The Sverdrup-Munk-Bretschneider (SMB) Wave Prediction Framework: A Theoretical, Historical, and Computational Manual

1. Introduction and Theoretical Scope

The prediction of wind-generated surface gravity waves constitutes one of the fundamental problems in geophysical fluid dynamics and coastal engineering. The primary objective is to forecast the statistical properties of the sea state—specifically the **Significant Wave Height** (H_s) and the **Significant Wave Period** (T_s)—given a set of meteorological forcing parameters (wind speed, duration) and geometric constraints (fetch length, water depth).

1.1 The Duality of Wave Modeling

In modern oceanography, wave prediction is generally approached through two distinct paradigms:

1. **Spectral (Phase-Averaged) Models:** Third-generation models like **SWAN** (Simulating Waves Nearshore) and **WAVEWATCH III** solve the **Action Balance Equation**:

   $$\frac{\partial N}{\partial t} + \nabla \cdot (\vec{c}_g N) = \frac{S}{\sigma}$$

   Where N is the wave action density, $\vec{c}_g$ is the group velocity vector, and S represents source terms (wind input, quadruplet interactions, whitecapping). While physically rigorous, these models require extensive computational grids and detailed boundary conditions.

2. **Parametric (Empirical) Models:** This software suite implements the parametric approach. Instead of solving the partial differential equation for the entire spectrum, parametric models rely on **Similarity Laws**. They assume that the wave spectrum, under constant wind forcing, maintains a self-similar shape (typically JONSWAP or Pierson-Moskowitz). This assumption allows the state of the sea to be described by simple non-dimensional relationships between total energy, peak frequency, and fetch.

1.2 The Scope of this Software

This repository provides a high-fidelity implementation of the **Sverdrup-Munk-Bretschneider (SMB)** method. Critically, it rejects the flawed equations of the 1984 *Shore Protection Manual* (SPM) in favor of the **Hurdle and Stive (1989) Unified Formulation**.

The Hurdle and Stive revision is mathematically significant because it ensures C^1 **continuity** (smoothness of both the function and its derivative) across the transition from deep water to shallow water. This prevents the generation of non-physical artifacts—such as sudden jumps in predicted wave height—that occur when using legacy lookup tables or piecewise functions.

---

2. Foundations of Ocean Wave Science: The Sverdrup-Munk Era

To understand the mechanics of the SMB method, one must examine the foundational physics established during the urgency of World War II.

2.1 The Energy Budget Hypothesis (1947)

Before 1947, wave forecasting was based on empirical correlations (e.g., Stevenson's formula). **Harald Sverdrup and Walter Munk** revolutionized the field by introducing the **Energy Budget Hypothesis**. They posited that the growth of waves is not merely kinematic but is governed by the conservation of energy flux.

They formulated the governing differential equation for the total wave energy E:

$$\frac{dE}{dt} + \frac{E}{C}\frac{dC}{dt} = R_{Total} - D_{iss}$$

Where:

- $E = \frac{1}{8}\rho g H^2$: The mean energy per unit surface area.
- C: The phase velocity of the significant wave.
- R_{Total}: The net energy transfer from the atmosphere to the sea.
- D_{iss}: Energy dissipation (which they assumed negligible for growing waves until breaking).

2.2 Atmospheric Energy Transfer Mechanisms

Sverdrup and Munk identified two distinct physical mechanisms for wind-to-wave energy transfer, a duality that remains central to fluid dynamics today:

1. **Normal Pressure (R_N):** This mechanism relies on **Form Drag** (or the sheltering effect). As wind blows over a wavy surface, the airflow separates. This creates a zone of high static pressure on the windward face of the crest and a zone of low pressure (suction) on the leeward face.
   - *Physics:* The pressure vector is phase-shifted relative to the surface elevation.

  - *Effectiveness:* This is the dominant growth mechanism for young, steep waves where the wind speed U is much greater than the wave phase speed C ($U \gg C$).

2. **Tangential Stress (R_T):** This represents **Skin Friction**. It is the transfer of momentum via viscous shear stress at the air-sea interface.
   - *Physics:* The wind "rubs" against the water surface, dragging particles forward.
   - *Effectiveness:* This mechanism operates even when waves are moving nearly as fast as the wind, but it is generally less efficient than form drag for rapid growth.

