Example: Robust Treatment Planning with Protons

Table of Contents

In this example we will (i) create a small artifical phantom (ii) create a scanned proton treatment plan considering a constant RBE of 1.1 (iii) we will enable dose calculation on nine selected worst case scenarios (iv) robustly optimize the pencil beam intensities on all 9 dose scenarios using the composite worst case paradigm (v) visualise all individual dose scenarios (vi) sample discrete scenarios from Gaussian uncertainty assumptions

Patient Data

Let's begin with a clear Matlab environment and import the liver patient into your workspace.

```
clc,clear,close all;
```

Create a CT image series

```
xDim = 150;
yDim = 150;
zDim = 50;

ct.cubeDim      = [xDim yDim zDim];
```

```
ct.resolution.x = 2;
ct.resolution.y = 2;
ct.resolution.z = 3;
ct.numOfCtScen  = 1;

% create an ct image series with zeros - it will be filled later
ct.cubeHU{1} = ones(ct.cubeDim) * -1000;
```

Create the VOI data for the phantom

Now we define structures a contour for the phantom and a target

```
ixOAR = 1;
ixPTV = 2;

% define general VOI properties
cst{ixOAR,1} = 0;
cst{ixOAR,2} = 'contour';
cst{ixOAR,3} = 'OAR';
cst{ixPTV,1} = 1;
cst{ixPTV,2} = 'target';
cst{ixPTV,3} = 'TARGET';

% define optimization parameter for both VOIs
cst{ixOAR,5}.TissueClass = 1;
cst{ixOAR,5}.alphaX      = 0.1000;
cst{ixOAR,5}.betaX       = 0.0500;
cst{ixOAR,5}.Priority    = 2;
cst{ixOAR,5}.Visible     = 1;
cst{ixOAR,6}.type        = 'square overdosing';
cst{ixOAR,6}.dose        = 30;
cst{ixOAR,6}.penalty     = 10;
cst{ixOAR,6}.EUD         = NaN;
cst{ixOAR,6}.volume      = NaN;
cst{ixOAR,6}.coverage    = NaN;
cst{ixOAR,6}.robustness  = 'none';

cst{ixPTV,5}.TissueClass = 1;
cst{ixPTV,5}.alphaX      = 0.1000;
cst{ixPTV,5}.betaX       = 0.0500;
cst{ixPTV,5}.Priority    = 1;
cst{ixPTV,5}.Visible     = 1;
cst{ixPTV,6}.type        = 'square deviation';
cst{ixPTV,6}.dose        = 60;
cst{ixPTV,6}.penalty     = 50;
cst{ixPTV,6}.EUD         = NaN;
cst{ixPTV,6}.volume      = NaN;
cst{ixPTV,6}.coverage    = NaN;
cst{ixPTV,6}.robustness  = 'none';
```

Lets create a cubic phantom

first the OAR

```
cubeHelper = zeros(ct.cubeDim);
xLowOAR    = round(xDim/2 - xDim/6);
xHighOAR   = round(xDim/2 + xDim/6);
yLowOAR    = round(yDim/2 - yDim/6);
yHighOAR   = round(yDim/2 + yDim/6);
zLowOAR    = round(zDim/2 - zDim/6);
zHighOAR   = round(zDim/2 + zDim/6);

for x = xLowOAR:1:xHighOAR
   for y = yLowOAR:1:yHighOAR
      for z = zLowOAR:1:zHighOAR
         cubeHelper(x,y,z) = 1;
      end
   end
end

% extract the voxel indices and save it in the cst
cst{ixOAR,4}{1} = find(cubeHelper);

% second the PTV
cubeHelper = zeros(ct.cubeDim);
radiusPTV = xDim/12;
for x = 1:xDim
   for y = 1:yDim
      for z = 1:zDim
         currPost = [x y z] - round([ct.cubeDim./2]);
         if  sqrt(sum(currPost.^2)) < radiusPTV
            cubeHelper(x,y,z) = 1;
         end
      end
   end
end

% extract the voxel indices and save it in the cst
cst{ixPTV,4}{1} = find(cubeHelper);

%a ssign relative electron densities
vIxOAR = cst{ixOAR,4}{1};
vIxPTV = cst{ixPTV,4}{1};

ct.cubeHU{1}(vIxOAR) = 1;  % assign HU of water
ct.cubeHU{1}(vIxPTV) = 1;  % assign HU of water
```

Treatment Plan

The next step is to define your treatment plan labeled as 'pln'. This structure requires input from the treatment planner and defines the most important cornerstones of your treatment plan.

First of all, we need to define what kind of radiation modality we would like to use. Possible values are photons, protons or carbon. In this example we would like to use protons for robust treatment planning. Next, we need to define a treatment machine to correctly load the corresponding base data. matRad features generic base data in the file 'carbon_Generic.mat'; consequently the machine has to be set accordingly

```
pln.radiationMode = 'protons';
```

```
pln.machine       = 'Generic';
```

Define the biological optimization model for treatment planning along with the quantity that should be used for optimization. Possible model values are: 'none': physical optimization; 'constRBE': constant RBE of 1.1; 'MCN': McNamara-variable RBE model for protons; 'WED': Wedenberg-variable RBE model for protons 'LEM': Local Effect Model and possible quantityOpt are 'physicalDose', 'effect' or 'RBExD'. As we use protons, we use a constant RBE of 1.1.

```
modelName    = 'constRBE';
quantityOpt  = 'RBExD';
```

The remaining plan parameters are set like in the previous example files

```
pln.numOfFractions        = 20;
pln.propStf.gantryAngles  = [0 90];
pln.propStf.couchAngles   = [0 0];
pln.propStf.bixelWidth    = 5;
pln.propStf.numOfBeams    = numel(pln.propStf.gantryAngles);
pln.propStf.isoCenter     = ones(pln.propStf.numOfBeams,1) *
 matRad_getIsoCenter(cst,ct,0);
pln.propOpt.runDAO        = 0;
pln.propOpt.runSequencing = 0;

% retrieve bio model parameters
pln.bioParam =
 matRad_bioModel(pln.radiationMode,quantityOpt,modelName);

% retrieve 9 worst case scenarios for dose calculation and
 optimziation
pln.multScen = matRad_multScen(ct,'wcScen');
```

Generate Beam Geometry STF

```
stf = matRad_generateStf(ct,cst,pln);
```

matRad: Generating stf struct... Progress: 100.00 %

Dose Calculation

```
dij = matRad_calcParticleDose(ct,stf,pln,cst);
```

```
matRad: Particle dose calculation...
shift scenario 1 of 7:
matRad: Particle dose calculation...
Warning: ray does not hit patient. Trying to fix afterwards...Beam 1
 of 2:
matRad: calculate radiological depth cube...done.
matRad: calculate lateral cutoff...done.
Progress: 100.00 %
Beam 2 of 2:
matRad: calculate radiological depth cube...done.
matRad: calculate lateral cutoff...done.
Progress: 100.00 %
shift scenario 2 of 7:
```

```
matRad: Particle dose calculation...
Warning: ray does not hit patient. Trying to fix afterwards...Beam 1
 of 2:
matRad: calculate radiological depth cube...done.
matRad: calculate lateral cutoff...done.
Progress: 100.00 %
Beam 2 of 2:
matRad: calculate radiological depth cube...done.
matRad: calculate lateral cutoff...done.
Progress: 100.00 %
shift scenario 3 of 7:
matRad: Particle dose calculation...
Warning: ray does not hit patient. Trying to fix afterwards...Beam 1
 of 2:
matRad: calculate radiological depth cube...done.
matRad: calculate lateral cutoff...done.
Progress: 100.00 %
Beam 2 of 2:
matRad: calculate radiological depth cube...done.
matRad: calculate lateral cutoff...done.
Progress: 100.00 %
shift scenario 4 of 7:
matRad: Particle dose calculation...
Warning: ray does not hit patient. Trying to fix afterwards...Beam 1
 of 2:
matRad: calculate radiological depth cube...done.
matRad: calculate lateral cutoff...done.
Progress: 100.00 %
Beam 2 of 2:
matRad: calculate radiological depth cube...done.
matRad: calculate lateral cutoff...done.
Progress: 100.00 %
shift scenario 5 of 7:
matRad: Particle dose calculation...
Warning: ray does not hit patient. Trying to fix afterwards...Beam 1
 of 2:
matRad: calculate radiological depth cube...done.
matRad: calculate lateral cutoff...done.
Progress: 100.00 %
Beam 2 of 2:
matRad: calculate radiological depth cube...done.
matRad: calculate lateral cutoff...done.
Progress: 100.00 %
shift scenario 6 of 7:
matRad: Particle dose calculation...
Warning: ray does not hit patient. Trying to fix afterwards...Beam 1
 of 2:
matRad: calculate radiological depth cube...done.
matRad: calculate lateral cutoff...done.
Progress: 100.00 %
Beam 2 of 2:
matRad: calculate radiological depth cube...done.
matRad: calculate lateral cutoff...done.
Progress: 100.00 %
```

```
shift scenario 7 of 7:
matRad: Particle dose calculation...
Warning: ray does not hit patient. Trying to fix afterwards...Beam 1
 of 2:
matRad: calculate radiological depth cube...done.
matRad: calculate lateral cutoff...done.
Progress: 100.00 %
Beam 2 of 2:
matRad: calculate radiological depth cube...done.
matRad: calculate lateral cutoff...done.
Progress: 100.00 %
```

