TROTS
The Radiotherapy Optimisation Test Set

Detailed Data Description

Sebastiaan Breedveld & Ben Heijmen
Erasmus University Medical Center, Rotterdam, The Netherlands

1 Introduction

The Radiotherapy Optimisation Test Set (TROTS) is an extensive set of problems originating from radiation therapy treatment planning. This document merely provides a detailed description of the data structure. Origin, acquisition, and description can be found in the original data article [1], and applications in [2,3]. The dataset can be found at [4].

1. Breedveld S and Heijmen B *TROTS - The Radiotherapy Optimisation Test Set*, under review
2. Breedveld S, Van den Berg B and Heijmen B *An interior-point implementation developed and tuned for radiation therapy treatment planning*, under review
3. Breedveld S, Craft D, Van Haveren R and Heijmen B *Multi-criteria Optimisation and Decision-Making in Radiotherapy*, under review
4. Dataset location: `http://www.erasmusmc.nl/radiotherapytrots`

2 Data format

The data is stored in *Matlab* files, version 7.3. The files are fully HDF5 compliant, and can therefore be read using general HDF5 tools.

Each file contains 3 structures: **problem** defining the mathematical optimisation problem, **data** containing the numerical data matrices, and **patient** containing the CT scan and other information required for visualisation of the data. In addition, a **solutionX** is included containing the result of the optimisation problem.

The contents of the structures are described in the next sections.

3 The *problem* structure

The **problem** structure is a list, where each entry defines an objective or constraint. Each entry has the following fields:

- **dataID** Reference index to the **data** structure, containing the respective numerical data.
- **Name** A name that refers to the clinical structure this constraint/objective is based on. Is ignored by the solver.
- **Minimise** If *True* for an *objective*, this objective will be *minimised*. If *True* for a *constraint* this is a *maximum* constraint. If set *False*, vice versa.
- **Type** Identifier for the used *cost-function* (see section 5.1).
- **Parameters** Sets parameters to configure the *cost-function*, given in **Type** (see section 5.1).
- **Objective** For a *constraint*, this is the value the cost-function is constrained to. For an *objective* in a multi-criteria setting, this is the aspired value.
- **Sufficient** For an *objective* in a multi-criteria problem, the objective value does not need to become lower (higher) than the given sufficient value. Is ignored when set empty.
- **Weight** Scalar to apply to the *objective*, useful to scalarise and weigh multiple objectives.
- **Priority** Natural number that indicates the priority of this *objective*. Used in multi-criteria optimisation.
- **Active** Can be *True* or *False* to enable or disable this objective/constraint.
- **IsConstraint** If *True*, this entry is a *constraint*, and an *objective* otherwise.
- **Chain** Extra information for *chain* function type (see section 5.1).

4 The *data* structure

The data structure contains 2 substructures: *matrix*, containing the numerical data, and *misc*, containing auxiliary data to configure the problem. The *matrix* structure has the following fields:

- **Name** A name that refers to the clinical volume (e.g. organ name or artificial structure name), or other background of this data.
- **A** The data *matrix*. Each matrix in the **data** structure has an equal number of columns, equal to the number of decision variables. In radiotherapy, this matrix is generally the *pencil-beam* dose matrix. The number of rows typically indicate the number of voxels (sampled elements in the CT where the dose is evaluated), and the number of columns equals the number of pencil-beam weights.
- **b** Offset *vector*, is 0 unless you are doing something exciting such as generating a treatment plan on top of an already delivered dose.

- **c** A *scalar* for quadratic cost-functions, empty otherwise.

- **Type** Indicating the matrix type. When *Type=0*, this is a "normal" matrix operating in the *fluence-to-dose* domain, where the argument d for the cost-functions is computed as $d = Ax + b$. When *Type=1*, this matrix operates in the *fluence* domain only (i.e. linear smoothing of the fluence map), but is treated equally as *Type=0* by mathematical solvers. When *Type=2*, this is a *quadratic* or *square* matrix, meaning that when the problem is extended with auxiliary decision variables (e.g. to solve mini-max problems when minimising a pointwise maximum), the padding should be done both in rows and columns.

The *misc* structure has the following fields:

- **size** Indicates the number of decision variables of the problem, equal to the number of columns for *all* A matrices.

- **real** Original problem size. The set $\{x_i : i \in 1, \ldots, real\}$ of the decision variables identifies the part representing the actual radiotherapy problem, i.e. the problem before the introduction of auxiliary decision variables (e.g. mini-max problems). This is required to be able to retrieve the original radiotherapy problem.

