KKWave 1.0 User Manual

Jan-Victor Björkqvist and Kimmo K. Kahma

May 2021

Contents

1 Background

KKWave is a parametric wave model that is heavily based on fetch growth relations, but can also to a certain extent account for duration limited growth. A version of this parametric model was used operationally at the Finnish Institute of Marine Research (FIMR) as an operational wave model before numerical models, such as WAM, were viable options. In later years numerical wave models have been proven to be more accurate in open sea areas with resolutions that still allows for reasonable running times. In coastal areas this is not necessarily the case, since a very high resolution (order of 200 m) is required. If decade long statistics are required for a single point, a lighter model, such as KKWave, can produce reasonably accurate predictions in a very reasonable time.

For open sea areas the implementation of the wave model is identical to the one that was used at FIMR by Kimmo Kahma and Heidi Pettersson. The coastal implementation has been added in 2019. The coastal implementation requires some boundary wave information. While these boundary conditions can also be generated with KKWave, it is possible to also use wave measurements of results from a numerical wave model to provide the boundary conditions for the coastal implementation of KKWave. As presented in this manual, the coastal implementation is always calibrated for a single point using some kind of near shore measurements. The calibration is dependent on the type of boundary conditions that are used.

2 Governing physics and propagation scheme

The wave growth in KKWave is estimates based on universal properties of the dimensionless wave energy, $\tilde{\varepsilon}$ and peak frequency, $\tilde{f}_p$, as a function of the dimensionless fetch, $\tilde{X}$, and time, $\tilde{t}$:

$$\tilde{\varepsilon} = \Phi_{\varepsilon}(\tilde{X}, \tilde{t}) \tag{1}$$

$$\tilde{f}_p = \Phi_{f_p}(\tilde{X}, \tilde{t}), \tag{2}$$

where the dimensionless variables are defines as

$$\tilde{\varepsilon} = g^2\varepsilon/U^4 \tag{3}$$

$$\tilde{f}_p = f_p U/g \tag{4}$$

$$\tilde{X} = gX/U^2 \tag{5}$$

$$\tilde{t} = gt/U. \tag{6}$$

Here U is the wind speed, X is the fetch, t is the time, and g is the acceleration caused by gravity.

The basic model for wave growth in KKWave are the fetch limited wave growth relations of Kahma and Calkoen (1992):

$$\tilde{\varepsilon} = p_2\tilde{X}^{p_1} \tag{7}$$

$$\tilde{f}_p = q_2\tilde{X}^{q_1}. \tag{8}$$

The coefficients p_1, p_2, q_1, q_2 in Equations 7 and 8 were given separately for stable and unstable conditions, as well as for an entire "composite data set". These equations (with 8 given in angular frequency) can be found in Kahma and Calkoen (1992) on page 1404 in the conclusions.

The model is expanded to also account for duration limited wave growth using an equivalent fetch and a propagation speed from deep water linear wave theory. The following step explaining the propagation scheme in the model is taken from Appendix A of **?**:

1. Calculate the dimensionless fetch, $\tilde{X}_{\varepsilon^{i-1}}$, based on the energy from the *previous* time step, ε^{i-1}, and the *current* wind speed, U^i.

2. Calculate a (dimensionless) duration based on the wind, U^i, and Eq. 8 as:

$$\tilde{t}_0 = \int_0^{\tilde{X}_{\varepsilon^{i-1}}} \left(\frac{1}{2}\tilde{c_p}\right)^{-1} \mathrm{d}\tilde{X} = \int_0^{\tilde{X}_{\varepsilon^{i-1}}} 4\pi \tilde{f}_p \; \mathrm{d}\tilde{X} = \int_0^{\tilde{X}_{\varepsilon^{i-1}}} 4\pi q_2 \tilde{X}^{q_1} \; \mathrm{d}\tilde{X} = \frac{4\pi q_2}{q_1+1}\tilde{X}_{\varepsilon^{i-1}}^{(q_1+1)}. \tag{9}$$

3. Determine an enhanced dimensionless time step as $\Delta\tilde{t} + \tilde{t}_0$, where Δt is the fixed time step of the model (e.g. 1 h).