Inverse Optimization for IMPT based on RBE-weighted dose

The goal of the fluence optimization is to find a set of bixel/spot weights which yield the best possible dose distribution according to the clinical objectives and constraints underlying the radiation treatment.

```
resultGUI = matRad_fluenceOptimization(dij,cst,pln);

% Trigger robust optimization and make the objective to a composite
 worst case objective
cst{ixPTV,6}.robustness  = 'COWC';
cst{ixOAR,6}.robustness  = 'COWC';

resultGUIrobust = matRad_fluenceOptimization(dij,cst,pln);
```

```
Calculating probabilistic quantities for optimization ...

******************************************************************************
This program contains Ipopt, a library for large-scale nonlinear
 optimization.
 Ipopt is released as open source code under the Eclipse Public
 License (EPL).
         For more information visit http://projects.coin-or.org/Ipopt
******************************************************************************

This is Ipopt version 3.11.8, running with linear solver ma57.

Number of nonzeros in equality constraint Jacobian...:        0
Number of nonzeros in inequality constraint Jacobian.:        0
Number of nonzeros in Lagrangian Hessian.............:        0

Total number of variables............................:     4172
                     variables with only lower bounds:     4172
                variables with lower and upper bounds:        0
                     variables with only upper bounds:        0
Total number of equality constraints.................:        0
Total number of inequality constraints...............:        0
        inequality constraints with only lower bounds:        0
   inequality constraints with lower and upper bounds:        0
        inequality constraints with only upper bounds:        0
```

```
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du
 alpha_pr  ls
   0 2.5143408e+001 0.00e+000 1.01e+000   0.0 0.00e+000    -  0.00e
+000 0.00e+000   0
   1 2.5138871e+001 0.00e+000 1.58e-002  -1.0 1.28e-002    -
 9.97e-001 1.00e+000f  1
   2 1.3293761e+001 0.00e+000 4.42e-003  -6.9 2.81e+000    -
 8.16e-001 1.00e+000f  1
   3 6.8735167e+000 0.00e+000 1.51e-003  -2.4 5.18e+000    -
 9.97e-001 1.00e+000f  1
   4 5.1914931e+000 0.00e+000 1.25e-003  -3.4 2.30e+000    -
 9.67e-001 1.00e+000f  1
   5 4.0296984e+000 0.00e+000 1.17e-003  -4.3 3.13e+000    -
 9.56e-001 1.00e+000f  1
   6 3.0258597e+000 0.00e+000 1.03e-003  -4.5 5.02e+000    -
 9.73e-001 1.00e+000f  1
   7 2.6524650e+000 0.00e+000 7.66e-004  -4.9 7.90e+000    -
 9.97e-001 2.79e-001f  1
   8 2.5417928e+000 0.00e+000 2.43e-003  -5.3 8.46e+000    -
 9.99e-001 8.46e-002f  1
   9 2.2585509e+000 0.00e+000 1.69e-003  -5.8 7.76e+000    -
 9.57e-001 2.68e-001f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du
 alpha_pr  ls
  10 2.1161822e+000 0.00e+000 1.79e-003  -6.3 6.15e+000    -  1.00e
+000 2.05e-001f  1
  11 1.9886354e+000 0.00e+000 7.42e-004  -6.7 7.04e+000    -  1.00e
+000 2.00e-001f  1
  12 1.9370953e+000 0.00e+000 3.00e-003  -7.5 4.63e+000    -  1.00e
+000 1.38e-001f  1
  13 1.8232200e+000 0.00e+000 1.03e-003  -7.7 7.09e+000    -  1.00e
+000 2.15e-001f  1
  14 1.7697359e+000 0.00e+000 1.72e-003  -6.9 9.68e+000    -  1.00e
+000 8.45e-002f  1
  15 1.6982116e+000 0.00e+000 1.42e-003  -5.3 1.02e+001    -
 9.39e-001 1.08e-001f  1
  16 1.6376416e+000 0.00e+000 8.59e-004  -4.8 1.21e+001    -
 6.69e-001 8.36e-002f  1
  17 1.6086117e+000 0.00e+000 1.72e-003 -10.9 7.25e+000    -
 3.73e-001 6.91e-002f  1
  18 1.5187020e+000 0.00e+000 6.71e-004  -4.7 1.24e+001    -
 8.06e-001 1.60e-001f  1
  19 1.4959634e+000 0.00e+000 2.72e-003  -4.2 4.89e+000    -
 7.44e-001 9.06e-002f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du
 alpha_pr  ls
  20 1.4161288e+000 0.00e+000 1.01e-003  -4.6 7.77e+000    -
 6.36e-001 2.06e-001f  1
  21 1.3802882e+000 0.00e+000 9.83e-004  -5.3 5.66e+000    -
 5.00e-001 1.35e-001f  1
  22 1.3464287e+000 0.00e+000 8.89e-004  -5.0 4.74e+000    -
 8.27e-001 1.71e-001f  1
  23 1.2947026e+000 0.00e+000 1.14e-003  -4.9 4.62e+000    -
 9.83e-001 2.93e-001f  1
```

```
  24 1.2598017e+000 0.00e+000 6.13e-004  -4.8 4.55e+000    -
 7.80e-001 2.43e-001f  1
  25 1.2307871e+000 0.00e+000 4.90e-004 -10.8 4.77e+000    -
 4.79e-001 2.12e-001f  1
  26 1.2091681e+000 0.00e+000 1.50e-003  -5.7 4.02e+000    -  1.00e
+000 2.12e-001f  1
  27 1.1711456e+000 0.00e+000 4.40e-004  -5.7 5.86e+000    -
 7.01e-001 2.76e-001f  1
  28 1.1615294e+000 0.00e+000 1.08e-003  -6.1 6.32e+000    -
 7.92e-001 6.51e-002f  1
  29 1.1373896e+000 0.00e+000 6.48e-004  -4.8 4.93e+000    -
 4.34e-001 2.24e-001f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du
 alpha_pr  ls
  30 1.1103275e+000 0.00e+000 2.77e-004  -5.1 8.20e+000    -
 6.73e-001 1.76e-001f  1
  31 1.1091785e+000 0.00e+000 8.26e-004 -11.0 2.26e+000    -
 3.62e-001 2.52e-002f  1
  32 1.0757153e+000 0.00e+000 3.62e-004  -5.3 7.72e+000    -
 5.57e-001 2.42e-001f  1
  33 1.0674972e+000 0.00e+000 2.99e-004  -7.3 5.04e+000    -
 3.47e-001 8.30e-002f  1
  34 1.0522781e+000 0.00e+000 3.86e-004  -5.9 6.00e+000    -
 6.19e-001 1.42e-001f  1
  35 1.0381292e+000 0.00e+000 2.04e-004  -4.8 5.38e+000    -
 4.04e-001 1.69e-001f  1
  36 1.0336405e+000 0.00e+000 3.79e-004  -5.3 3.81e+000    -
 3.27e-001 7.07e-002f  1
  37 1.0203831e+000 0.00e+000 4.26e-004  -5.1 6.24e+000    -
 7.22e-001 1.38e-001f  1
  38 1.4517631e+000 0.00e+000 1.58e-003  -3.1 2.54e+001    -
 5.66e-002 1.64e-001f  1
  39 1.2918728e+000 0.00e+000 1.26e-003  -4.2 6.74e+000    -
 8.18e-001 2.38e-001f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du
 alpha_pr  ls
  40 1.1519901e+000 0.00e+000 8.33e-004  -4.2 2.79e+000    -
 4.07e-001 3.72e-001f  1
  41 1.0182493e+000 0.00e+000 1.42e-003  -4.2 2.30e+000    -
 7.92e-001 9.88e-001f  1
  42 1.0050827e+000 0.00e+000 1.18e-003  -4.8 1.17e+000    -
 9.83e-001 6.78e-001f  1
