	0.000	37.5	-0.001	-12.6
20	MM	0.000	0.0	0.004	17.6	0.004	17.6
21	SSA	0.000	0.0	0.002	172.8	0.002	172.8
22	SA	0.000	0.0	0.003	69.1	0.003	69.1
23	MSF	0.000	0.0	0.003	28.7	0.003	28.7
24	MF	0.000	0.0	0.001	85.4	0.001	85.4
25	RHO(1)	0.000	64.9	0.001	283.5	0.001	218.6
26	Q(1)	0.001	65.4	0.001	186.8	0.000	121.4
27	T(2)	0.002	321.9	0.000	260.8	-0.002	-61.1
28	R(2)	0.000	325.1	0.000	233.1	0.000	-92.0
29	2Q(1)	0.000	69.3	0.000	206.9	0.000	137.6
30	P(1)	0.006	54.5	0.001	69.8	-0.005	15.3
31	2SM(2)	0.000	0.0	0.000	204.0	0.000	204.0
32	M(3)	0.000	0.0	0.001	212.5	0.001	212.5
33	L(2)	0.005	257.0	0.006	51.2	0.001	-205.8
34	2MK3(3)	0.000	0.0	0.002	59.2	0.002	59.2
35	K(2)	0.007	326.7	0.001	337.7	-0.006	11.0
36	M(8)	0.021	268.8	0.006	63.5	-0.015	-205.3
37	MS(4)	0.000	0.0	0.003	68.4	0.003	68.4

E.2. Skill assessment table for tidal current speed. Current speeds are in m/s.

```
Station:                          Trout River Cut
Observed data time period from: / 7/22/1998  to / 9/16/1998
Data gap is filled using SVD method
Data are filtered using   3.0 Hour Fourier Filter
```

VARIABLE	X	N	IMAX	SM	RMSE	SD	NOF	CF	POF	MDNO	MDPO	WOF
CRITERION	-	-	-	-	-	-	<1%	>90%	<1%	<N	<N	<.5%
SCENARIO: TIDAL SIMULATION ONLY												
U			87361	0.371								
u			87361	0.445								
U-u	26 cm/s	24h	87361	-0.073	0.101	0.069	0.0	99.8	0.0	0.0	0.0	
AFC-afc	26 cm/s	25h	703	-0.113	0.118	0.034	0.0	100.0	0.0	0.0	0.0	
AEC-aec	26 cm/s	25h	692	-0.093	0.099	0.035	0.0	100.0	0.0	0.0	0.0	
TFC-tfc	.5h	25h	703	0.248	0.347	0.242	0.0	82.9	0.0	0.0	0.0	
TEC-tec	.5h	25h	692	-0.496	0.678	0.463	10.1	40.8	0.0	25.0	0.0	
TSF-tsf	.25h	25h	693	-0.461	0.525	0.252	1.6	58.9	0.0	0.0	0.0	
TEF-tef	.25h	25h	696	0.109	0.231	0.203	0.0	96.8	0.0	0.0	0.0	
TSE-tse	.25h	25h	695	-0.077	0.237	0.224	0.0	95.1	0.0	0.0	0.0	
TEE-tee	.25h	25h	695	0.116	0.253	0.225	0.0	93.7	0.0	0.0	0.0	
SCENARIO: HINDCAST												
U			13419	0.500								
u			13419	0.452								
U-u	26 cm/s	24h	13419	0.048	0.139	0.131	0.0	97.3	0.0	0.0	0.0	
AFC-afc	26 cm/s	25h	105	0.158	0.165	0.048	0.0	98.1	0.0	0.0	0.0	
AEC-aec	26 cm/s	25h	88	0.038	0.104	0.097	0.0	100.0	0.0	0.0	0.0	
TFC-tfc	.5h	25h	105	0.356	0.608	0.495	0.0	43.8	4.8	0.0	0.0	
TEC-tec	.5h	25h	88	-0.720	0.876	0.501	31.8	29.5	0.0	37.6	0.0	
TSF-tsf	.25h	25h	98	-0.453	0.512	0.239	1.0	68.4	0.0	0.0	0.0	
TEF-tef	.25h	25h	98	-0.190	0.258	0.176	0.0	98.0	0.0	0.0	0.0	
TSE-tse	.25h	25h	95	0.432	0.466	0.177	0.0	53.7	0.0	0.0	0.0	
TEE-tee	.25h	25h	95	0.256	0.314	0.182	0.0	94.7	0.0	0.0	0.0	
SCENARIO: SEMI-OPERATIONAL NOWCAST												
U			14409	0.496								
u			14409	0.391								
U-u	26 cm/s	24h	14409	0.106	0.250	0.227	0.0	71.5	4.9	0.0	1.6	
AFC-afc	26 cm/s	25h	102	-0.084	0.104	0.061	0.0	100.0	0.0	0.0	0.0	
AEC-aec	26 cm/s	25h	92	0.343	0.347	0.049	0.0	6.5	0.0	0.0	0.0	
