Medical Physics Programming Projects

Overview

Centres all around the world are constantly reinventing the wheel. It's all well and good wanting to collaborate, however there is a discoverability problem. We don't know what other teams are working on and what they plan to work on. We also don't necessarily know the problems that people have previously solved with code and would be willing to share.

This document is designed for each team of Medical Physicists from around the world to present a range of tools that they have already built, tools that they are working on now, and tools that they wish to work on in the future. By collating this document should another group share common interests with you, reach out to them, and consider building the desired functionality together into the Python open source Medical Physics repository PyMedPhys.

Please only share projects where you would be willing to have the source code be made open source. Have each new programming group create a new page and organise the programming groups alphabetically. Under each clinical Medical Physics team organise your programming projects under the following four categories:

Developed and in use

This is code that is currently being used within your clinic. It is certainly allowed to have rough edges, but nevertheless it should be at the bare minimum be currently serving a purpose within your clinic.

Working but not in regular use

Code that is at least partially working, and would be useful to other clinics to use as a starting point to pick up the project, but it has not been rolled out into standard use within your centre.

Scratch

At some point this was worked on, and there is some resemblance of working code. It might be useful to someone.

Ideas

Big picture of projects that would be keen to work on, but no working code has been produced as of yet.

Table of Contents

Cancer Care Associates (RCCC and NBCC)

Developed and in use

- Exporting profiles from DICOM and Mephysto files for comparing beam models in bulk to our reference measurement files
    - https://nbviewer.jupyter.org/github/pymedphys/pymedphys/blob/33e890a3efa0460b5d7ffac5ab72bab83bf4de22/examples/labs/howto-drafts/tps_compare.ipynb
- Cross site SSH networking server to programmatically access Mosaiq and Linac backups across sites
    - A centralised Physics server allowing various tools (such as DICOM transfer from the Linac) to be accessible from any of our physical locations over a private SSH network
- Elekta Linac TRF decoding
    - https://docs.pymedphys.com/user/interfaces/cli/trf.html
- Logfile orchestration, indexing, and management
    - https://docs.pymedphys.com/user/interfaces/cli/logfile.html
- Extracting data from the Mosaiq database
    - Comparing fields between Mosaiq instances
    - Downloading QCLs for analysis
    - https://github.com/pymedphys/pymedphys/blob/33e890a3efa0460b5d7ffac5ab72bab83bf4de22/pymedphys/_mosaiq/helpers.py
- A custom winston lutz tool
    - In use, but needs serious refinement and improvement
    - https://nbviewer.jupyter.org/github/pymedphys/pymedphys/blob/33e890a3efa0460b5d7ffac5ab72bab83bf4de22/examples/labs/rccc/wlutz/mviso-analysis.ipynb
- Monaco archiving and GUI for RTs to de-archive patients
    - https://github.com/pymedphys/pymedphys/tree/33e890a3efa0460b5d7ffac5ab72bab83bf4de22/examples/labs/rccc/monaco_archive_and_dearchive
- Electron insert factor modelling
    - https://www.physicamedica.com/article/S1120-1797(15)01018-2/fulltext
    - https://docs.pymedphys.com/user/library/electronfactors.html
    - https://nbviewer.jupyter.org/github/pymedphys/pymedphys/blob/33e890a3efa0460b5d7ffac5ab72bab83bf4de22/examples/labs/rccc/electron_insert_factor/Electron%20insert%20user%20interface.ipynb
- Create arbitrary Elekta linac service mode QA fields using pydicom and dicomtk, create and send to service mode within Python
    - Usable, but still very clunky
    -
    - https://nbviewer.jupyter.org/github/pymedphys/pymedphys/blob/33e890a3efa0460b5d7ffac5ab72bab83bf4de22/examples/labs/rccc/create_service_mode_plan/20191003_1026.ipynb

- Extending DICOM CT superiorly or inferiorlyhttps://github.com/pymedphys/pymedphys/blob/33e890a3efa0460b5d7ffac5ab72bab83bf4de22/pymedphys/_dicom/ct/extend.py

