GSOC 2020: Filter Design Tool Enhancements

Siddharth Kapoor

March 29, 2020

1 Introduction

Digital Signal Processing has been used in solving real world problems from quite some time, and the use of filters has long grown. From 1st order to low pass filters to really complicated filter banks, evolution of filters and the related tools has given us immense power to solve new and exciting problems. GNU Radio, being an open source signal processing toolkit comprises of strong signal processing concepts, as block diagrams which makes it for the user to grasp and build up on concepts. This proposal addresses the idea of "Filter Design Tool Enhancements", in particular extending the capabilities for the digital filters in GNU Radio platform.

2 Topic Background

Digital Signal Processing has been using the techniques of filter banks and polyphase decomposition for designing various filters such as interpolators and decimators. These filters have have had numerous applications in the real world and continue to do so. Filter Design has been imlplemented in various languages as tool boxes, or signal processing blocks due to its popular use. Polyphase decomposition enabled desigining of filters with dramatic computational efficiency, in multirate signal processing.
GNU Radio in filter designing uses pfb.py for Polyphase filter banks, filterbank.py to generate synthesis and analysis filter taps. The CSV file is made in main_window.py in the function 'action_save_dialog'. The optfir.py designs optimal FIR filters using pm_remez.cc file using Parks-McClellan algorithm for FIR filter design.

3 The Deliverable

Deliverable 0: Updating PyQt4 to PyQt5 all throughout filters and their GUIs.
Deliverable 1: CSV file saver and reader function to save the file, or use it in filter blocks.
Deliverable 2: Function for All Pass Polyphase Filter Bank and its support
Deliverable 3: Function for Two-Path All Pass Polyphase Filter Bank and it's support.

4 License

GNU Radio is licensed under the GNU General Public License v3.0, and I intend to use the same license to distribute my code.

5 Schedule

- '**May 4, 2020 - June 1, 2020** - Community Bonding
  - Introduction to the mentor and the community
  - Introduction to code
- **June 1, 2020 - June 6, 2020** - Updating PyQt4 to PyQT5
  - The gr-qtgui code needs to be updated from PyQt4 to PyQt5.
  - The ticket number is #649 and #651.
  - This includes the examples and gui files.
  - Also updating gr-filter gui to PyQt5.
- **June 8, 2020 - June 13, 2020**
  - Improving the functionality of the function action_save_dialog
  - Integrating the filter design tool into GRC
  - Writing a function to use the CSV saved files to be directly read to be used in actual blocks

  - Input for the function - saved CSV file, with a specific format, Output - The parameters for the block to be added
  - Adding documentation on updating to PyQt5

- **June 14, 2020 - June 29, 2020**
  - Writing test cases for the above function
  - Writing documentation for the function
  - Adding support for bessel type in the IIR filter
  - Start preparing for Phase 1 evaluations

- **June 29, 2020 - July 8, 2020**
  - Start with Phase 1 evaluations
  - Finishing up implementing the function and updating to PyQt5
  - Buffer week for any lag in the work.

- **July 9, 2020 - July 16, 2020**
  - Adding support for Cascade Integrator Comb Filters specifically multiply free filter.
  - Write a function for multiply free filter
  - For reference - Chapter 11, Page 326, Harris F.J., Multirate Signal Processing for Communication Systems(2004)
  - The function will return taps like any other filter block

- **July 17, 2020 - July 24, 2020**
  - Adding documentation for cascaded filter
  - Adding example cases in examples here.
  - Adding test cases for cascaded filters
  - Buffer week for cascaded filter and it's related documentation and testing
  - Preparing for Phase 2 evaluations

- **July 25, 2020 - July 31, 2020**

  - Phase 2 Evaluations
  - Improving support for polyphase filter banks.
  - Adding a function for All Pass Recursive Polyphase Filters
  - References -
    * `https://github.com/trondeau/gnuradio/blob/filter/design_tool_newgui/gr-filter/python/filter/pfb.py`
    * Harris F.J., Multirate Signal Processing for Communication Systems(2004)

- **August 3, 2020 - August 8, 2020**
  - Add test cases for All Pass Recursive PFB
  - Add documentation for the filter
  - Add examples for the filter
  - Perform Testing

- **August 10, 2020 - August 15, 2020**
  - Adding support for Two-Path All Pass Recursive PFB (This is done to add support for M-path All Pass Filter in future)
  - Add Documentaion for the Two-Path All Pass PFB
  - Add Test Cases and example codes for the same.