2.3 The Definition of Significant Wave Height

The sea surface is a stochastic, multi-scale phenomenon. To make it tractable for engineering, Sverdrup and Munk defined the **Significant Wave Height** ($H_{1/3}$).

- **Statistical Definition:** The average height of the highest one-third of waves in a zero-upcrossing record.
- **Spectral Definition:** In deep water, assuming a narrow-band spectrum, wave heights follow a **Rayleigh Distribution**. Under this assumption:

$$H_{1/3} \approx 4\sqrt{m_0}$$

  Where m_0 is the variance (total energy) of the sea surface elevation.

---

3. The Evolution of Parametric Forecasting (1950-1984)

The transition from Sverdrup and Munk's theoretical basis to standard engineering curves involved decades of data collection and refinement, culminating in the flawed standardization of the 1980s.

3.1 Bretschneider's Fully Developed Sea (FDS)

C.L. Bretschneider (1958) utilized vast datasets from weather ships to define the upper limits of wave growth. He formalized the concept of the **Fully Developed Sea**.

- **Physical Concept:** FDS is a dynamic equilibrium where the energy input from the wind (R_{Total}) is exactly balanced by the energy dissipation (D_{iss}) due to whitecapping and turbulence.
- **Implication:** Once a sea is fully developed, neither an increase in fetch (distance) nor an increase in duration (time) will produce larger waves. The wave field is saturated.

Bretschneider established the dimensionless asymptotic limit for wave height as:

$$\frac{gH_{1/3}}{U^2} \approx 0.282$$

(Note: This coefficient was later revised to 0.25 by Hurdle & Stive to correct for instrument bias in early data).

3.2 The Shore Protection Manual (1984) Standardization Failures

The 1984 edition of the *Shore Protection Manual* (SPM) became the global standard, but it introduced two critical methodological errors that this software explicitly corrects.

3.2.1 The Adjusted Wind Speed (U_A) Error The SPM attempted to account for the non-linear aerodynamics of the sea surface. The drag coefficient C_D increases with wind speed because the sea surface becomes rougher (more ripples/foam) as wind increases. The SPM defined an "Adjusted Wind Speed" U_A (or wind stress factor) calculated as:

$$U_A = 0.71 \cdot (U_{10})^{1.23}$$

- **The Flaw:** While physically motivated to represent Friction Velocity (u_*), this factor was applied to equations that had already been calibrated against raw wind speeds.
- **Consequence:** As noted by Bishop, Donelan, and Kahma (1992), this led to "double-counting" the non-linearity. The result was a systematic **overprediction** of wave heights by 10-20% and wave periods by 5-10% in high wind conditions.

3.2.2 The Mathematical Discontinuity The SPM provided three separate sets of equations for Deep, Intermediate, and Shallow water.

- **Deep Water:** Dependent only on Wind and Fetch.
- **Shallow Water:** Heavily dependent on Depth (d).
- **The Singularity:** At the transition depth (where the deep and shallow curves should meet), the equations often yielded different values.
  - *Example:* Calculating for a 50km fetch at 20m/s wind might yield $H_s = 4.2m$ using the deep water formula, but $H_s = 3.8m$ using the shallow water formula at a depth of 200m (which should be effectively deep).
  - *Impact:* This step-change forced engineers to arbitrarily smooth data or choose a "conservative" (often overly expensive) design value.

The **Hurdle and Stive (1989)** formulation used in this software resolves this by using a single, unified equation that asymptotically approaches the deep water limit as $d \to \infty$ and the shallow water limit as $d \to 0$.

4. The Hurdle and Stive (1989) Unification

To resolve the critical mathematical discontinuities and physical inconsistencies inherent in the 1984 *Shore Protection Manual* (SPM), **Hurdle and Stive (1989)** proposed a unified, continuous formulation. This revision constitutes the theoretical backbone of the `calculator_gui.cpp` and `calculator.py` tools provided in this repository.