- **InitialiseMatrixID** List of indices to the *matrix* substructure required for initialising the problem. In general, these matrices represent the tumour matrices.

- **InitialiseReferenceDose** Right hand side for the least-squares initialisation.

- **InitialiseRegularisationMatrixID** Reference to a *matrix* substructure that regulates the least-squares problem.

5 The *patient* structure

The patient structure is not required for optimisation, but useful in visualisation and interpretation of the results. We briefly describe the structure, exact details can be found in the scripts. Units are in *mm*, 1-based matrix index and degrees.

- **Identifier** String, identifies a particular patient (e.g.. *Prostate 07*).

- **PatientPosition** Can be HFS (head first supine), FFS (feet first supine), HFP (head first prone), or FFP (feet first prone), and defines in which position the patient is imaged. For HFS and HFP, the images are viewed from the feet position, so "right is left". Is not used by the viewer, but required for the user to interpret the view. See `https://public.kitware.com/IGSTKWIKI/index.php/DICOM_data_orientation`.

- **CT** 3D matrix in int16 format, contains the CT data. The dimensions are ordered x, y and z, and follow the LPS system when the **PatientPosition** is HFS (`https://public.kitware.com/IGSTKWIKI/index.php/DICOM_data_orientation`). The minimum value of -1024 is defined as "outside patient".

- **Resolution** Resolution in mm of a CT element.
- **Isocentre** Focal point for the treatment device, generally in the centre of the tumour.
- **Offset** Spatial position in mm, identifying the position of the $(1, 1, 1)$ corner of the CT.
- **DoseBox** Describes for each dimension the minimum and maximum matrix elements where dose is expected (dosimetric region of interest).
- **StructureNames** List of names for all delineated structures.
- **SampledVoxels** For each structure a $3 \times N$ matrix describing the positions in the CT matrix where the dose is evaluated. This position corresponds one-on-one with the matrices in the **data** structure.
- **Contours** Structure that contains the delineations of the structures.
- **Beams** Contains information of the treatment setup.

5.1 The *cost-functions*

The cost-functions are referenced by type index. Except for the quadratic type (2), we use $d(x) = Ax + b$ as argument for all functions, and m is the number of voxels/length of d.

1. **linear** type, computes the pointwise minimum $f(x) = \min(d(x))$, pointwise maximum $f(x) = \max(d(x))$, or mean $f(x) = \frac{1}{m}\sum_{i=1}^{m} d_i(x)$. The mean is always precomputed into a single-row matrix for numerical efficiency.
2. **quadratic** type, computes $f(x) = \frac{1}{2}x^T Ax + b^T x + c$.
3. **gEUD** type, the *generalised mean*, or in radiotherapy better known as the *generalised equivalent uniform dose*, computed as $f(x) = \left(\frac{1}{m}\sum_{i=1}^{m} d_i(x)^a\right)^{\frac{1}{a}}$, where a is given as *Parameters*. For $a \geq 1$, this function is convex and can be minimised, for $a \leq -1$, this function is concave and needs to be maximised.
4. **LTCP** type, the *logarithmic tumour control probability*, exclusively used for the tumour. $f(x) = \frac{1}{m}\sum_{i=1}^{m} e^{-\alpha(d_i(x)-d^p)}$ where d^p is the prescribed dose, given as the first element of *Parameters*, and α the cell-sensitivity, given as the second element of *Parameters*.
5. **DVH** type, the *dose-volume histogram* or *partial-volume* cost-function. In its exact form, $f(x) = \frac{1}{m}\sum_{i=1}^{m} I_{d_i(x)<d^c}(d_i(x))$ where d^c is the critical dose level (given as the first element of *Parameter*) and I the indicator function. This simply results in the fraction of d larger than d^c. The solver described in [2] uses the smoothed approximation $f(x) = \frac{1}{m}\sum_{i=1}^{m} \frac{\left(\frac{d_i(x)}{d^c}\right)^p}{1+\left(\frac{d_i(x)}{d^c}\right)^p}$ during optimisation, where the parameter p is the steepness, given as the second element of *Parameters*.

6. **Chain** type, describing chain functions. The format can be given 2 or 3 elements, (a, i) or (a, i, j). The first form describes functions of the form $a \cdot g_i(x)$, where a is a scalar and i is an index to an other objective given in the *problem* structure. When *Chain* has multiple rows, the different functions are accumulated, resulting in functions of the type $a_1 g_{i_1}(x) + a_2 g_{i_2}(x) + \ldots$. The 3-element form describes functions of the form $a \cdot g_i(x) \leq x_j$, a type that simplifies mini-max constraints.