4. As an inverse to step 2, calculate a new fetch $\tilde{X}_t$ based on the enhanced duration $\Delta\tilde{t} + \tilde{t}_0$, and Eq. 8 as:

$$\tilde{X}_t = \left(\frac{(\tilde{t} + \tilde{t}_0)(q_1+1)}{4\pi q_2}\right)^{1/(q_1+1)}. \tag{10}$$

5. Calculate the minimum dimensionless fetch for a fully developed sea (using the wind speed at 10 metre height) as:

$$\frac{U}{c_p} = \frac{2\pi f_p U}{g} = 2\pi\tilde{f}_p = 0.82 \Longleftrightarrow \tilde{X}_{FD} = \left(\frac{0.82}{2\pi q_2}\right)^{1/q_1}. \tag{11}$$

6. Set the relevant fetch to $\tilde{X}^i = \min\{\tilde{X}_t, \tilde{X}_{FD}, \tilde{X}\}$, where $\tilde{X}$ is the physical dimensionless fetch.

7. Calculate the new energy, ε^i, and peak frequency, f_p^i, using $\tilde{X}^i$, U^i, and Equations 7 and 8.

3 Traditional open sea implementation

This mode is the original type of wave model that was originally used at FIMR for providing open sea forecasts.

3.1 Single point model

In its simplest form the model is only implemented for one point and the waves are calculated using the propagation scheme outlines in Section 2. This setup is suitable for a small enclosed basin with relatively short fetches, and therefore an absence of swell.

As a minimum requirement for implementing the model the user needs to define the fetches for each wind sector (typically around 10 deg), and provide the model with a time series of the wind speed and wind direction. The model also requires air and water temperature time series, since the stratification (stable of unstable) is determined based on the temperature difference. Nonetheless, if the user uses artificial temperatures of equal values the model will use the growth relation based on the composite data set, which is not expected to have a significant effect on the model performance.

3.2 Adding boundary points

Slightly more nuance can be achieved by adding boundary points from where waves might propagate to the final point of interest. This setup is suitable for points in enclose basins to where waves might propagate from neighbouring basins, and/or the basin is large enough that swell is expected to be non-negligible.

The wind and temperature forcing files are required for each boundary point separately. In addition to the fetches at the boundary points, the user also needs to pre-calculate i) a rough estimate for the propagation time from the boundary points to the location of interest, ii) the estimated attenuation of the wave energy when it travels from the boundary points to the location of interest (because of e.g. directional spread and sheltering), and iii) the valid directions when waves actually propagate to the location of interest (i.e. waves only arrive from a westerly boundary point during westerly winds).

When boundary points are defined, the model automatically calculates the waves for the boundary points first using the same propagation scheme as for the actual site of interest. However, the results from a previous run can also be provided as the boundary conditions if the user so chooses.

4 Coastal implementation

The coastal implementation as it exists in KKWave now was never a part of the original FIMR forecast model. This type of coastal implementation was created for the need of the study conducted by **?**. It is designed to work in a region where we have a local fetch that generates waves, while attenuated waves propagating from the open sea are also present. The implementation essentially requires some near shore measurements that are used to calibrate the model, which happens in four steps:

1. Implement the coastal point as a single point model using only the local fetch, which gives a time series of the simulated locally generated wave variance, ε_{local}.
2. Use an open sea implementation to generate a coinciding wave time series, which will serve as the incident boundary wave field with the variance ε_{bnd}. These wave data can also be taken from any other source if the user so chooses.
3. Use the wave measurement, ε_{obs}, to determine the amount of long open sea wave energy that propagates to the coastal site as $\varepsilon_{long} = \varepsilon_{obs} - \varepsilon_{local}$.
4. Determine attenuation coefficients, $k(U_d)$, for the incident wave energy as a function of the wind direction (sectors) using a linear fit between ε_{long} and ε_{bnd}.

The coastal implementation is then—in addition to the normal forcing data and fetches—provided with the attenuation factors.

5 Calculating wave variables

The peak wave direction is the wind direction, although in turning wind conditions the response is delayed based on a user defined relaxation time. If the fetch geometry is expected to results in a systematic disagreement between the wind and wave directions (Holthuijsen, 1983; Pettersson et al., 2010), the user can provide a table relating the wind direction to the wave direction.