4.1 The Mathematical Imperative: Solving the Piecewise Singularity

Legacy models (SPM 1984) relied on a **piecewise approach**, utilizing distinct, non-matching power laws for deep water and shallow water regimes. This created a mathematical singularity at the "transition depth"—the point where the deep water and shallow water curves theoretically intersect. In practice, they often did not intersect, leading to a "jump" in predicted wave height where $H_s(d) \neq H_s(d + \epsilon)$.

Hurdle and Stive solved this by applying the **Hyperbolic Tangent** (tanh) **Transformation**. The hyperbolic tangent function is mathematically ideal for coastal engineering because of its asymptotic properties:

- **Behavior at ∞:** $\lim_{x\to\infty} \tanh(x) = 1$ (Representing Deep Water, where depth does not limit growth).
- **Behavior at 0:** $\lim_{x\to 0} \tanh(x) \approx x$ (Representing Shallow Water, where growth is linearly constrained by depth).

By nesting these functions, Hurdle and Stive derived a single set of equations that are C^1 **continuous** (continuous in both value and first derivative) across the entire domain of fluid depth.

4.2 The Unified Fetch-Limited Equations

The core of the software's physics engine is based on the following dimensionless equations. These calculate the energy (height) and dispersion (period) based on the dimensionless fetch ($\hat{F}$) and dimensionless depth ($\hat{d}$).

4.2.1 Significant Wave Height (H_s)

$$\frac{gH_s}{U_A^2} = A_H \cdot \tanh(K_{d1}) \cdot \tanh^{0.5}\left[\frac{B_H \cdot \hat{F}}{\tanh^2(K_{d1})}\right]$$

Variable Definitions:

- $\hat{d} = \frac{gd}{U_A^2}$: Dimensionless Depth.
- $\hat{F} = \frac{gF}{U_A^2}$: Dimensionless Fetch.
- $K_{d1} = 0.6(\hat{d})^{0.75}$: The depth-limiting wavenumber proxy.
- $A_H = 0.25$: The asymptotic deep-water stability coefficient.

- $B_H = 4.3 \times 10^{-5}$: The deep-water growth rate.

Physical Insight - The Damping Term: The term $\tanh(K_{d1})$ acts as a **friction valve.**

- **Deep Water:** As $\hat{d} \to \infty$, $K_{d1} \to \infty$, and $\tanh(K_{d1}) \to 1$. The equation simplifies to the pure wind-forcing power law: $H_s \propto \hat{F}^{0.5}$.
- **Shallow Water:** As $\hat{d} \to 0$, $\tanh(K_{d1})$ decreases. This suppresses the effective fetch inside the brackets and clamps the maximum possible wave height outside the brackets, physically simulating energy loss due to bottom friction and percolation.

4.2.2 Significant Wave Period (T_s)

$$\frac{gT_s}{U_A} = A_T \cdot \tanh(K_{d2}) \cdot \tanh^{1/3}\left[\frac{B_T \cdot \hat{F}}{\tanh^3(K_{d2})}\right]$$

Variable Definitions:

- $K_{d2} = 0.76(\hat{d})^{0.375}$: The dispersive depth-limiting term.
- $A_T = 8.3$: The asymptotic deep-water period coefficient.
- $B_T = 4.1 \times 10^{-5}$: The deep-water period growth rate.

4.3 Key Advantages of the Unified Model

1. Elimination of Step Changes (Artifact Removal) By utilizing the nested tanh structure, the software ensures that a calculation performed at 10.0m depth and one performed at 10.1m depth yields results that differ only infinitesimally. This removes the "design anomalies" common in SPM-based tools, where a slight change in input depth could trigger a regime switch and a drastic jump in design wave height.

2. Recalibration of Coefficients (Correction of Bias) Hurdle and Stive performed a re-analysis of the underlying datasets (including the JONSWAP and Bretschneider data). They concluded that the SPM 1984 systematically overpredicted wave energy. This overprediction was largely due to the "double-counting" of wind stress non-linearity (using U_A on curves already calibrated to U_{10}).