TFC-tfc	.5h	25h	102	-0.620	0.784	0.483	20.6	26.5	0.0	0.0	0.0	
TEC-tec	.5h	25h	92	-0.849	0.977	0.486	50.0	18.5	0.0	49.5	0.0	
TSF-tsf	.25h	25h	102	-0.057	0.505	0.505	0.0	72.5	5.9	0.0	24.5	
TEF-tef	.25h	25h	102	-0.428	0.557	0.359	1.0	38.2	0.0	0.0	0.0	
TSE-tse	.25h	25h	103	-0.635	0.759	0.417	8.7	17.5	0.0	36.3	0.0	
TEE-tee	.25h	25h	94	-1.126	1.189	0.385	80.9	14.9	0.0	472.6	0.0	
SCENARIO: SEMI-OPERATIONAL FORECAST												
U00-u00	26 cm/s	24h	240	0.114	0.256	0.229	0.0	72.1	4.6	0.0	0.0	
U06-u06	26 cm/s	24h	240	0.111	0.247	0.221	0.0	70.0	2.5	0.0	0.0	
U12-u12	26 cm/s	24h	240	0.119	0.259	0.230	0.0	68.8	3.8	0.0	0.0	
U18-u18	26 cm/s	24h	240	0.113	0.253	0.227	0.0	69.2	2.9	0.0	0.0	
U24-u24	26 cm/s	24h	240	0.118	0.260	0.232	0.0	67.1	2.9	0.0	0.0	
AFC-afc	26 cm/s	25h	88	-0.067	0.091	0.063	0.0	100.0	0.0	0.0	0.0	
AEC-aec	26 cm/s	25h	98	0.359	0.362	0.048	0.0	2.0	0.0	0.0	0.0	
TFC-tfc	.5h	25h	88	-0.597	0.784	0.511	22.7	27.3	1.1	24.8	0.0	
TEC-tec	.5h	25h	98	-0.648	0.844	0.544	30.6	27.6	0.0	61.6	0.0	
TSF-tsf	.25h	25h	93	0.045	0.523	0.524	0.0	76.3	11.8	0.0	85.4	
TEF-tef	.25h	25h	93	-0.417	0.558	0.373	0.0	37.6	0.0	0.0	0.0	
TSE-tse	.25h	25h	95	-0.574	0.725	0.446	6.3	21.1	0.0	0.0	0.0	
TEE-tee	.25h	25h	88	-1.079	1.149	0.397	78.4	15.9	0.0	223.0	0.0	

COMPARISON: PERSISTENCE FORECAST											
U00-u00	26 cm/s	24h	240	0.001	0.052	0.052	0.0	100.0	0.0	0.0	0.0
U06-u06	26 cm/s	24h	240	0.001	0.101	0.102	0.0	97.5	0.4	0.0	0.0
U12-u12	26 cm/s	24h	240	0.001	0.103	0.103	0.0	96.7	0.4	0.0	0.0
U18-u18	26 cm/s	24h	240	0.002	0.101	0.102	0.0	97.5	0.0	0.0	0.0
U24-u24	26 cm/s	24h	240	0.002	0.104	0.104	0.0	97.5	0.0	0.0	0.0
AFC-afc	26 cm/s	25h	108	0.003	0.075	0.075	0.0	98.1	0.0	0.0	0.0
AEC-aec	26 cm/s	25h	70	-0.001	0.053	0.054	0.0	100.0	0.0	0.0	0.0
TFC-tfc	.5h	25h	108	0.097	0.391	0.380	0.0	81.5	4.6	0.0	24.5
TEC-tec	.5h	25h	70	0.246	0.503	0.442	0.0	68.6	5.7	0.0	0.0
TSF-tsf	.25h	25h	102	0.025	0.394	0.395	1.0	87.3	5.9	0.0	0.0
TEF-tef	.25h	25h	105	0.089	0.353	0.343	0.0	85.7	1.9	0.0	0.0
TSE-tse	.25h	25h	73	0.090	0.376	0.367	0.0	91.8	5.5	0.0	0.0
TEE-tee	.25h	25h	73	0.148	0.368	0.339	0.0	87.7	4.1	0.0	25.0
COMPARISON: ASTRONOMICAL TIDE ONLY											
U-u	26 cm/s	24h	14409	-0.004	0.075	0.075	0.0	99.0	0.0	0.0	0.0
AFC-afc	26 cm/s	25h	116	-0.005	0.053	0.053	0.0	100.0	0.0	0.0	0.0
AEC-aec	26 cm/s	25h	111	0.000	0.033	0.034	0.0	100.0	0.0	0.0	0.0
TFC-tfc	.5h	25h	116	0.101	0.375	0.362	0.0	83.6	3.4	0.0	37.2
TEC-tec	.5h	25h	111	0.255	0.551	0.490	0.9	60.4	7.2	0.0	49.9
TSF-tsf	.25h	25h	114	0.071	0.413	0.409	0.0	84.2	5.3	0.0	13.5
TEF-tef	.25h	25h	115	0.130	0.371	0.349	0.0	87.0	4.3	0.0	24.8
TSE-tse	.25h	25h	115	0.100	0.360	0.347	0.0	83.5	2.6	0.0	11.3
TEE-tee	.25h	25h	115	0.122	0.344	0.323	0.0	87.0	2.6	0.0	11.5

E.3. Skill assessment table for tidal current direction. Current directions are in degrees.