  43 1.0014463e+000 0.00e+000 8.48e-004  -5.8 1.51e+000    -
 9.87e-001 1.53e-001f  1
  44 9.8971082e-001 0.00e+000 3.50e-004  -6.1 3.22e+000    -
 9.96e-001 2.56e-001f  1
  45 9.7658573e-001 0.00e+000 3.51e-004  -5.8 2.96e+000    -
 8.70e-001 3.38e-001f  1
  46 9.7160991e-001 0.00e+000 5.06e-004  -5.9 2.29e+000    -
 8.01e-001 1.85e-001f  1
  47 9.6213601e-001 0.00e+000 1.96e-004  -5.6 4.35e+000    -
 7.25e-001 2.32e-001f  1
  48 1.0061886e+000 0.00e+000 1.53e-004  -3.5 3.38e+001    -
 2.01e-002 1.10e-001f  1
```

```
  49 9.8598861e-001 0.00e+000 2.10e-004  -5.0 1.20e+001    -
 4.79e-001 1.59e-001f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du
 alpha_pr  ls
  50 9.5840042e-001 0.00e+000 3.03e-004  -5.0 6.28e+000    -
 4.73e-001 4.04e-001f  1
  51 9.5389976e-001 0.00e+000 4.99e-004  -5.8 2.95e+000    -
 9.54e-001 1.36e-001f  1
  52 9.4766191e-001 0.00e+000 5.28e-004 -11.0 2.85e+000    -
 4.65e-001 1.99e-001f  1
  53 9.4049656e-001 0.00e+000 4.57e-004  -7.2 3.59e+000    -
 1.39e-001 2.04e-001f  1
  54 9.3820439e-001 0.00e+000 6.38e-004  -6.8 3.53e+000    -
 8.08e-001 7.00e-002f  1
  55 9.2857075e-001 0.00e+000 5.13e-004  -7.3 4.27e+000    -
 3.65e-001 2.62e-001f  1
  56 9.2656435e-001 0.00e+000 4.13e-004  -7.8 3.29e+000    -
 8.84e-001 7.53e-002f  1
  57 9.2179277e-001 0.00e+000 3.41e-004  -5.0 2.00e+000    -
 5.44e-001 3.36e-001f  1
  58 9.1813391e-001 0.00e+000 3.33e-004  -6.0 4.05e+000    -
 3.93e-001 1.30e-001f  1
  59 9.1484105e-001 0.00e+000 6.25e-004  -5.8 5.07e+000    -
 7.23e-001 9.64e-002f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du
 alpha_pr  ls
  60 9.0990422e-001 0.00e+000 1.32e-003  -6.4 3.07e+000    -
 7.08e-001 2.40e-001f  1
  61 9.0404991e-001 0.00e+000 1.18e-003  -6.5 7.14e+000    -
 1.13e-001 1.36e-001f  1
  62 9.0127907e-001 0.00e+000 7.47e-004  -5.2 6.90e+000    -
 2.94e-001 7.10e-002f  1
  63 8.9734650e-001 0.00e+000 6.20e-004  -5.1 5.99e+000    -
 1.90e-001 1.27e-001f  1
  64 8.9549998e-001 0.00e+000 4.08e-004  -7.3 3.98e+000    -
 2.86e-001 8.49e-002f  1
  65 8.9268778e-001 0.00e+000 2.74e-004  -5.5 6.35e+000    -
 5.48e-001 8.42e-002f  1
  66 8.9122229e-001 0.00e+000 5.57e-004  -5.7 1.80e+000    -
 6.17e-001 1.47e-001f  1
  67 8.8743941e-001 0.00e+000 5.23e-004  -6.8 3.99e+000    -
 2.50e-001 1.77e-001f  1
  68 8.8173352e-001 0.00e+000 3.04e-004  -5.7 9.07e+000    -
 3.78e-001 1.33e-001f  1
  69 9.7492008e-001 0.00e+000 4.07e-004  -3.7 1.86e+001    -
 3.92e-002 2.89e-001f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du
 alpha_pr  ls
  70 9.3871104e-001 0.00e+000 3.36e-004  -4.7 1.19e+001    -
 5.25e-001 2.64e-001f  1
  71 9.2153831e-001 0.00e+000 6.96e-004  -4.7 6.61e+000    -
 5.49e-001 1.96e-001f  1
  72 9.0226009e-001 0.00e+000 2.96e-004  -4.7 5.23e+000    -
 4.66e-001 3.18e-001f  1
```

```
  73 8.9581887e-001 0.00e+000 5.97e-004  -4.7 1.68e+000    -
 4.34e-001 2.41e-001f  1
  74 8.8799318e-001 0.00e+000 4.80e-004  -4.7 1.94e+000    -  1.00e
+000 3.71e-001f  1
  75 8.8228050e-001 0.00e+000 4.51e-004  -4.7 1.29e+000    -
 7.30e-001 5.86e-001f  1
  76 8.7893233e-001 0.00e+000 4.85e-004  -4.7 8.51e-001    -  1.00e
+000 7.93e-001f  1
  77 8.7634330e-001 0.00e+000 5.91e-004  -5.4 1.46e+000    -
 9.24e-001 3.81e-001f  1
  78 8.7372255e-001 0.00e+000 2.93e-004  -6.1 3.63e+000    -
 8.28e-001 1.50e-001f  1
  79 8.7050988e-001 0.00e+000 9.70e-004  -6.6 3.35e+000    -
 8.84e-001 1.90e-001f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du
 alpha_pr  ls
  80 9.8404644e-001 0.00e+000 6.39e-004  -3.9 2.83e+001    -
 7.33e-002 4.75e-001f  1
  81 9.1370307e-001 0.00e+000 5.05e-004  -4.8 1.33e+001    -
 7.70e-001 4.25e-001f  1
  82 8.7130362e-001 0.00e+000 4.55e-004  -4.8 6.52e+000    -
 5.19e-001 7.37e-001f  1
  83 8.7086569e-001 0.00e+000 1.53e-003  -4.8 2.80e+000    -
 4.80e-001 1.00e+000f  1
  84 8.6978017e-001 0.00e+000 3.03e-004  -4.8 1.61e+000    -  1.00e
+000 1.23e-001f  1
  85 8.6492256e-001 0.00e+000 4.39e-004  -5.4 2.67e+000    -
 9.96e-001 3.27e-001f  1
  86 8.5744046e-001 0.00e+000 6.80e-004  -6.1 3.40e+000    -
 8.51e-001 4.48e-001f  1
  87 8.5640310e-001 0.00e+000 5.98e-004  -7.1 2.65e+000    -
 8.46e-001 7.80e-002f  1
  88 8.5437586e-001 0.00e+000 5.83e-004  -7.1 2.21e+000    -
 2.01e-002 1.72e-001f  1
  89 8.5320038e-001 0.00e+000 3.43e-004  -7.6 2.93e+000    -
 3.48e-001 7.58e-002f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du
 alpha_pr  ls
  90 8.4885647e-001 0.00e+000 3.90e-004  -5.2 3.10e+000    -
 5.15e-001 2.89e-001f  1
  91 8.4603017e-001 0.00e+000 2.02e-004  -5.4 3.51e+000    -
 3.74e-001 1.75e-001f  1
  92 8.4507457e-001 0.00e+000 2.85e-004  -6.7 2.26e+000    -
 3.27e-001 9.51e-002f  1
  93 8.4159038e-001 0.00e+000 2.59e-004  -5.9 5.61e+000    -
 6.34e-001 1.48e-001f  1
  94 8.4037430e-001 0.00e+000 3.56e-004  -5.8 3.12e+000    -
 5.94e-001 9.38e-002f  1
  95 8.3875319e-001 0.00e+000 5.54e-004  -7.5 2.84e+000    -
 5.35e-001 1.36e-001f  1
  96 8.3476779e-001 0.00e+000 3.80e-004  -8.0 5.10e+000    -
 1.82e-001 1.90e-001f  1
  97 8.3389073e-001 0.00e+000 4.18e-004  -6.3 3.79e+000    -
 4.86e-001 5.78e-002f  1
```

```
  98 8.2994728e-001 0.00e+000 3.36e-004  -6.0 7.26e+000    -
 1.53e-001 1.46e-001f  1
  99 8.4315028e-001 0.00e+000 3.16e-004  -4.0 2.93e+001    -
 1.07e-002 9.00e-002f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du
 alpha_pr  ls
 100 8.3775119e-001 0.00e+000 2.82e-004  -5.6 9.60e+000    -
 7.20e-002 1.28e-001f  1
 101 8.3366980e-001 0.00e+000 4.41e-004  -5.6 1.11e+001    -
 7.67e-001 8.96e-002f  1
 102 8.3144879e-001 0.00e+000 4.79e-004  -5.6 3.74e+000    -
 5.03e-001 1.35e-001f  1
 103 8.3060285e-001 0.00e+000 3.19e-004  -5.6 1.99e+000    -
 4.99e-001 9.45e-002f  1
 104 8.2708608e-001 0.00e+000 5.16e-004  -5.6 2.57e+000    -
 4.35e-001 3.12e-001f  1
 105 9.0326423e-001 0.00e+000 8.06e-004  -3.5 9.86e+001    -
 1.37e-002 1.70e-001f  1
 106 8.6164288e-001 0.00e+000 1.52e-003  -4.8 1.10e+001    -
 8.64e-001 4.57e-001f  1