- **August 17, 2020 - August 24, 2020**
  - Completing the support for PFB filters
  - Completing the support for Cascaded filters
  - Buffer week for adding documentation and debugging
  - Preparing for final evaluations

- **August 24, 2020 - August 31, 2020**
  - Submission for final evaluations
  - Completion of any leftovers
  - Final Presentation/Report Submission

– Completing documentation and community approval.

Apart from this a weekly report will be submitted to the mentor, in order to inform him of the progress and to discuss any suggestions and code blocks. A personal blog ,open to the community, will also be maintained to make the community about the progress on day-to-day basis.

6 Proof of your coding capabilities / prerequisite capabilities

1. These are some of the courses that I have taken relevant to the project - Linear Signal and Systems (LSS), Digital Signal Processing, Advanced Digital Signal Processing, Digital System Design using Verilog, Image Processing and Computer Vision, Information Processing and Compression, Statistical Analysis, Data Structure and Algorithms, Matrix Theory and Stochastic Processes, Embedded Systems, Digital Electronics and Computer Architecture.

2. Languages I have worked on -

   - MATLAB, C/C++, Python - great proficiency
   - Verilog,Assembly Language, LaTeX, Arduino IDE - mediocre proficency
   - Proficient with using Linux and Unix, along with boards such as RPi, Arduino and MSP430.

7 Formalities

1. I am Siddharth Kapoor, currently residing in Delhi, India.

2. I am a final year undergraduate, pursuing Electronics and Communication Engineering at National Institute of Technology Karnataka located in Surathkal, Karnataka, India.

3. No, the GSoC work will not be in any way affiliated to my university and hence I won't be recieving any credit from university's side.

8 Candidate Background

1. I have proficiency in C/C++ from our CS101 courses in first year and since then we have been taught DSA and I learnt Dynamic Programming for various online competitions. I am also proficient in Python which has led me to do more projects in signal processing space particularly image processing and Deep Learning.

2. These are some of the projects I did -

   (a) **Customer Profiling and Language Detection Using Real-Time Call Center Speech Data** -
   We aim to predict age, gender and language of a customer given an audio recording between a call center agent and the customer. We do this by extracting MFCC features, VAD and then training a SVR and DNN using these features. We feed First Order Statistics calculated from a GMM-UBM to the SVR/DNN Model. (Project is ongoing)
   Report till now - Customer Profiling and Language Detection using Real-Time Call-Center Data
   Tools Used - Python, in particular Keras, HTK Toolkit, Kaldi

   (b) **Chest X-Ray image analysis using Deep Learning** -
   Trained models using dataset of 100,000 Chest X-Ray images to identify 14 different diseases (labels) in a given Chest X-Ray image. Tools Used - Python libraries - Pytorch, Pyignite and tensorflow for training models, sklearn for pre-processing and visualizing data.
   Published Paper - Jointly Learning Convolutional Representations to Compress Radiological Images and Classify Thoracic Diseases in the Compressed Domain
   GitHub repo Link - `https://github.com/ekagra-ranjan/AE-CNN`

   Apart from these I did a lot of small projects on my own to understand and apply different python libraries such as PyGame, PyTurtle etc. Although these projects weren't very helpful for me academically but they helped me expand my scope my knowledge for coding.

3. I haven't contributed as much to open source but this is where GSoC comes in,although I have participated in Hacktober Fest from last 3

years.

4. I have high proficiency in English and Hindi, and I am available on Skype, Google Hangouts, Google Meet and Zoom.

5. After GSoC I intend to contribute actively and get involved more in the open-source community and propose projects of my own. Signal Processing has always been my field of interest and research, and adding the current functionalities to an open source application would be really helpful for me and the community.

9 Acknowledge the three strikes rule

I, Siddharth Kapoor, acknowledge the three strikes rule for the application of GSoC 2020 under the organization GNU Radio.

10 The secret code word

"Cyberspectrum is the best spectrum"

11 References

1. Lyons, R. P., Understanding Digital Signal Processing, Section 10.4 (Polyphase Filters) and Section 13.20 (A Practical Spectrum Analyzer)

2. Proakis, J. G., and Manolakis, D. G., Digital Signal Processing, Section 10.5.2 (Polyphase Filter Structures)