To correct this, they adjusted the leading coefficients:

- **Height Coefficient:** Reduced from **0.283** (SPM) to **0.25** (H&S). This reduces predicted wave heights by approximately 12% in fully developed seas.
- **Period Coefficient:** Increased from **7.54** (SPM) to **8.3** (H&S). This correction is crucial because wave period drives the calculation of wavelength and orbital velocity; an underprediction of period (as in SPM) can lead to dangerous underestimations of bottom scour velocities.

3. Kinematic Consistency in Time-Domain (Duration) A fundamental law of wave mechanics is that the fetch distance (F) covered by a wave field is the integral of its group velocity (c_g) over time (t):

$$F(t) = \int_0^t c_g(t')\,dt'$$

In shallow water, the group velocity decreases as depth decreases ($c_g \to \sqrt{gd}$). The legacy SPM equations for duration-limited growth violated this integral constraint in shallow water, implying waves were traveling faster than physically possible.

Hurdle and Stive derived their duration-limited equations by **inverting the fetch-limited equations** and strictly enforcing the relationship between group velocity and energy density. This ensures that the software's duration-limited predictions are kinetically valid—a wave field will never be predicted to traverse a distance faster than its own group velocity allows.

---

5. Computational Physics Engine: Detailed Derivation

The following sections detail the exact formulas, logic, and computational implementation used in the provided software. The physics engine relies on dimensionless analysis, scaling all variables by the gravitational acceleration g ($9.8066\,\mathrm{m/s}^2$) and the wind stress factor U_A.

5.1 Atmospheric Forcing: The Wind Stress Factor

The driving force for wave generation in the model is the wind stress acting on the sea surface. The software performs the mandatory conversion of input wind as the first step in the calculation pipeline.

Formula:

$$U_A = 0.71 \cdot (U_{10})^{1.23}$$

- **Input:** U_{10} (Wind speed at 10m elevation, m/s).
- **Output:** U_A (Adjusted wind speed, m/s).
- **Physics:** This power law reflects that as wind speed increases, surface roughness enhances aerodynamic drag. A doubling of wind speed results in more than a doubling of stress ($2^{1.23} \approx 2.34$).

5.2 Dimensionless Analysis and Scaling

To generalize the wave growth relationships across different scales, the system is non-dimensionalized using three fundamental groups:

1. **Dimensionless Fetch ($\hat{F}$):**

$$\hat{F} = \frac{gF}{U_A^2}$$

2. **Dimensionless Depth ($\hat{d}$):**

$$\hat{d} = \frac{gd}{U_A^2}$$

3. **Dimensionless Duration ($\hat{t}$):**

$$\hat{t} = \frac{gt}{U_A}$$

These groups represent the ratios of inertial forces to gravity and wind stress forces. For example, a small $\hat{d}$ indicates shallow water where bottom friction significantly inhibits growth, while a large $\hat{d}$ indicates deep water where the bottom is negligible.

5.3 Unified Wave Growth Equations (Fetch-Limited)

The core of the physics engine is the implementation of the Hurdle & Stive (1989) unified model. These equations predict the wave characteristics when the fetch (distance) is the limiting factor for growth.

5.3.1 Significant Wave Height (H_s) The dimensionless significant wave height is calculated using a nested hyperbolic tangent structure:

$$\frac{gH_s}{U_A^2} = A_H \cdot \tanh(K_{d1}) \cdot \tanh^{0.5}\left[\frac{B_H \cdot \hat{F}}{\tanh^2(K_{d1})}\right]$$

Components:

- $A_H = 0.25$: The revised asymptotic limit coefficient.
- $B_H = 4.3 \times 10^{-5}$: The growth rate coefficient.
- $K_{d1} = 0.6 \cdot (\hat{d})^{0.75}$: The depth-limiting wavenumber term.

Asymptotic Analysis:

- **Deep Water Limit ($\hat{d} \to \infty$):** The term K_{d1} becomes large, and $\tanh(K_{d1}) \to 1$. The equation simplifies to $\frac{gH_s}{U_A^2} = 0.25 \cdot \tanh^{0.5}(4.3 \times 10^{-5} \cdot \hat{F})$.
- **Shallow Water Limit ($\hat{d} \to 0$):** As depth decreases, $\tanh(K_{d1})$ decreases, acting as a damping factor that restricts the maximum possible wave height regardless of how large the fetch $\hat{F}$ becomes. This physically represents energy dissipation due to bottom friction.