 107 8.3881693e-001 0.00e+000 8.49e-004  -4.8 3.56e+000    -
 6.93e-001 1.00e+000f  1
 108 8.3524452e-001 0.00e+000 6.82e-004  -4.8 6.61e-001    -
 2.51e-001 4.49e-001f  1
 109 8.3401738e-001 0.00e+000 4.40e-004  -4.8 7.54e-001    -
 6.02e-001 1.00e+000f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du
 alpha_pr  ls
 110 8.3225462e-001 0.00e+000 2.59e-004  -4.8 5.58e-001    -
 9.83e-001 1.00e+000f  1
 111 8.3073510e-001 0.00e+000 3.12e-004  -4.8 1.85e+000    -
 9.13e-001 9.20e-001f  1
 112 8.2919747e-001 0.00e+000 2.68e-004  -4.8 2.53e+000    -
 7.52e-001 8.48e-001f  1
 113 8.2782953e-001 0.00e+000 2.27e-004  -4.8 2.92e+000    -
 6.92e-001 8.67e-001f  1
 114 8.2726078e-001 0.00e+000 2.34e-004  -4.8 1.85e+000    -
 4.54e-001 4.36e-001f  1
 115 8.2626849e-001 0.00e+000 2.75e-004  -4.8 2.17e+000    -
 5.61e-001 1.00e+000f  1
 116 8.2416433e-001 0.00e+000 1.89e-004  -5.7 2.32e+000    -
 7.91e-001 3.62e-001f  1
 117 8.2226643e-001 0.00e+000 2.21e-004  -6.1 4.62e+000    -
 9.99e-001 1.61e-001f  1
 118 8.1861743e-001 0.00e+000 5.47e-004  -5.3 3.13e+000    -
 5.65e-001 3.57e-001f  1
 119 8.4373575e-001 0.00e+000 5.20e-004  -3.5 1.14e+002    -
 1.93e-003 6.75e-002f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du
 alpha_pr  ls
 120 8.1653415e-001 0.00e+000 5.15e-004  -5.3 1.61e+001    -
 2.88e-002 5.14e-001f  1
 121 8.1405498e-001 0.00e+000 2.04e-004  -5.5 6.19e+000    -
 9.83e-001 1.17e-001f  1
```

```
 122 8.0975357e-001 0.00e+000 2.42e-004  -5.8 3.23e+000    -
 6.79e-001 5.14e-001f  1
 123 8.0818888e-001 0.00e+000 1.89e-004 -11.0 2.94e+000    -
 7.97e-001 2.54e-001f  1
 124 8.0593906e-001 0.00e+000 4.61e-004  -7.0 4.02e+000    -
 9.02e-001 2.99e-001f  1
 125 8.0405687e-001 0.00e+000 2.68e-004  -6.7 3.91e+000    -
 8.68e-001 2.82e-001f  1
 126 8.2370134e-001 0.00e+000 9.88e-004  -4.4 6.15e+001    -
 4.61e-002 2.50e-001f  1
 127 8.1465043e-001 0.00e+000 6.14e-004  -5.2 9.02e+000    -
 4.83e-001 4.31e-001f  1
 128 8.1152699e-001 0.00e+000 4.51e-004  -5.2 5.67e+000    -
 6.05e-001 2.46e-001f  1
 129 8.0852882e-001 0.00e+000 1.55e-004  -5.2 4.13e+000    -
 6.43e-001 3.51e-001f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du
 alpha_pr  ls
 130 8.0522921e-001 0.00e+000 8.73e-005  -5.2 2.68e+000    -
 6.64e-001 6.62e-001f  1
 131 8.0402637e-001 0.00e+000 1.61e-004  -5.2 8.54e-001    -
 7.46e-001 1.00e+000f  1
 132 8.0220337e-001 0.00e+000 1.42e-004  -6.0 4.49e+000    -
 8.03e-001 3.29e-001f  1
 133 8.0090277e-001 0.00e+000 1.27e-004 -11.0 7.58e+000    -
 5.71e-001 1.89e-001f  1
 134 8.7563442e-001 0.00e+000 2.87e-004  -3.8 2.24e+002    -
 1.44e-002 2.08e-001f  1
 135 8.1119711e-001 0.00e+000 1.79e-003  -5.1 5.91e+001    -  1.00e
+000 6.81e-001f  1
 136 8.0321074e-001 0.00e+000 5.85e-005  -5.1 7.16e+000    -
 9.01e-001 1.00e+000f  1
 137 8.0060919e-001 0.00e+000 1.45e-004  -7.6 1.05e+001    -
 8.00e-001 2.70e-001f  1
 138 7.9904915e-001 0.00e+000 9.31e-005  -5.7 6.86e+000    -
 9.27e-001 2.30e-001f  1
 139 7.9695747e-001 0.00e+000 7.84e-004  -6.4 6.71e+000    -
 9.92e-001 3.37e-001f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du
 alpha_pr  ls
 140 7.9510022e-001 0.00e+000 4.11e-004  -6.2 9.26e-001    -
 7.41e-001 1.00e+000f  1
 141 7.9465045e-001 0.00e+000 3.24e-004  -6.6 2.55e+000    -
 9.95e-001 1.00e-001f  1
 142 7.9380669e-001 0.00e+000 1.53e-004  -6.9 3.22e+000    -
 9.25e-001 1.65e-001f  1
 143 7.9188029e-001 0.00e+000 4.48e-004  -7.0 4.14e+000    -
 6.74e-001 3.24e-001f  1
 144 7.9760830e-001 0.00e+000 8.60e-004  -5.2 1.18e+001    -
 6.56e-002 5.82e-001f  1
 145 7.9573094e-001 0.00e+000 4.99e-004  -5.6 6.86e+000    -
 4.57e-001 1.87e-001f  1
 146 7.9274504e-001 0.00e+000 3.11e-004  -5.6 6.90e+000    -
 4.00e-001 3.21e-001f  1
```

```
 147 7.9222087e-001 0.00e+000 3.49e-004  -5.6 2.41e+000    -
 5.21e-001 1.60e-001f  1
 148 7.9076968e-001 0.00e+000 2.78e-004  -5.6 2.80e+000    -
 5.53e-001 4.13e-001f  1
 149 7.9053587e-001 0.00e+000 2.51e-004  -6.2 2.73e+000    -
 6.57e-001 5.97e-002f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du
 alpha_pr  ls
 150 7.8885813e-001 0.00e+000 5.76e-004  -6.3 2.20e+000    -
 6.65e-001 7.70e-001f  1
 151 7.8867581e-001 0.00e+000 3.73e-004  -7.4 4.64e+000    -
 7.95e-001 4.18e-002f  1
 152 8.4999620e-001 0.00e+000 4.57e-004  -4.5 5.20e+001    -
 1.24e-002 5.69e-001f  1
 153 7.9772783e-001 0.00e+000 3.42e-003  -5.3 4.09e+001    -
 8.06e-001 6.01e-001f  1
 154 7.9158988e-001 0.00e+000 2.41e-004  -5.3 5.15e+000    -
 8.83e-001 7.05e-001f  1
 155 7.8958491e-001 0.00e+000 4.56e-004  -5.3 1.98e+000    -
 8.17e-001 1.00e+000f  1
 156 7.8914603e-001 0.00e+000 1.76e-004  -5.3 1.29e+000    -  1.00e
+000 8.36e-001f  1
 157 7.8893168e-001 0.00e+000 3.53e-004  -5.3 1.97e+000    -
 8.13e-001 4.06e-001f  1
 158 7.8852980e-001 0.00e+000 2.51e-004  -5.3 3.64e+000    -
 7.55e-001 2.59e-001f  1

Number of Iterations....: 158

                                   (scaled)                 (unscaled)
Objective...............:  7.8852979970877035e-001
 7.8852979970877035e-001
Dual infeasibility......:  2.5115776509959323e-004
 2.5115776509959323e-004
Constraint violation....:  0.0000000000000000e+000
 0.0000000000000000e+000
Complementarity.........:  8.5367597548351157e-006
 8.5367597548351157e-006
Overall NLP error.......:  2.5115776509959323e-004
 2.5115776509959323e-004


Number of objective function evaluations             = 159
Number of objective gradient evaluations             = 159
Number of equality constraint evaluations            = 0
Number of inequality constraint evaluations          = 0
Number of equality constraint Jacobian evaluations   = 0
Number of inequality constraint Jacobian evaluations = 0
Number of Lagrangian Hessian evaluations             = 0
Total CPU secs in IPOPT (w/o function evaluations)   =      4.994
Total CPU secs in NLP function evaluations           =      7.011

EXIT: Solved To Acceptable Level.
Calculating final cubes...
```