```
Station:                          Trout River Cut
Observed data time period from: / 7/22/1998  to / 9/16/1998
Data gap is filled using SVD method
Data are filtered using   3.0 Hour Fourier Filter
-----------------------------------------------------------------------------------
VARIABLE    X     N   IMAX    SM    RMSE    SD     NOF   CF    POF   MDNO  MDPO WOF
CRITERION   -     -     -      -      -      -     <1%  >90%   <1%    <N    <N  <.5%
-----------------------------------------------------------------------------------

     SCENARIO: TIDAL SIMULATION ONLY
D                     87361  99.655
d                     87361  98.166
D-d      22.5 dg 24h 87361   1.489  6.516  6.344   0.0 100.0   0.0    0.0  0.0
DFC-dfc  22.5 dg 24h   703  -2.144  3.098  2.238   0.0 100.0   0.0    0.0  0.0
DEC-dec  22.5 dg 24h   692   7.988  8.137  1.550   0.0 100.0   0.0    0.0  0.0
     SCENARIO: HINDCAST
D                     13419  91.497
d                     13419 102.247
D-d      22.5 dg 24h 13419  -3.455 13.711 13.269   0.3  97.4   0.0    3.1  0.0
DFC-dfc  22.5 dg 24h   105 -19.299 66.698 64.152   3.8  96.2   0.0   38.1  0.0
DEC-dec  22.5 dg 24h    88   6.927  8.858  5.554   0.0  96.6   0.0    0.0  0.0
     SCENARIO: SEMI-OPERATIONAL NOWCAST
D                     14409 255.207
d                     14409 121.243
D-d      22.5 dg 24h 14409 -20.934 25.148 13.937   0.5  46.1   0.0    1.4  0.0
DFC-dfc  22.5 dg 24h   102  -3.849  5.218  3.540   0.0 100.0   0.0    0.0  0.0
DEC-dec  22.5 dg 24h    92 -33.974 34.167  3.650   0.0   0.0   0.0    0.0  0.0
     SCENARIO: SEMI-OPERATIONAL FORECAST
D00-d00  22.5 dg 24h   240 -21.193 25.626 14.453  94.3   2.5   2.5  114.0  0.0
D06-d06  22.5 dg 24h   240 -21.279 25.660 14.384  93.7   1.3   5.0  114.0  0.0
D12-d12  22.5 dg 24h   240 -21.627 25.833 14.175  94.2   1.3   3.9  114.0  0.0
D18-d18  22.5 dg 24h   240 -21.332 25.657 14.301  93.0   0.6   4.4   96.0  0.0
D24-d24  22.5 dg 24h   240 -21.369 25.677 14.282  94.9   1.3   3.2  114.0  0.0
DFC-dfc  22.5 dg 24h    88  -4.899  6.075  3.612   0.0 100.0   0.0    0.0  0.0
DEC-dec  22.5 dg 24h    98 -33.949 34.132  3.543   0.0   0.0   0.0    0.0  0.0
     COMPARISON: PERSISTENCE FORECAST
D00-d00  22.5 dg 24h   240  -0.526  5.549  5.541  44.2   4.1  43.6   30.0 24.0
D06-d06  22.5 dg 24h   240   0.210 18.808 18.867  51.3   2.5  42.4   24.0 18.0
D12-d12  22.5 dg 24h   240   4.459 25.475 25.159  48.5   1.8  47.2   42.0 12.0
D18-d18  22.5 dg 24h   240   4.065 30.646 30.469  45.1   4.9  46.3   36.0 24.0
D24-d24  22.5 dg 24h   240   2.046 39.987 40.057  50.0   1.2  47.0   42.0 42.0
DFC-dfc  22.5 dg 24h   108  10.581 72.878 72.442   0.9  85.2   4.6    0.0  0.0
DEC-dec  22.5 dg 24h    70  -2.870 38.787 38.960  11.4  72.9  10.0    0.0  0.0
     COMPARISON: ASTRONOMICAL TIDE ONLY
D-d      22.5 dg 24h 14409  -0.476  4.391  4.365   0.0 100.0   0.0    0.0  0.0
DFC-dfc  22.5 dg 24h   116  -0.125  3.240  3.251   0.0 100.0   0.0    0.0  0.0
DEC-dec  22.5 dg 24h   111  -1.455  3.454  3.147   0.0 100.0   0.0    0.0  0.0
-----------------------------------------------------------------------------------
```