5.3.2 Significant Wave Period (T_s) The unified equation for the significant wave period follows a similar structure but with different exponents to reflect the physics of wave dispersion:

$$\frac{gT_s}{U_A} = A_T \cdot \tanh(K_{d2}) \cdot \tanh^{1/3}\left[\frac{B_T \cdot \hat{F}}{\tanh^3(K_{d2})}\right]$$

Components:

- $A_T = 8.3$: The revised asymptotic limit coefficient (replacing SPM's 7.54).
- $B_T = 4.1 \times 10^{-5}$: The growth rate coefficient.
- $K_{d2} = 0.76 \cdot (\hat{d})^{0.375}$: The depth-limiting wavenumber term.

5.4 Duration-Limited Growth and the t_{min} Switch

In many real-world meteorological scenarios, the wind field does not persist long enough for waves to travel the entire fetch length. In such cases, the sea state is **duration-limited** rather than fetch-limited. The software implements a robust switching logic to handle this.

5.4.1 Minimum Duration Approximation (t_{min}) Calculating t_{min} analytically requires inverting the complex growth equations, which is computationally expensive. To solve this, the software utilizes a modern logarithmic approximation derived by **Etemad-Shahidi et al. (2009)**. This approximation provides a high-accuracy fit for the SMB/CEM curves.

$$L_{fetch} = \ln(\hat{F})$$

$$\text{Exponent} = \sqrt{0.0161 \cdot L_{fetch}^2 - 0.3692 \cdot L_{fetch} + 2.2024} + 0.8798 \cdot L_{fetch}$$

$$\frac{gt_{min}}{U_A} = 6.5882 \cdot \exp(\text{Exponent})$$

5.4.2 Duration-Limited Formulas If the user-input duration $t_{actual} < t_{min}$, the system uses time-growth equations adapted to match the Hurdle & Stive asymptotes (0.25 and 8.3). This ensures that if the duration were extended to infinity, the result would converge exactly to the fully developed sea state predicted by the fetch equations.

- **Significant Wave Height (H_s):**

$$\frac{gH_s}{U_A^2} = 0.25 \cdot \tanh\left[0.000528 \cdot (\hat{t})^{0.75}\right]$$

- **Significant Wave Period (T_s):**

$$\frac{gT_s}{U_A} = 8.30 \cdot \tanh\left[0.003790 \cdot (\hat{t})^{0.41}\right]$$

Coefficient Origin: The coefficients 0.000528 and 0.003790 are specific calibration constants derived to maintain consistency between the time-domain growth rates (typically proportional to $t^{0.75}$) and the spatial domain growth rates.

5.5 Controlling Factor Logic

A robust feature of the software implementation is the logic used to determine the final sea state. Rather than forcing the user to pre-determine the limiting factor, the software calculates **potentials**:

1. Calculate $H_{s,fetch}$ (Assuming infinite time).
2. Calculate $H_{s,duration}$ (Assuming infinite fetch).
3. **Determine Controlling Condition:**

$$H_{s,final} = \min(H_{s,fetch}, H_{s,duration})$$

This `min()` function ensures the model is physically conservative: a wave cannot grow larger than the fetch allows, nor larger than the duration allows.

---

6. Computational Numerical Methods

6.1 Exact Linear Dispersion (Newton-Raphson)

Once T_s is determined, the software calculates the Wavelength (L). This requires solving the **Linear Dispersion Relation**:

$$L = \frac{gT^2}{2\pi} \tanh\left(\frac{2\pi d}{L}\right)$$

This is a **transcendental equation**. While many tools use approximations (e.g., Fenton, 1988), this software implements an **exact numerical solution** using the **Newton-Raphson method**.