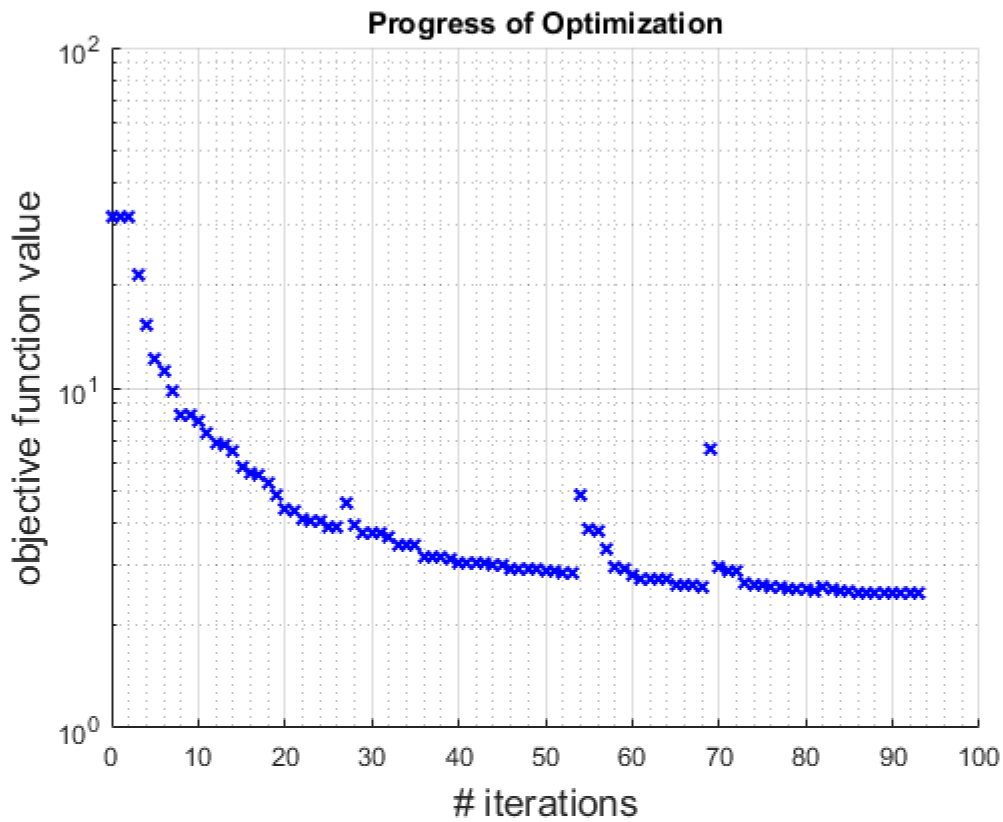
```
Calculating probabilistic quantities for optimization ...

******************************************************************************
This program contains Ipopt, a library for large-scale nonlinear
 optimization.
 Ipopt is released as open source code under the Eclipse Public
 License (EPL).
         For more information visit http://projects.coin-or.org/Ipopt
******************************************************************************

This is Ipopt version 3.11.8, running with linear solver ma57.

Number of nonzeros in equality constraint Jacobian...:        0
Number of nonzeros in inequality constraint Jacobian.:        0
Number of nonzeros in Lagrangian Hessian.............:        0

Total number of variables............................:     4172
                     variables with only lower bounds:     4172
                variables with lower and upper bounds:        0
                     variables with only upper bounds:        0
Total number of equality constraints.................:        0
Total number of inequality constraints...............:        0
        inequality constraints with only lower bounds:        0
   inequality constraints with lower and upper bounds:        0
        inequality constraints with only upper bounds:        0

iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du
 alpha_pr  ls
   0 3.1813028e+001 0.00e+000 1.01e+000   0.0 0.00e+000    -  0.00e
+000 0.00e+000   0
   1 3.1817276e+001 0.00e+000 1.97e-002  -1.0 1.58e-002    -
 9.97e-001 1.00e+000f  1
   2 3.1897478e+001 0.00e+000 1.71e-002  -1.0 3.47e-001    -  1.00e
+000 1.00e+000f  1
   3 2.1502183e+001 0.00e+000 6.91e-003  -1.7 3.68e+000    -
 9.03e-001 1.00e+000f  1
   4 1.5386871e+001 0.00e+000 6.82e-003  -2.8 2.61e+000    -
 9.95e-001 1.00e+000f  1
   5 1.2207141e+001 0.00e+000 5.65e-003  -3.6 3.21e+000    -
 9.87e-001 1.00e+000f  1
   6 1.1168421e+001 0.00e+000 5.99e-003  -4.4 1.81e+000    -
 9.98e-001 1.00e+000f  1
   7 9.7620227e+000 0.00e+000 5.84e-003  -5.1 2.87e+000    -  1.00e
+000 1.00e+000f  1
   8 8.3740689e+000 0.00e+000 4.64e-003  -5.8 4.47e+000    -  1.00e
+000 8.55e-001f  1
   9 8.3531432e+000 0.00e+000 4.63e-003  -6.2 5.81e+000    -  1.00e
+000 6.88e-003f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du
 alpha_pr  ls
  10 7.9712470e+000 0.00e+000 4.41e-003  -6.7 1.39e+001    -  1.00e
+000 6.93e-002f  1
  11 7.3382838e+000 0.00e+000 4.23e-003  -6.7 1.74e+001    -  1.00e
+000 8.60e-002f  1
```

```
  12 6.8951250e+000 0.00e+000 6.95e-003  -6.9 1.86e+001    -
 9.31e-001 8.25e-002f  1
  13 6.8251706e+000 0.00e+000 6.73e-003  -3.5 1.58e+000    -
 3.14e-002 1.22e-001f  1
  14 6.5213531e+000 0.00e+000 1.20e-002  -5.1 1.50e+001    -
 9.99e-001 6.87e-002f  1
  15 5.8179752e+000 0.00e+000 8.20e-003  -4.1 1.44e+001    -
 2.41e-001 1.64e-001f  1
  16 5.6615052e+000 0.00e+000 6.89e-003  -4.8 1.61e+001    -
 9.98e-001 4.09e-002f  1
  17 5.5660484e+000 0.00e+000 1.60e-002  -2.9 8.26e-001    -
 1.39e-001 7.17e-001f  1
  18 5.2820314e+000 0.00e+000 1.05e-002  -3.8 1.43e+001    -
 5.92e-001 9.71e-002f  1
  19 4.8139590e+000 0.00e+000 3.83e-003  -4.3 1.37e+001    -
 8.38e-001 1.97e-001f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du
 alpha_pr  ls
  20 4.3962652e+000 0.00e+000 4.25e-003  -4.7 1.39e+001    -  1.00e
+000 2.14e-001f  1
  21 4.3497289e+000 0.00e+000 7.98e-003  -3.4 2.44e+000    -
 1.47e-001 1.73e-001f  1
  22 4.1103680e+000 0.00e+000 3.94e-003  -5.0 1.02e+001    -
 7.92e-001 2.03e-001f  1
  23 4.0452905e+000 0.00e+000 8.79e-003  -3.0 3.76e+000    -
 1.15e-001 1.43e-001f  1
  24 4.0428604e+000 0.00e+000 5.59e-002  -9.8 9.55e-001    -
 4.56e-001 2.11e-002f  1
  25 3.8936978e+000 0.00e+000 4.45e-002  -3.4 1.34e+001    -
 1.82e-001 1.52e-001f  1
  26 3.8799346e+000 0.00e+000 2.76e-002  -3.5 1.93e+000    -
 6.27e-001 8.40e-002f  1
  27 4.5627777e+000 0.00e+000 2.21e-002  -1.6 1.16e+002    -
 8.63e-002 3.29e-002f  1
  28 3.9601993e+000 0.00e+000 1.10e-002  -2.7 5.02e+000    -
 5.11e-001 1.00e+000f  1
  29 3.7665979e+000 0.00e+000 1.12e-002  -4.1 7.76e+000    -
 4.13e-001 2.37e-001f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du
 alpha_pr  ls
  30 3.7511793e+000 0.00e+000 1.32e-002  -8.9 8.12e-001    -
 6.81e-001 9.03e-002f  1
  31 3.7231452e+000 0.00e+000 1.24e-002  -3.4 6.45e+000    -
 1.17e-001 1.14e-001f  3
  32 3.6450224e+000 0.00e+000 2.00e-002  -4.8 1.13e+000    -  1.00e
+000 3.08e-001f  1
  33 3.4424090e+000 0.00e+000 5.90e-003  -3.5 2.00e+001    -
 7.89e-001 1.18e-001f  1
  34 3.4382629e+000 0.00e+000 3.64e-003  -3.7 7.49e+000    -
 9.66e-001 4.62e-002f  3
  35 3.4338959e+000 0.00e+000 7.92e-003  -9.7 5.83e-001    -
 2.32e-001 3.90e-002f  1
  36 3.1909479e+000 0.00e+000 3.84e-003  -4.9 4.28e+000    -
 6.83e-001 5.20e-001f  1
```

```
  37 3.1814594e+000 0.00e+000 1.98e-002  -4.5 1.59e+000    -  1.00e
+000 7.08e-002f  1
  38 3.1755642e+000 0.00e+000 4.59e-003  -5.2 4.01e+000    -
 9.97e-001 1.65e-001f  2
  39 3.1618919e+000 0.00e+000 1.63e-002  -4.9 1.01e+000    -
 9.57e-001 1.63e-001f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du
 alpha_pr  ls
  40 3.0589006e+000 0.00e+000 7.61e-003  -4.9 9.03e+000    -
 7.44e-001 1.81e-001f  1
  41 3.0488067e+000 0.00e+000 2.33e-003  -4.2 1.25e+001    -
 5.45e-001 1.10e-002f  4
  42 3.0375911e+000 0.00e+000 5.78e-003  -4.4 1.66e+000    -
 2.27e-001 1.78e-001f  1
  43 3.0366034e+000 0.00e+000 9.77e-003  -6.2 9.32e-002    -
 3.97e-001 8.27e-002f  1
  44 3.0163349e+000 0.00e+000 9.68e-003  -3.6 9.23e+000    -
 3.74e-001 1.66e-001f  1
  45 3.0150970e+000 0.00e+000 1.26e-002  -9.9 7.27e-001    -
 6.61e-001 4.81e-002f  1
  46 2.9537174e+000 0.00e+000 9.11e-003  -3.6 1.69e+001    -
 1.10e-001 1.97e-001f  1
  47 2.9430499e+000 0.00e+000 1.14e-002  -4.5 9.09e-001    -
 9.87e-001 1.49e-001f  1
  48 2.9272985e+000 0.00e+000 4.87e-003  -4.1 6.65e+000    -  1.00e
+000 2.21e-001f  1
  49 2.9245817e+000 0.00e+000 1.26e-002  -5.0 2.72e+000    -
 6.79e-001 4.54e-002f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du
 alpha_pr  ls
  50 2.8845169e+000 0.00e+000 3.14e-003  -5.2 7.38e+000    -
 7.31e-001 1.10e-001f  1
  51 2.8822060e+000 0.00e+000 2.14e-003  -5.7 7.21e+000    -
 8.40e-001 1.26e-002f  4
  52 2.8723832e+000 0.00e+000 2.39e-002  -4.2 7.72e-001    -  1.00e
+000 3.34e-001f  1
  53 2.8632641e+000 0.00e+000 1.29e-002  -3.9 2.52e-001    -
 4.87e-001 1.00e+000f  1
  54 4.8272193e+000 0.00e+000 1.16e-002  -2.0 7.54e+001    -
 2.50e-002 1.49e-001f  1
  55 3.8381635e+000 0.00e+000 8.54e-003  -3.4 1.66e+001    -
 1.64e-002 4.71e-001f  1
  56 3.7792074e+000 0.00e+000 5.36e-003  -3.4 9.72e+000    -
 4.32e-001 3.43e-002f  1
  57 3.3527218e+000 0.00e+000 4.62e-003  -3.4 1.03e+001    -  1.00e
+000 2.69e-001f  1
  58 2.9922063e+000 0.00e+000 2.53e-003  -3.4 9.40e+000    -
 8.60e-001 6.23e-001f  1
  59 2.9517168e+000 0.00e+000 8.26e-003  -5.1 2.03e+000    -
 8.35e-001 1.96e-001f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du
 alpha_pr  ls
  60 2.8193889e+000 0.00e+000 2.26e-003  -5.1 6.17e+000    -
 5.34e-001 4.13e-001f  1
```

```
 61 2.7364803e+000 0.00e+000 2.99e-003 -10.2 2.91e+000    -
 2.36e-001 2.28e-001f  1
 62 2.7356731e+000 0.00e+000 1.05e-002 -10.4 3.47e-001    -
 6.95e-001 2.09e-002f  1
 63 2.7385076e+000 0.00e+000 4.98e-003  -3.6 1.28e+001    -
 4.91e-001 1.91e-002f  4
 64 2.7267915e+000 0.00e+000 9.28e-003  -4.0 3.06e-001    -
 2.78e-001 5.52e-001f  1
 65 2.6299845e+000 0.00e+000 4.01e-003  -4.2 9.17e+000    -
 5.34e-001 1.70e-001f  1
 66 2.6220345e+000 0.00e+000 9.60e-003  -4.3 8.62e-001    -
 8.85e-001 1.71e-001f  1
 67 2.6185096e+000 0.00e+000 4.53e-002  -4.6 7.04e-001    -
 9.96e-001 8.00e-002f  1
 68 2.6067868e+000 0.00e+000 7.27e-003  -3.8 4.56e+000    -
 8.54e-001 4.37e-002f  4
 69 6.6305245e+000 0.00e+000 5.64e-003  -1.9 2.76e+002    -
 3.13e-002 1.16e-001f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du
 alpha_pr  ls
 70 2.9789722e+000 0.00e+000 4.08e-003  -3.1 2.29e+001    -  1.00e
+000 1.00e+000f  1
 71 2.8763730e+000 0.00e+000 2.61e-003  -3.1 8.01e+000    -
 6.81e-001 2.90e-001f  2
 72 2.8939924e+000 0.00e+000 2.87e-003  -3.1 6.07e-001    -  1.00e
+000 1.00e+000f  1
 73 2.6859626e+000 0.00e+000 1.80e-003  -4.7 5.53e+000    -
 8.23e-001 1.00e+000f  1
 74 2.6367161e+000 0.00e+000 1.74e-003  -4.7 5.44e+000    -
 4.08e-001 1.10e-001f  1
 75 2.6263869e+000 0.00e+000 2.13e-003  -4.7 1.04e+001    -
 2.37e-001 6.81e-002f  3
 76 2.6029707e+000 0.00e+000 8.77e-003  -4.7 1.02e+000    -
 7.32e-001 4.03e-001f  1
 77 2.5874070e+000 0.00e+000 1.83e-003  -4.0 9.33e+000    -
 6.91e-001 2.20e-001f  1
 78 2.5766211e+000 0.00e+000 4.73e-003  -5.7 8.23e-001    -  1.00e
+000 1.78e-001f  1
 79 2.5661821e+000 0.00e+000 1.81e-003  -4.4 3.57e+000    -
 6.12e-001 3.66e-002f  3
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du
 alpha_pr  ls
 80 2.5610822e+000 0.00e+000 2.75e-003  -4.5 9.11e-001    -
 1.91e-001 8.52e-002f  1
 81 2.5169488e+000 0.00e+000 1.73e-003  -5.2 3.90e+000    -
 9.99e-001 1.96e-001f  1
 82 2.6089070e+000 0.00e+000 2.99e-003  -3.3 4.48e+000    -
 3.17e-002 4.01e-001f  1
 83 2.5468759e+000 0.00e+000 1.72e-003  -4.5 6.08e+000    -
 6.55e-001 2.02e-001f  2
 84 2.5199829e+000 0.00e+000 4.16e-003  -4.5 8.55e-001    -  1.00e
+000 8.93e-001f  1
 85 2.5157005e+000 0.00e+000 8.82e-003  -4.5 5.91e-001    -
 7.97e-001 2.50e-001f  3
```