Working but not in regular use

- DICOM anonymisation
  - https://docs.pymedphys.com/user/library/dicom.html#anonymisation
- Gamma index
  - https://docs.pymedphys.com/user/library/gamma.html
  - https://docs.pymedphys.com/user/examples/gamma_from_dicom.html
- MU density determination
  - https://docs.pymedphys.com/user/library/mudensity.html
  - https://simonbiggs.net/epsm2018
- Brachytherapy dose calculation from DICOM RT plan file
  - https://github.com/SimonBiggs/teap-brachytherapy-portfolio/raw/master/Independent%20dose%20verification%20of%20Brachytherapy%20plans%20using%20DICOM%20data%20exports.docx
  - https://nbviewer.jupyter.org/github/SimonBiggs/teap-brachytherapy-portfolio/blob/master/TG43%20Implementation.ipynb
  - https://github.com/SimonBiggs/brachytherapy-dicom-utilities
- Tooling using GDCM to help RTs decompress their own DICOM files before importing into the TPS
- Access to DoseCHECK databases
- Film tooling
  - Extracting arbitrary reference frames from the TPS by contouring a rotated box to pull dose out of the TPS in a spatially accurate well defined way -- Used with the film cube within the Max Phantom.
  - https://github.com/pymedphys/pymedphys/blob/33e890a3efa0460b5d7ffac5ab72bab83bf4de22/pymedphys/labs/cube.py#L329
- Scale printing of shapes for phantoms
- Building programmatic CT phantoms and structures with Python

Scratch

- Reading the Elekta iCOM stream
- Automated tests to verify internal links within word documents on the shared drive haven't been broken by file shuffles
  - Designed to be run automatically each night and email physics

Ideas

- A robust way to determine whether or not an MLC modulated plan needs to undergo measurement. Currently we measure all IMRT and VMAT plans. We have purchased Sun Nuclear DoseCHECK and are looking to have some classes of plans not

undergo measurement instead be checked via DoseCHECK combined with a Monaco/Mosaiq/Log file verification.
- Automated first pass of structure set QA
- Have great PyMedPhys documentation so that even a non-python user can get started with PyMedPhys and begin building tools for their centre
- An open source easy to install, easy to use, dose calculation tool
    - Configurable with beam model and CT2ED
    - Generic models for machines currently available provided and iterated on by the community
    - Use should be to provide CT DICOM files, DICOM structure files, as well as DICOM plan files via command line and it outputs a DICOM dose file.
    - Installation should be as simple as either `pip install pymedphys` or downloading an executable from a website.
- Create a server which is easy to install and set up via Docker which bundles Orthanc, QATrack+, pylinac, pymedphys, as well as Jupyter Hub for providing Python to all users within your centre via the web browser JupyterLab within JupyerHub.
- Create an Elekta ball bearing mount based QA procedure field and process. Design it such that should a morning warmup quasar phantom be used instead of the BB phantom this field could also be delivered after the kV alignment check on some mornings. The field could be a combination of two arcs with opposing collimator angles followed by varied field sizes, and varied MLC shapes sequenced in an optimum manner.
    - Something could trigger (maybe iCOM) that this test has been undergone. If it is delivered via Mosaiq, then a Mosaiq database listener could be used to trigger the extraction, analysis, reporting, and adding to the trend.
    - Alternatively, at the conclusion of the field the iView images could be sent to an Orthanc server, but this precludes using iView movie mode and hence limits the ability to use arcs (a potential multiplicative speed loss).
    - Challenges
        - iView movies can't be sent by DICOM
        - iView images are stored in an old lossless JPEG format (not insurmountable)
        - To automatically access iView images an approach could be to create a SAMBA share on the NSS, then map that share to drive Z:/ say, and for these QA tests, choose a database that has been created on the Z:/ drive on the iView machine
- Have an automated TPS OIS comparison tool that listens to Mosaiq and confirms that the OIS parameters agree with what was approved within the TPS as well as what was finally delivered on the Linac. Be robust to the needs to split MU in the OIS if MU > 1000 and the like. This robustness can be achieved by leaning on comparing MU Densities instead of comparing exact control point definitions. Test to see if the MU Density per gantry angles section, as well as collimator angles, and turn table positions all agree within rounding errors bounds for the OIS TPS comparison and within reasonable uncertainties for the TPS Logfile comparison.
    - Challenges

- Need a way to automatically export plans from Mosaiq and detect which one was approved

JARMC

Developed and in use

- Verify plan monitor units from an exported Mosaiq RTP file based on hand-calculation using an XML data-book. If MapCheck fluence maps are present, instead verify the composite isocenter dose from the CAX diode measurement. Present the result as an excel file including a setup summary for the paper-chart.
- Calculate couch lat, vert, long positions for entry in Mosaiq based on Pinnacle plan parameters.

Working but not in regular use

- Conversion of beam profiles from Pinnacle ASCII to RFA format, readable by RayStation.
- Renaming of exported e-mails into a form convenient for archiving and searching.