Algorithm Steps:

1. Define dimensionless wavenumber $kh = \frac{2\pi d}{L}$.
2. Rearrange dispersion relation: $\frac{\omega^2 d}{g} = kh \tanh(kh)$, where $\omega = 2\pi/T$.
3. Define the root-finding function: $f(kh) = kh \tanh(kh) - Y$, where $Y = \frac{\omega^2 d}{g}$.
4. Derive the gradient: $f'(kh) = \tanh(kh) + kh \cdot \operatorname{sech}^2(kh)$.
5. Iterate: $kh_{n+1} = kh_n - \frac{f(kh_n)}{f'(kh_n)}$ until error $< 10^{-6}$.

This ensures maximum precision across all depth regimes, from deep water ($kh > \pi$) to shallow water ($kh < \pi/10$).

6.2 The Miche Stability Criterion

To ensure physical sustainability, the model applies the **Miche Criterion** to check for wave breaking.

$$\left(\frac{H}{L}\right)_{max} = 0.142 \cdot \tanh(kd)$$

- **Deep Water Limit:** $H/L \leq 0.142 \approx 1/7$.
- **Shallow Water Limit:** $H/d \approx 0.78$ (breaking index).

If the predicted H_s violates this limit, the software caps the wave height and flags the condition as **"BREAKING / UNSTABLE"**.

---

7. Software Reference: Script Descriptions and Usage

This section provides a detailed breakdown of every script in the repository, explaining its purpose, internal logic, and usage.

7.1 `calculator.py`: The Interactive Core

This is the primary Python implementation of the physics engine.

- **Purpose:** Provides a text-based, interactive environment for single-point calculations. It is ideal for rapid sensitivity analysis (e.g., "What if the fetch increases by 5km?").
- **Class `Tee`:**
  - **Description:** A utility class that mimics the Unix `tee` command.
  - **Functionality:** It overrides `sys.stdout`. Every `print()` statement in the script is simultaneously written to the console and to a log file named `report.txt`. This creates an automatic audit trail of all calculations performed during a session.
- **Logic Flow:**
  1. **Input:** Prompts user for Wind Speed (U_{10}), Fetch, Duration, and Depth.
  2. **Conversion:** Calculates Adjusted Wind Speed (U_A).
  3. **Dual Calculation:** Computes *both* Fetch-Limited potential and Duration-Limited potential using the Hurdle & Stive formulas.
  4. **Minimization:** Selects the smaller of the two heights as the controlling case.
  5. **Equivalent Fetch:** If duration-limited, it reverse-calculates the "Equivalent Fetch" (the distance required to generate that specific wave height).
- **Usage:** Run `python calculator.py` in a terminal.

7.2 `calculator_gui.cpp`: The High-Performance Application

This is the C++ implementation designed for Windows.

- **Purpose:** A graphical user interface (GUI) tool for end-users who prefer not to use the command line.
- **Class `SMBEngine`:**
  - **Description:** A stateless physics engine class.
  - **Methods:** Contains static methods like `calculate_adjusted_wind_speed`, `calculate_deep_water`, and `solve_dispersion`. Being stateless ensures thread safety.
- **Threading Model:**
  - **Description:** The application uses a worker thread (`std::thread`) to perform the calculations.
  - **Reasoning:** The Newton-Raphson solver (`solve_dispersion`) is iterative. Running this on the main UI thread could cause the window to freeze. Offloading it to a worker thread ensures the application remains responsive.
- **Class `StringLogger`:** Handles the formatting of the output report, ensuring decimal precision and unit alignment.
- **Compilation:** Requires a C++17 compliant compiler (GCC/MinGW) with static linking flags: `g++ -O3 -std=c++17 -static -static-libgcc -static-libstdc++ -o calculator_gui.exe calculator_gui.cpp -mwindows -lgdi32`.

7.3 `smb_chart_deep.py` and `smb_chart_10m.py`

These scripts generate engineering contour charts.

- **Libraries:** `numpy` (for mesh generation), `matplotlib` (for plotting).
- **Logic:**
  1. **Grid Generation:** Creates a meshgrid of Wind Speed (0-40 m/s) and Fetch (0-200 km).
  2. **Vectorized Calculation:** Applies the SMB formulas to the entire grid simultaneously to compute matrices for H_s, T_s, and t_{min}.
  3. **Contouring:** Uses `plt.contour` to draw isolines:
     - **Solid Black:** Significant Wave Height (H_s).
     - **Dashed Black:** Significant Wave Period (T_s).
     - **Dotted Black:** Minimum Duration (t_{min}).
- **Distinction:** `smb_chart_deep.py` assumes infinite depth ($\tanh(K_d) = 1$), while `smb_chart_10m.py` applies the shallow water damping functions for a fixed depth of 10 meters.