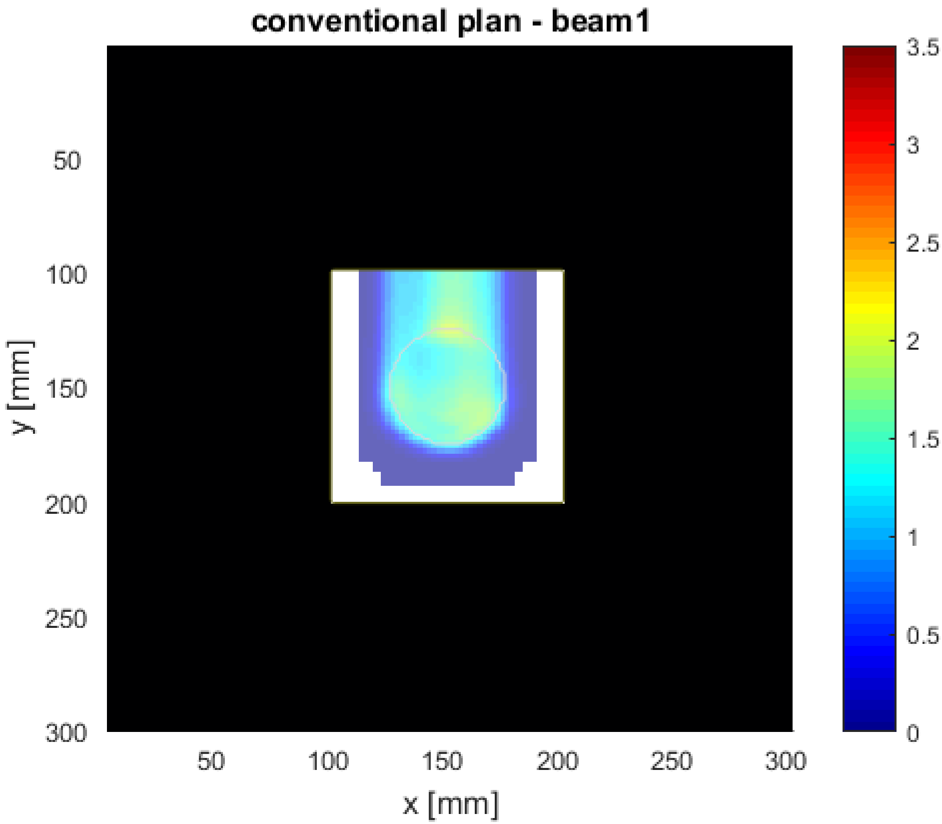
```
  86 2.5073972e+000 0.00e+000 8.46e-003  -5.6 5.26e-001    -  1.00e
+000 3.78e-001f  1
  87 2.5064796e+000 0.00e+000 7.69e-002  -5.7 7.19e-001    -  1.00e
+000 2.59e-002f  1
  88 2.4934543e+000 0.00e+000 5.13e-002 -11.0 7.16e+000    -
 3.36e-001 5.38e-002f  2
  89 2.4907493e+000 0.00e+000 3.43e-002 -11.0 1.17e+000    -
 6.23e-001 1.01e-001f  3
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du
 alpha_pr  ls
  90 2.4862100e+000 0.00e+000 2.72e-002  -6.4 6.79e-002    -  1.00e
+000 8.81e-001f  1
  91 2.4861375e+000 0.00e+000 1.84e-002  -7.4 1.05e-001    -  1.00e
+000 9.67e-003f  1
  92 2.4840593e+000 0.00e+000 1.41e-002 -11.0 2.31e+000    -
 1.80e-001 4.47e-002f  2
  93 2.4817970e+000 0.00e+000 2.09e-002  -8.2 2.06e-001    -  1.00e
+000 3.17e-001f  1

Number of Iterations....: 93

                                   (scaled)                 (unscaled)
Objective...............:  2.4817969714687096e+000
 2.4817969714687096e+000
Dual infeasibility......:  2.0912872362818229e-002
 2.0912872362818229e-002
Constraint violation....:  0.0000000000000000e+000
 0.0000000000000000e+000
Complementarity.........:  1.1765452496183732e-005
 1.1765452496183732e-005
Overall NLP error.......:  2.0912872362818229e-002
 2.0912872362818229e-002


Number of objective function evaluations             = 183
Number of objective gradient evaluations             = 94
Number of equality constraint evaluations            = 0
Number of inequality constraint evaluations          = 0
Number of equality constraint Jacobian evaluations   = 0
Number of inequality constraint Jacobian evaluations = 0
Number of Lagrangian Hessian evaluations             = 0
Total CPU secs in IPOPT (w/o function evaluations)   =      2.414
Total CPU secs in NLP function evaluations           =     29.520

EXIT: Solved To Acceptable Level.
Calculating final cubes...
```