Scratch

- Characterization and editing of beam profiles.
- Extraction of beam profiles from Profiler and MapCheck into RFA format.
- Using a Tomotherapy treatment plan sinogram exported from RayStation, unshuffle the image into a set of more-visualizable “fluence maps”.
- Using exported Mosaiq Treatment History Report (including MLC component), create a scatter diagram that illustrated the couch lat, long, vert values used over a course of treatment.
- Using an office copier to scan a gafchromic film (PNG file) and a corresponding exported dose distribution, find the dose response curve for the film/copier pair.
- Utility for checking good order in networked storage space used to store various patient dosimetric records and do some automatic tidying.
- Process Mosaiq’s “chart check” PDF report to produce a list of active patients.

Ideas

- Automated processing of occupational exposure reports into a consolidated report.
- Convert the hand calc program to accept DICOM and not depend on Excel.

Mays Cancer Center (Univ of Texas Health Science Center)

Working but not in regular use

- Data transfer check from TPS to Record and Verify
    a. It will compare DICOM information from the TPS versus the data that is transferred to the R&V
    b. It will check for default parameters (for example: dose grid resolution, default CT to density table, correct dose calculation algorithm, etc)
    c. The third check will be to identify differences on weekly/daily basis.
- Translate existing MatLab codes to python
    a. Several codes have been written in the past in MatLab. One of the efforts we need to focus is to translate them into python going forward.

Ideas

1. Initial plan quality checks
    a. Check plan quality in terms of DVH objectives, hot spots, cold spots
    b. Calculate radiobiological parameters (Probability of injury, Probability of benefit and P+)
    c. Calculate Plan score (we have created a scoring method to basically convert the DVH in to 1 number) and track the score per site.
2. Build CT phantoms for deformable imaging QA
    a. Create CTs and then use a known deformation to warp them.
    b. Export them to the 3rd party software and test their deformation. Test performance for
        i. various noise levels
        ii. Artifacts
        iii. Incomplete sets
3. Create a test suite for automated TG53 and MPPG5a
    a. Create CTs, RT Structures, Beams, and calculated dose
    b. Use the TPS exported plans to quickly create a report for annual TG53 or MPPG5a
4. Automated monthly QA
    a. Using mostly EPID, XVI, CAT phantom, logfiles and a 2D array acquire data for periodic (monthly) linac QA
        i. EPID will be used for most mechanical tests (light field coincidence, couch position accuracy, gantry and collimator accuracy, ODI)
        ii. Imaging QA using EPID and XVI
        iii. Dosimetric QA with 2D array
        iv. MLC performance using log files and/or EPID

b. Make measurements
c. Create a report
d. Trend results

Tom Baker Cancer Centre (Univ of Calgary)

Working but not in regular use

- Automated monthly QA (in testing within the centre)
    a. Asymmetric Jaws automated QA. Tool for calculating jaws junction shifts for linear accelerators. The script opens every DICOM file in a given folder and creates a combined profile resulting from the superposition of the two or more fields. It then detects the peak/through formed by the gap/overlap of the fields. A window is then selected around this point and a Savitzky-Golay smoothing filter is then applied to the combined profile. This new curve is then used iteratively to minimize the profile created every time one of the profiles slide to close the gap or decrease the overlap. The profile will achieve it greatest level of homogeneity when the dosimetric penumbra of both fields are matched in space. The final result is the optimal calculation of the gap/overlap between the two profiles.
    https://github.com/peterg1t/QCJawsLinac
    b. Light/Rad automated QA. Tool for calculating light/radiation coincidence for linear accelerators. The script opens a DICOM file and automatically detects the innermost bibs in the phantom using the Laplacian of Gaussian algorithm, for a 15 cm x 15 cm field. Tests have been done with a MED-TEC MT-IAD-1 phantom (I would like to extend this to other phantoms). The final result is the distance between the difference of the center of the bib and radiation field edge minus the distance between the visible bib and the phantom edge.
    https://github.com/peterg1t/QCLightRad
    c. MV spatial distortion and MV imaging/treatment center coincidence automated QA. This QC determines the spatial distortion and treatment center of a Linac using the graticule issued to every Varian machine.
    https://github.com/peterg1t/QCGraticule
    d. Oncentra 3D data visualizer. Integrated tool for visualization of Oncentra structure, plan and dose files in mayavi. It opens brachytherapy DICOM files exported by Oncentra Prostate and displays a 3D view of structure, plan and/or dose
    e. Oncentra data explorer. Oncentra Prostate exports a bulk of DICOM files in the format RP/RS/RD-SOP Instance UID. This script scans a folder full of DICOM files and print the comments of each DICOM file which are more helpful in the identification of the files than the UID numbers.

Scratch

1. Development of a Radial distribution function to assess the spatial distribution of sources in a brachytherapy treatment.