Requirement Specification: Interference Injection Tool

# 1. Introduction

The Interference Injection Tool is a signal processing application designed to inject artificially generated interference into clean signal data and analyze the resulting impact. It provides users with precise control over the interference parameters, enabling the visualization, manipulation and injection of interference into a clean signal. The tool is ideal for researchers and engineers working in telecommunications, wireless systems, and other domains that require analysis of how interference affects signal integrity.

# 2. Functional Requirements

## 2.1 User Interface

The tool will have a graphical user interface (GUI) with two primary tabs:  
1. Add Interference Tab: This tab allows users to import clean signals and interference data, visualize both, adjust interference parameters and inject the interference into the clean signal.  
2. Analyse Export Tab: This tab lets users load and analyze the interfered signal, visualize it, and compute key statistics such as mean, maximum and minimum values for each frequency.

## 2.2 Add Interference Tab

This tab contains the primary functionality for adding interference to a clean signal. It includes options for importing signals, visualizing them, adjusting interference parameters and previewing the effects of interference injection.

1. Import Clean Signal (CSV):
   1. The clean signal file is uploaded in CSV format. Each row of the CSV represents a PSD (Power Spectral Density) value at a specific point in time.
   2. The CSV file contains the following information:
      1. Start Frequency: The lowest frequency in the signal.
      2. Stop Frequency: The highest frequency in the signal.
      3. PSD Values: Power levels in decibel-milliwatts (dBm) for each frequency within the defined range.
2. Visualization of clean signal:
   1. Upon importing, the tool generates an equidistant frequency range between the start and stop frequencies, corresponding to each PSD value.
   2. The signal is visualized in a graph where the X-axis represents frequency and the Y-axis represents PSD values in dBm.
   3. Users can navigate through the time dimension of the signal using "Previous" and "Next" buttons to visualize how the signal evolves over time.
3. Import Interference Data (CSV file):
   1. The interference data is loaded from a .csv file, which contains power values in dBm that represent interference.
   2. Since interference data files can be large, the user can specify how many values to load (via a vector length parameter) and an offset (starting point in the file).
4. Visualization of Interference data:
   1. The interference signal is visualized on a separate graph, aligned with the clean signal.
   2. The interference is represented as power values across the same frequency range as the clean signal, allowing for an intuitive comparison between the two.
5. Interference Parameter Adjustments:
   1. Users can modify the interference data through the following parameters:
      1. Interferer Spectrum Width: Adjusts the frequency bandwidth of the interference. This is calculated as the difference between the start and stop frequencies.
      2. Interferer Offset: Allows users to shift the power values of the interference, effectively controlling the amplitude of the interference injected into the clean signal.
      3. Interferer Center Frequency: Sets the central frequency point of the interference. Adjusting this parameter shifts the interference along the frequency axis.
      4. Interferer Vector Length: Determines how many values to load and visualize from the interference data. If the interference data contains 10,000 points and the vector length is set to 1,000, the user can browse through 10 pages of interference, with each page containing 1,000 points.
   2. Visualization:
      1. Each adjustment made to the interference parameters is immediately reflected on the graph, allowing users to fine-tune interference before applying it to the clean signal.
6. Interference Injection:
   1. Once the interference parameters are set, users can inject the interference onto the clean signal using one of two strategies:
      1. Round Robin Strategy: Sequentially injects each interference "page" into the corresponding clean signal row. If the number of clean signal rows exceeds the number of interference pages, the process loops, reapplying interference from the beginning.
      2. One-to-All Strategy: Applies the currently visualized interference page to all clean signal rows.
7. Signal Processing:
   1. To inject interference, the following operations occur:
      1. Interpolation: The interference signal is interpolated to match the frequency points of the clean signal.
      2. Power Conversion: Both the clean signal and interference are converted from decibel-milliwatts (dBm) to milliwatts using the formula: A black text on a white background

         Description automatically generated
      3. Signal Combination: The milliwatt values of the clean signal and interference are summed for each frequency point.
      4. Back to dBm: The combined values are converted back to dBm using: PdBm=10×log10(PmW)
   2. A "Preview" button allows users to see how the interference will affect the clean signal before committing to the change. (visualization)
8. Export Functionality:
   1. Once interference is injected, the user can export the new, interfered signal as a CSV file.
   2. The exported CSV maintains the structure of the original clean signal file (start frequency, stop frequency, PSD values) with the addition of interference.
   3. Users are prompted to provide a file name for the export.

## 2.3 Analyze Export Tab

This tab focuses on analyzing the interfered signal that has been exported from the "Add Interference" tab.

1. Import and Visualize Exported Signal:
   1. Users can import the interfered signal CSV file and visualize it similarly to the clean signal visualization in the first tab.
   2. As with the clean signal, the user can navigate through the interfered signal in time (row by row), displaying the PSD values across the frequency spectrum.
2. Statistical Analysis:
   1. For each frequency, the tool calculates and displays the mean, maximum, and minimum values across the rows (representing time).
   2. This allows users to easily analyze the effects of interference on the clean signal.
3. Export Analysis:
   1. Users can export the calculated statistics (mean, max, min for each frequency) as a separate CSV file for further analysis or documentation purposes.

# 3. Non-Functional Requirements

## 3.1 Performance

The tool should handle large datasets efficiently without significant delays in visualization or processing.

## 3.2 Usability

The interface should be intuitive and provide real-time feedback to user actions (e.g., parameter adjustments).

## 3.3 Scalability

The tool should be capable of managing varying signal sizes and data types without performance degradation.

# 4. Technical Specifications

The system should support CSV and DAT file formats for input/output operations.  
The final tool should be compatible with modern operating systems (e.g., Windows, Linux).