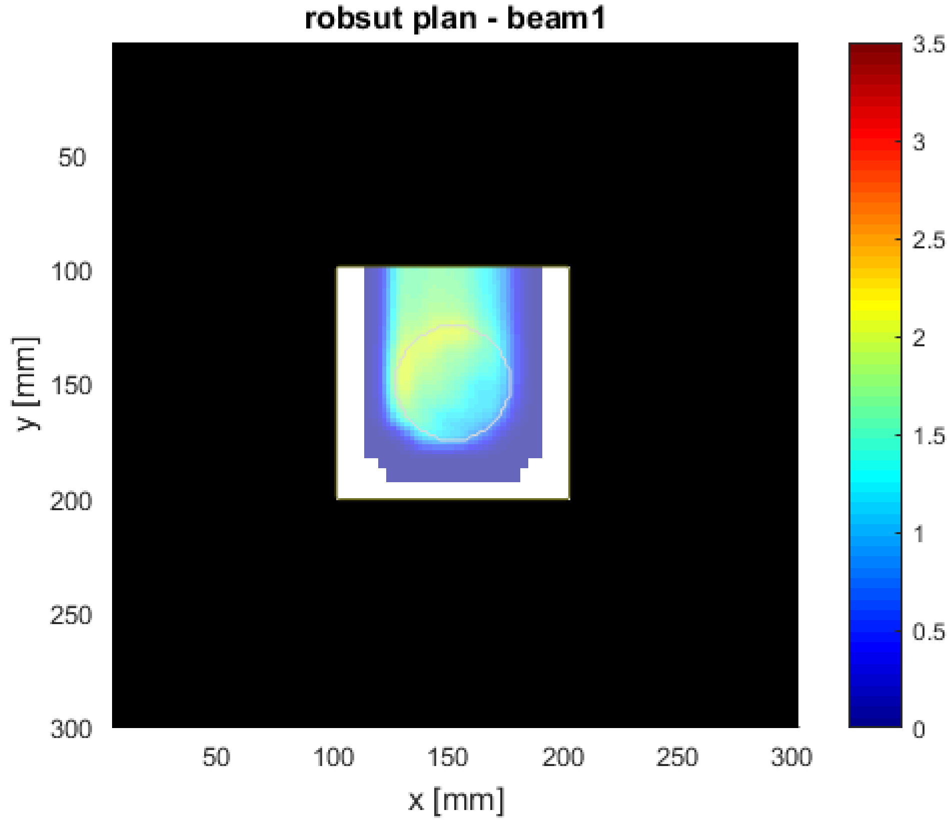


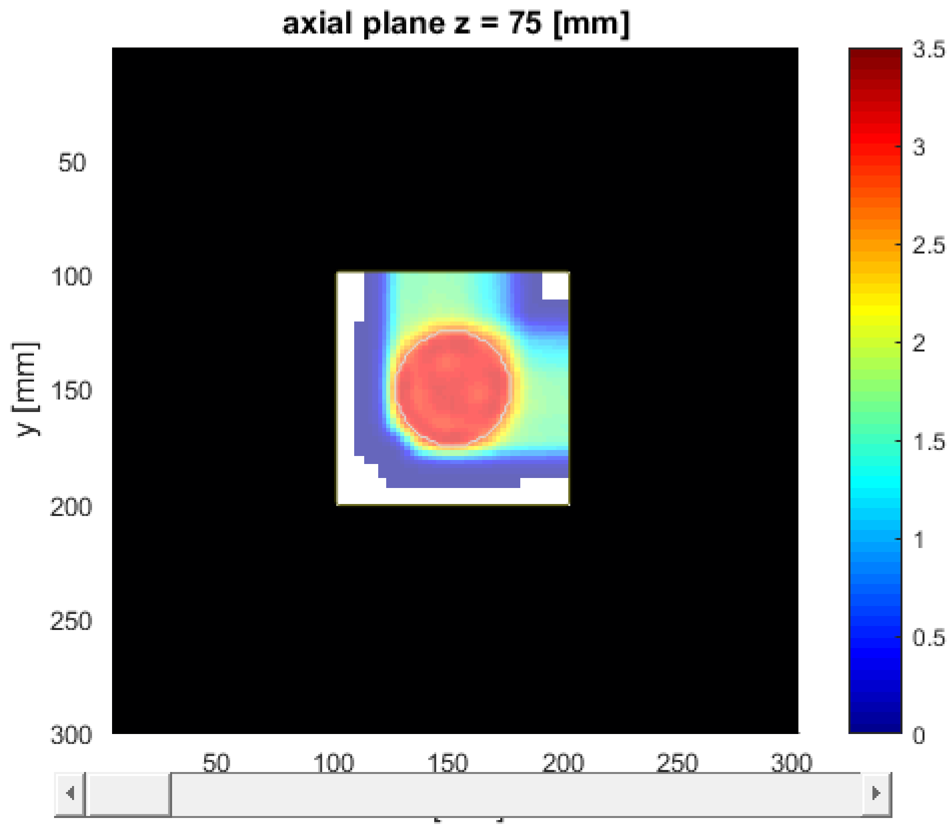
Visualize results

```
addpath('tools')
plane      = 3;
slice      = round(pln.propStf.isoCenter(1,3)./ct.resolution.z);
doseWindow = [0 3.5];

figure,matRad_plotSliceWrapper(gca,ct,cst,1,resultGUI.RBExD_beam1
  ,plane,slice,[],[],colorcube,[],doseWindow,[]);title('conventional
 plan - beam1')
figure,matRad_plotSliceWrapper(gca,ct,cst,1,resultGUIrobust.RBExD_beam1,plane,slic
[],[],colorcube,[],doseWindow,[]);title('robsut plan - beam1')

% create an interactive plot to slide through individual scnearios
f = figure;title('individual scenarios');
numScen = 1;
matRad_plotSliceWrapper(gca,ct,cst,1,resultGUIrobust.(['RBExD_'
 num2str(round(numScen))]),plane,slice,[],[],colorcube,[],doseWindow,
[]);
b = uicontrol('Parent',f,'Style','slider','Position',[50,5,419,23],...

  'value',numScen, 'min',1, 'max',pln.multScen.totNumScen,'SliderStep',
 [1/(pln.multScen.totNumScen-1) , 1/(pln.multScen.totNumScen-1)]);
b.Callback = @(es,ed)
 matRad_plotSliceWrapper(gca,ct,cst,1,resultGUIrobust.(['RBExD_'
```

```
 num2str(round(es.Value))]),plane,slice,[],[],colorcube,[],doseWindow,
[]);
```



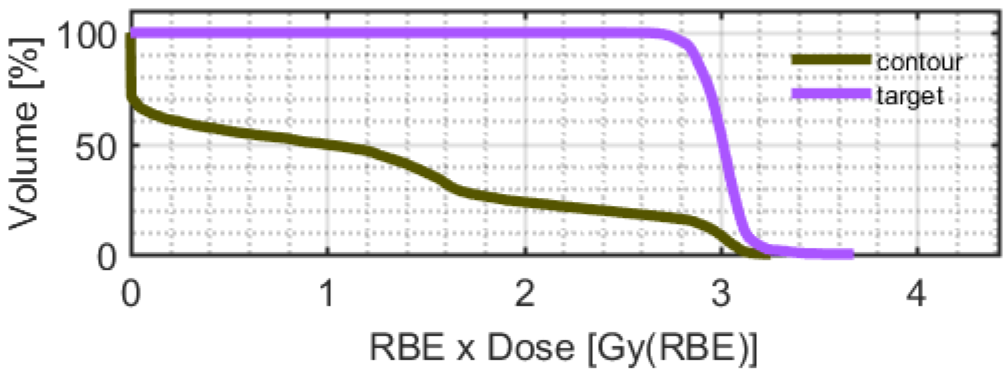

robsut plan - beam1
y [mm]
x [mm]
50
100
150
200
250
300
3.5
3
2.5
2
1.5
1
0.5
0


axial plane z = 75 [mm]
y [mm]
50
100
150
200
250
300
3.5
3
2.5
2
1.5
1
0.5
0

Indicator calculation and show DVH and QI

```
[dvh,qi] = matRad_indicatorWrapper(cst,pln,resultGUIrobust);
```

```
 0                contour - Mean dose =  1.12 Gy +/-  1.12 Gy (Max dose
=  3.26 Gy, Min dose =  0.00 Gy)
                          D2% =  3.11 Gy, D5% =  3.05 Gy, D50% =
0.96 Gy, D95% =  0.00 Gy, D98% =  0.00 Gy,
                          V0Gy = 100.00%, V0.7Gy =  53.17%, V1.4Gy =
40.83%, V2.2Gy =  21.53%, V2.9Gy =  13.57%, V3.6Gy =   0.00%,

 1                 target - Mean dose =  3.01 Gy +/-  0.11 Gy (Max dose
=  3.68 Gy, Min dose =  2.51 Gy)
                          D2% =  3.25 Gy, D5% =  3.18 Gy, D50% =
3.02 Gy, D95% =  2.83 Gy, D98% =  2.76 Gy,
                          V0Gy = 100.00%, V0.7Gy = 100.00%, V1.4Gy =
100.00%, V2.2Gy = 100.00%, V2.9Gy =  84.18%, V3.6Gy =   0.02%,
                          CI = 0.9185, HI = 11.88 for reference dose
of 3.0 Gy
```



