Technical Documentation

Pipeline Imperfection Extractor Program

1.   Introduction

This application is designed to help pipeline engineer to easily extract all "eligible" imperfection geometry within a set of pipeline profile data. This application processes pipeline profile data to identify and analyze key geometric features such as peaks, valleys, and related coordinates to be processed further to extract and populate all important parameter of imperfections. It uses *Python* and integrates libraries such as *tkinter* for the graphical user interface (GUI), pandas for data handling, and *numpy* for numerical computations.

The end results of this application are useful to identify which imperfection is prone to upheaval buckling by comparing the allowed configuration with detected imperfections.

2.   Key Features

- **User Interface:**
  A simple GUI built with *tkinter* allows users to upload data files, process them, and save results.

- **Peak and Valley Detection:**
  Identifies significant peaks and valleys in raw pipeline profile data and saves the results in an organized format.

- **Imperfection Detection:**
  Analyzes sorted peak and valley data to detect geometric imperfections, calculate their height and length, and identify relevant coordinates.

- **Data Export:**
  Allows users to save the detected peaks, valleys, and imperfections in Excel format for further use.

- **Error Handling:**
  Includes robust error messages for invalid input files, empty data, or unsupported formats.
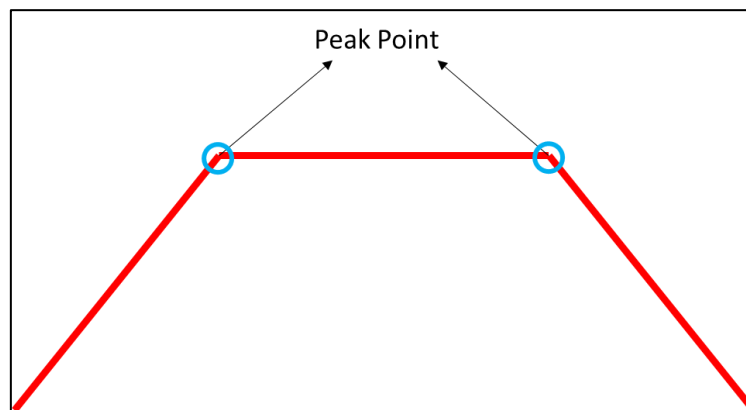
3.   Assumptions for Calculations

- **Peak and Valley Detection:**
  A peak is defined as a point higher than its neighbours, while a valley is lower than its neighbours. Beside the normal peak and valley, the algorithm also considered "plane" condition as illustrated in the following figure:
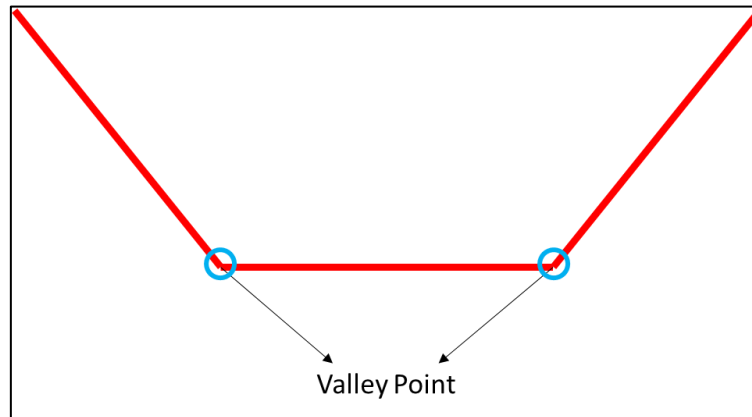
Figure 1 Plane Valley Illustration

Figure 2 Plane Peak Illustration

Plane valley is considered in the algorithm by locating a point with one higher neighbour and one equal neighbour. While Plane Peak is considered by locating a point with one lower neighbour and one equal neighbour

- **Imperfection Detection:**

  Imperfections are determined by analyzing sequences of peaks and valleys:

  1. **Rule 1, Normal Imperfection:**

     A valley → peak → valley sequence is considered as normal imperfection. In this configuration the imperfection height will be calculated as the minimum elevation difference between peak and each valley. To calculate the imperfection width, the application will handle an initial check to identify which side of valley is resulting in minimum height, and then start interpolating to find X coordinate in the other side to form an ideal imperfection configuration as assumed in OTC 6335.

     This rule is excluding cases where the valley's elevation equals the peak's elevation or "stairway" configuration.

  2. **Rule 2, Plateau Imperfection:**

     A valley → peak → peak → valley sequence is considered as plateau imperfection. The calculation for imperfection height and width is typical with Rule 1, but this rule will identify midpoint between two peaks as the peak coordinate.

4. **Program Workflow**

   - **Data Format Requirements:**
     - **X-coordinates**: Horizontal position of the data points. This value shall be positioned starting at cell B2 downward
     - **Elevations**: Vertical position or height. This value shall be positioned starting at cell C2 downward.

   - **User Input:**
     - Upload raw pipeline profile data or pre-sorted data.
     - Ensure the input file complies with the required format.

   - **Peak and Valley Detection:**
     - Extracts X-coordinates and elevations from the uploaded file.
     - Identifies significant peaks and valleys.
     - Saves the results to sheet 1 of result Excel file.

- **Imperfection Detection:**

  o   Analyzes sequences of peaks and valleys to detect imperfections.

  o   Calculates imperfection height and width.

  o   Saves the results, including midpoint coordinates for sequences with two peaks to sheet 2 of result Excel file.

5.   **Known Limitations**

- **Simplistic Peak and Valley Detection:**

  Plateaus and close data points are handled using predefined rules, which may not capture all nuances.

- **Input Format Dependency:**

  Input files must strictly follow the specified format for successful processing.

- **Interpolation:**

  Assumes linear interpolation when calculating imperfection lengths.