7.4 `smb-nomogram-deep.py` and `smb-nomogram-shallow.py`

These scripts generate professional alignment charts (Nomograms).

- **Libraries:** `pynomo` (nomogen), `pyx` (PostScript rendering).

- **Methodology (Type 9 Nomogram):**
  - These scripts solve the alignment problem for the equation $f(u) + g(v) = h(w)$.
  - They define the functions for Wind, Fetch, and Wave Height/Period.
  - The `nomogen` library numerically optimizes the position and curvature of the scales so that a straight line drawn across the three axes connects mathematically consistent values.
- **Isopleth:** The scripts automatically draw a sample "isopleth" (a grey reference line) representing a specific test case (e.g., Lake Garda: 25 m/s wind, 45 km fetch) to demonstrate how to read the chart.

7.5 `tables.py`

This script generates a printable PDF lookup table.

- **Libraries:** `reportlab` (PDF generation).
- **Logic:**
  - Iterates through three nested loops:
    1. **Depths:** [5m, 10m, 25m, 50m, Deep].
    2. **Wind Speeds:** [10, 15, 20, 25, 30 m/s].
    3. **Fetches:** [5, 10, 25, 50 km].
  - Calculates the full wave state (H_s, T_s, t_{min}) for every combination.
  - Formats the data into a grid and draws it onto a PDF canvas, creating a multi-page reference document "tables.pdf".

---

8. Bibliography

This section cites the foundational texts and revisions referenced throughout this document.

1. **Fenton, J.D. (1999).** "Numerical methods for nonlinear waves." In P.L.-F. Liu (Ed.), *Advances in Coastal and Ocean Engineering* (Vol. 5, pp. 241–324). World Scientific: Singapore.
2. **U.S. Army Corps of Engineers (2008).** *Coastal Engineering Manual (CEM).* Engineer Manual 1110-2-1100, Washington, D.C.
3. **U.S. Army Corps of Engineers (1984).** *Shore Protection Manual (SPM).* Volume 1, Coastal Engineering Research Center, Vicksburg, MS.
4. **World Meteorological Organization (2018).** *Guide to Wave Analysis and Forecasting.* WMO-No. 702.
5. **Sverdrup, H.U., and Munk, W.H. (1947).** *Wind, Sea, and Swell: Theory of Relations for Forecasting.* H.O. Pub. No. 601, U.S. Navy Hydrographic Office.
6. **Bretschneider, C.L. (1952).** "Revised Wave Forecasting Relationships." *Proceedings of the 2nd Conference on Coastal Engineering*, ASCE.

7. **Bretschneider, C.L. (1958).** "Revisions in Wave Forecasting: Deep and Shallow Water." *Proceedings of the 6th Conference on Coastal Engineering*, ASCE.
8. **Hurdle, D.P., and Stive, R.J.H. (1989).** "Revision of SPM 1984 Wave Hindcast Model to Avoid Inconsistencies in Engineering Applications." *Coastal Engineering*, 12, 339-351.
9. **Bishop, C.T., Donelan, M.A., and Kahma, K.K. (1992).** "Shore Protection Manual's Wave Prediction Reviewed." *Coastal Engineering*, 17, 25-48.
10. **Etemad-Shahidi, A., Kazeminezhad, M.H., and Mousavi, S.J. (2009).** "On the Prediction of Wave Parameters Using Simplified Methods." *Journal of Coastal Research*, SI 56, 505-509.
11. **Miche, M. (1944).** "Mouvements ondulatoires de la mer en profondeur constante ou décroissante." *Annales des Ponts et Chaussées.*
12. **Fenton, J.D., and McKee, W.D. (1990).** "On Calculating the Lengths of Water Waves." *Coastal Engineering*, 14, 499-513.