5.1 Peak frequency, f_p

The peak frequency is determined as the peak frequency of the most dominant of the systems: 1) local sea, local swell, or any of the waves propagating from the boundary points.

5.2 Peak wave direction, D_p

The peak wave direction is determined based on the wind direction with pre-determined correction for slanting fetch.

The peak direction is determined as the peak direction of the most dominant of the systems: 1) local sea, local swell, or any of the waves propagating from the boundary points.

5.3 Significant wave height, H_s

In the single point model the significant wave height follows directly from the wave variance obtained using the governing equations and the propagation scheme (see Sect. 2). If the implementation consists of several points—and thus several waves systems—the significant wave height is determines as follows:

For an **open sea implementation** the wave energy, ε, is determined by the governing equations. If f_p from the previous time step is lower then for the current time step, then swell is determined to be present. The swell energy is attenuated by a factor of α_s, and the wind sea energy is multiplied by a factor of $(1-\alpha_s)$ to avoid oversaturation. The energy from these two attenuated systems are added together to form the local wave energy.

The wave energy from the boundary points (determined by a previous run) is attenuated by coefficients β^k, where k is the number of the boundary point. This factor accounts for the attenuation because of spreading, shoreline sheltering, etc. and has to be predetermined. The energy is set to zero if the wave direction is outside of a pre-determined sector where wave propagation from that area is possible. The local energy is multiplied by a factor of $(1-\beta^k)$ to avoid oversaturation. The sum of the energy from different systems are finally calculated.

For a **coastal implementation** the significant wave height is determines using the estimate for the local wave variance and the attenuated wave variance from the boundary point: $H_s = \sqrt{\varepsilon_{local} + k(U_d) \cdot \varepsilon_{bnd}}$ (see Sect. 4 for a description of how the attenuation factors are determined).

6 Example implementations

The automatic weather stations at Harmaja, Kalbådagrund, the Helsinki Lighthouse and Utö, operated by the Finnish Meteorological Institute (FMI), measure the wind speed, wind direction, and air temperature. Harmaja is located 2 km south of the Suomenlinna wave buoy, the Helsinki Lighthouse and Kalbådagrund station is situated in the centre of the GoF, while the Utö weather station is in the northern Baltic Proper (Fig. **??**). The wind sensors at Harmaja, Kalbådagrund, Helsinki Lighthouse and Utö are located 18, 31, 33 and 19 metres above the mean sea level, respectively. For an interested reader, a more extensive overview of the Utö measurements is given by **?**.

Data from Harmaja are available since October 1991. The observations were first saved every third hour, until in 1996 the resolution increased to 30 minutes. From July 1996 the data were saved with 10-20 minute intervals. Since February 2006 the data are available with a 10 minute resolution. The Kalbådagrund data are available every third hour up to February 1997, when hourly measurements began. Since May 2005 the time resolution was 10–40 minutes (10 minutes since October 2007). Data from Utö are available every third hour until June 1998. Denser 10–40 minute data were saved between June 1998 and September 2006, after which the data are available every 10 minutes. For this paper we calculated hourly values from the heterogeneous data, either by interpolating the 3-hour data or by calculating hourly averages of denser measurements. All gaps shorter than six hours were interpolated. Continuous data blocks that were shorter than 12 hours were removed. Data from the Helsinki Lighthouse were only used for the period April – December 2019 when no data from the Kalbådagrund station were availalbe.

Sea surface temperature (SST) data were only available from Harmaja starting end of June 1995. For other times and locations we estimated the SST using a mean yearly cycle from the GoF monitoring station LL7 (Fig. **??**). The mean is defined from 434 CTD soundings from R/V *Aranda* between 1992 and 2017. We processed the air and water temperature data to hourly values in the same fashion as the wind speed data, and interpolated all gaps.

7 Setting up the model

All the forcing data is given in ascii files with the first six columns being the time stamp (e.g. `2015 01 01 00 00 00`). The wind data has eight columns (also wind speed (m/s) and wind direction (deg)). The temperature data has seven columns (temperature in °C).

7.0.1 Pre-processing of forcing data

The forcing data is preprocessed in the following matter:

1. The start time t_0 is established as the first time in the wind input file
2. Data with the time withing 0.5dt of t_0 is picked in the temperature forcing data
3. The time is increased to $t_1 = t_0 + dt$
4. Data with the time withing 0.5dt of t_1 is picked in the wind and temperature forcing data The time is increased to $t_2 = t_1 + \mathrm{d}t$ etc.

