Chalmers University of Technology
University of Gothenburg
Mathematical Sciences
Optimization
Caroline Olsson
Christoffer Cromvik

MVE165
MMG630
Applied Optimization
Assignment information

5th May 2009

# Assignment 3b: Radiation Therapy

Given below is the description of the project in Radiation Therapy. In short, the project consists of forming an optimization model (a nonlinear program), implementing the model, and solving it. It is important that you understand the accompanying lecture on Radiation Therapy, and that you have completed the computer exercise on nonlinear optimization.

This document and supplementary files for the implementation can be found on the course homepage:
`http://www.math.chalmers.se/Math/GrundUtb/CTH/mve165/0809/`

To pass the assignment you should (in groups of two persons) do exercises 1 and 2 and write a detailed report on the answers and explanations. Students aiming for a higher grade should also do exercises 3 and 4 and include the received results in the report. The report should be e-mailed to `anstr@chalmers.se`

**at the latest on Friday 15 of May 2009.**

Your shall also `(ii)` present your project orally at a seminar

**on Monday 18 of May 2009.**

We also ask you to estimate the number of hours you spent on this assignment and note this in your report or presentation.

## Problem background

You will design and solve an optimization problem for an IMRT treatment of a tumor situated in the head-and-neck region. The objective is to eradicate the tumor, while sparing the organs-at-risk. An oncologist has defined the planned target volume (PTV), which consist of the tumor and a marginal around it. The oncologist has also defined the organs at risk (OAR) to be the left and right parotid glands, the spinal cord and the brain stem.

There are mainly two approaches to model the problem: either with physical criteria or with biological objectives. You will use both in this assignment. Let $d$ denote a vector with the discretized dose (in voxels). An example of a physical criterion is quadratic deviation from an upper dose limit $u$ for an organ with voxels $J$:

$$f = \sum_{j \in J} \max\{d_j - u, 0\}^2$$

The biological criteria that we will use are the generalized Equivalent Uniform

Dose (gEUD)

$$gEUD = \left( \sum_{j \in J} d_j^{1/n} \right)^n ,$$

where the parameter $n$ controls the volume effect, and the Normal Tissue Control Probability (NTCP)

$$\text{NTCP} = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{u} \exp^{-t^2/2} dt,$$
$$u = \frac{\text{gEUD(d)} - D_{50}}{mD_{50}},$$

where $D_{50}$ is the homogeneous dose corresponding to 50% risk of complication and $m$ determines the slope of the risk.

The PTV comprises 16274 voxels; the left parotid gland 1791 voxels; the right parotid gland 1903 voxels; the spinal cord 1888 voxels and the brain stem 1075 voxels. The treatment energy has been determined to 6 MeV, the number of beams to five and the beam entry angles to 0, 72, 144, 216, and 288. The prescription dose for the PTV is 68 Gy which means that doses between 64.6 Gy (-5%) and 71.4 Gy (+5%) are desirable.

## Exercises to perform

The optimization of a treatment plan is a multi-objective problem. Choose one (or more) of the methods:

- Weighted sum of objectives
- $\varepsilon$-constrained method
- Preemptive optimization
- Soft constraints

to solve the problems below. Only use two objectives by either combining some of the functions or put them as constraints. You choose, but motivate! Remember to scale all functions appropriately.

> Since the problems are rather large, it is essential that you at least provide analytic gradients to all functions. Providing analytic Hessians, where available, is even better!

1. Optimize with quadratic deviations for the target, the spinal cord, and the brain stem. See Table 1 for parameter values.
2. Optimize with quadratic deviation for the target. Use gEUD for the spinal cord, the brain stem, and the parotid glands. Compute NTCP for the organs-at-risk for your optimal solutions. See Table 2 for parameter values.

| Organ | Upper dose limit (Gy) | Dose-volume (DV) |
|---|---|---|
| Spinal cord | 45 | – |
| Brain stem | 50 | – |
| Parotid gland | – | 30% of volume less than 25 Gy |

Table 1: Parameters for physical criteria.

| Organ | $n$ | $m$ | $D_{50}$ | $gEUD_0$ |
|---|---|---|---|---|
| Spinal cord | 0.05 | 0.175 | 66.5 | 45 |
| Brain stem | 0.16 | 0.14 | 65 | 50 |
| Parotid gland | 0.7 | 0.18 | 46 | 26 |

Table 2: Parameters for biological criteria. The parameter $gEUD_0$ is the (preferred) upper bound for gEUD. It corresponds to approximately 5% risk of complication. Excess dose to the spinal cord and the brain stem may lead to paralysis. Excess dose to the parotid glands may lead to permanent loss of salivary production.

3. Optimize with quadratic deviation for the target and use NTCP for the organs-at-risk. Compare your results to those of Exercise 2).
4. Optimize with quadratic deviations for the target, the spinal cord, and the brain stem. Use DV-functions for the parotid glands.

## Practical details

Given the beamlet intensities $x$, you can compute the doses $d$ for the voxels in a structure by using an influence matrix $K$, using the linear relations

$$d = Kx.$$

The data for a head–and–neck test is found in `imrtdata_hn.mat`, which contains the influence matrix `K`, and the two structures `target` and `oar`. These structures contain the field `ind` which is the indices of $d$ which comprise the target/organ. They also contain `mask` which is a binary three-dimensional matrix which indicates the location. It can be used to make three-dimensional plots in Matlab. If `D=d(target.ind),dose=zeros(256,256,73);`, then you can set the doses for visualization with `dose(target.mask)=D`. The structure `oar` also contains the name of the organ.

To create a Dose-Volume Histograms (DVH) from your computed (optimal) beamlet intensities $x$, use

```
>> pltdvh(x, K, target, oar)
```

Use the newest installed version of matlab : `matlab2008b`

```
> /chalmers/sw/sup/matlab-2008b/bin/matlab
```

Experiment with `options` for `fmincon`. Particularly with `Algorithm`, and `Hessian`.

You may need to supply data to your objective function (`objfunc.m`) besides the variable vector `x`. You can either use global variables, or (better) write your `objfunc.m` with the header

```
function [f, g, H] = objfunc(x, data1, data2)
```

and when you use `fmincon`, use

```
x = fmincon(@(x) objfunc(x, data1, data2), ...
```

The symbol `@(x)` transforms the function into a function handle which only depends on `x`.

Computational speed is essential for this application, so implement well. Vectorize your code as much as possible. *Avoid* reallocating sparse matrices like

```
d = ones(lengh(ind),1);
g = K(ind,:)'*d
```

*Instead*, use

```
d = zeros(size(K,1),1);
d(ind) = ones(length(ind));
g = K'*d;
```

which creates an equivalent result.

To supply analytic gradients, you need some differentiability skill. Here is a hint: Assume $y = Ax$ and $f(y) = \hat{f}(Ax)$. If you you want to express the gradient of $\hat{f}$ with respect to $x$, use the equivalence $\nabla_x \hat{f} = A^{\mathrm{T}} \nabla_y f$.