As a result the times in the models will have time intervals of exactly dt. If no suitable data is found for the time step t_n, then the value is set to `NaN`.

Name	Type	Description	Default/Typical
Info.folder	string	Folder where KKWave.m is located	pwd
Info.name	string	Name of run (controls output file names)	N/A
Info.dt	double	Model time step (seconds)	1–6 h
Info.relaxtime	double	Wave direction reaction time to wind (seconds)	1–12 h
Info.depth	double	water depth (m)	N/A
Info.BoundaryOutput	logical	Output the calculated boundaries?	1
Info.BoundaryRun	logical	Save boundaries for a nested run?	1
Info.Coastal	logical	Run model as a nearshore point	0
Info.IgnoreBoundary	logical	Run with no boundary waves (used for e.g. nearshore site calibration)	0
Info.BoundaryInput	string	Name of boundary input file if one wants to use previously calculated wave boundaries	N/A

Table 1: Description of the Info-struct that is set in the KKWave_User-file.

7.1 Treatment of boundary conditions

7.2 Determining wave parameters

To use this mode set Info.Coastal=0 in the KKWave_User-file. In this mode KKWave calculates the boundary waves for the areas that have been defined by the Bnd-struct array (see Table 7.2). It is, however, also possible to use previously calculated wave boundaries (set by Info.BoundaryInput).

All of the setting that are required to run the model are set in the KKWave_User-file. The

8 The Gulf of Finland implementation

9 The Suomenlinna implementation

10 Validation

References

Holthuijsen, L. H.
1983. Observations of the Directional Distribution of Ocean-Wave Energy in Fetch-Limited Conditions. *Journal of Physical Oceanography*, 13(2):191–207.

Kahma, K. K. and C. J. Calkoen
1992. Reconciling Discrepancies in the Observed Growth of Wind-generated Waves. *Journal of Physical Oceanography*, 22(12):1389–1405.

Pettersson, H., K. K. Kahma, and L. Tuomi
2010. Wave Directions in a Narrow Bay. *Journal of Physical Oceanography*, 40(1):155–169.

Name	Type	Description	Default/Typical
`Site.name`	string	Name of the site	`Info.name`
`Site.X`	vector	The fetch as a function of the direction	N/A
`Site.Xeff`	vector	The effective fetch accouning for e.g. islands and narrow fetch	N/A
`Site.DX`	vector	The wave direction as a function of the wind direction (accounts for slanting fetch)	N/A
`Site.att`	double	Attenuation factor for waves that have become swell	0.9
`Site.wind_att`	double	Attenuation factor for wind forcing (the model was originally calibrated for area winds)	0.9–1
`Bnd(n).name`	string	Name of the boundary area `n`	N/A
`Bnd(n).X`	vector	The fetch as a function of the direction for boundary area `n`	N/A
`Nesting(n).valid_dir`	1x2 vector	Direction interval (from) when are `n` actually propagates waves `n`	N/A
`Nesting(n).att`	double or vector	The attenuation of the boundary waves for boundary area `n` (can be a single value or direction dependent)	N/A
`Nesting(n).BoundaryDelay`	double	Multiples of `Info.dt` that it takes for waves from boundary `n` to arrive to the site (0=no delay)	Depends on typical T_p and distance

Table 2: Description of the `Site`-struct, the `Bnd`-struct array, and the `Nesting`-struct array that is set in the `KKWave_User`-file.

Name	Type	Description	Default/Typical
`Forcing.wind_file`	String	File name for site wind data	N/A
`Forcing.ta_file`	String	File name for site air temperature data	N/A
`Forcing.tw_file`	String	File name for site water temperature data	N/A
`Bnd_Forcing(n).wind_file`	String	File name for boundary area `n` wind data	N/A
`Bnd_Forcing(n).ta_file`	String	File name for boundary area `n` air temperature data	N/A
`Bnd_Forcing(n).tw_file`	String	File name for boundary area `n` water temperature data	N/A

Table 3: Description of the `Forcing`-struct and `Bnd_Forcing`-struct arrays that is set in the `KKWave_User`-file.