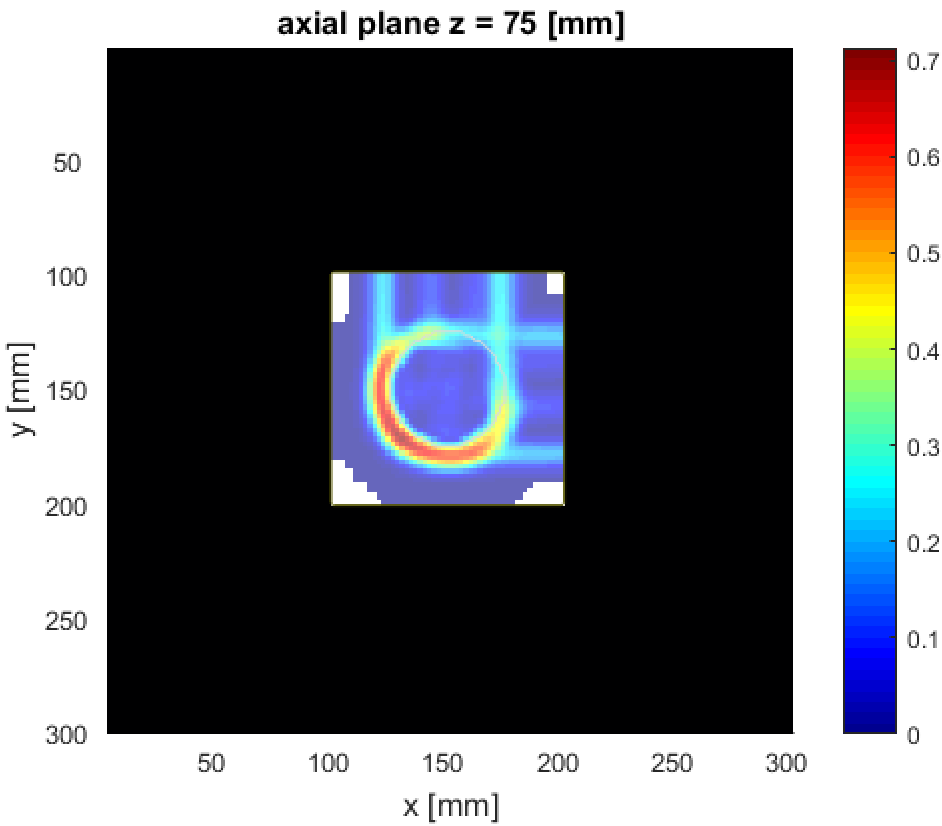
	mean	std	max	min	D_2
contour	1.1203	1.1241	3.2611	0	3.1055
target	3.0102	0.1145	3.6825	2.5143	3.2545

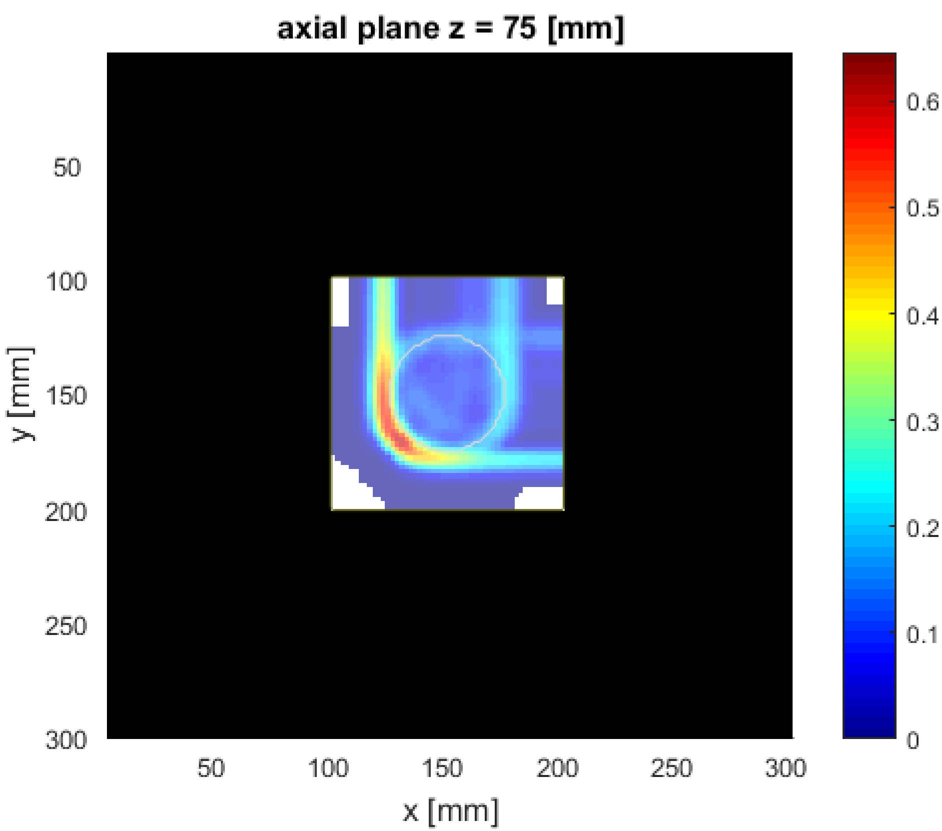
Perform sampling

```
addpath(['tools' filesep 'samplingAnalysis'])
```

```
% select structures to include in sampling; leave empty to sample dose
 for all structures
structSel = {}; % structSel = {'PTV','OAR1'};
[caSamp, mSampDose, plnSamp, resultGUInomScen]          =
 matRad_sampling(ct,stf,cst,pln,resultGUI.w,structSel,[],[]);
[cstStat, resultGUISamp, param]                          =
 matRad_samplingAnalysis(ct,cst,plnSamp,caSamp, mSampDose,
 resultGUInomScen,[]);

[caSampRob, mSampDoseRob, plnSampRob, resultGUInomScen] =
 matRad_sampling(ct,stf,cst,pln,resultGUIrobust.w,structSel,[],[]);
[cstStatRob, resultGUISampRob, paramRob]                 =
 matRad_samplingAnalysis(ct,cst,plnSampRob,caSampRob, mSampDoseRob,
 resultGUInomScen,[]);

figure,title('std dose cube based on sampling - conventional')
matRad_plotSliceWrapper(gca,ct,cst,1,resultGUISamp.stdCube,plane,slice,
[],[],colorcube,[],[0 max(resultGUISamp.stdCube(:))],[]);

figure,title('std dose cube based on sampling - robust')
matRad_plotSliceWrapper(gca,ct,cst,1,resultGUISampRob.stdCube,plane,slice,
[],[],colorcube,[],[0 max(resultGUISampRob.stdCube(:))],[]);

Using 20samples in total
matRad: Realizations variable will need: 0.0036276 GB
Warning: Could not check out parallel computing toolbox.
matRad: Particle dose calculation...
shift scenario 1 of 1:
matRad: Particle dose calculation...
Warning: ray does not hit patient. Trying to fix afterwards...Beam 1
 of 2:
matRad: calculate radiological depth cube...done.
matRad: calculate lateral cutoff...done.
Progress: 100.00 %
Beam 2 of 2:
matRad: calculate radiological depth cube...done.
matRad: calculate lateral cutoff...done.
Progress: 100.00 %
Finished nominal Scenario Calculation. Computation time: 0h
  0               contour - Mean dose =  1.02 Gy +/-  1.11 Gy (Max dose
 =  3.11 Gy, Min dose =  0.00 Gy)
                           D2% =  3.05 Gy, D5% =  3.02 Gy, D50% =
 0.66 Gy, D95% =  0.00 Gy, D98% =  0.00 Gy,
                           V0Gy = 100.00%, V0.638Gy =  50.25%, V1.28Gy
 =  37.44%, V1.91Gy =  21.90%, V2.55Gy =  17.37%, V3.19Gy =   0.00%,

  1                target - Mean dose =  2.99 Gy +/-  0.07 Gy (Max dose
 =  3.19 Gy, Min dose =  2.65 Gy)
                           D2% =  3.09 Gy, D5% =  3.07 Gy, D50% =
 3.00 Gy, D95% =  2.83 Gy, D98% =  2.77 Gy,
                           V0Gy = 100.00%, V0.638Gy = 100.00%, V1.28Gy
 = 100.00%, V1.91Gy = 100.00%, V2.55Gy = 100.00%, V3.19Gy =   0.01%,
                           CI = 0.9355, HI =  8.32 for reference dose
 of 3.0 Gy
```

```
Approximate Total calculation time: 0.03h. Estimated finish: 23-
Mar-2018 18:03:37
Progress: 100.00 %
matRad: Performing gamma index analysis with parameters2  2[Using
 20samples in total
matRad: Realizations variable will need: 0.0036276 GB
Warning: Could not check out parallel computing toolbox.
matRad: Particle dose calculation...
shift scenario 1 of 1:
matRad: Particle dose calculation...
Warning: ray does not hit patient. Trying to fix afterwards...Beam 1
 of 2:
matRad: calculate radiological depth cube...done.
matRad: calculate lateral cutoff...done.
Progress: 100.00 %
Beam 2 of 2:
matRad: calculate radiological depth cube...done.
matRad: calculate lateral cutoff...done.
Progress: 100.00 %
Finished nominal Scenario Calculation. Computation time: 0h
  0              contour - Mean dose =  1.12 Gy +/-  1.12 Gy (Max dose
 =  3.26 Gy, Min dose =  0.00 Gy)
                           D2% =  3.11 Gy, D5% =  3.05 Gy, D50% =
 0.96 Gy, D95% =  0.00 Gy, D98% =  0.00 Gy,
                           V0Gy = 100.00%, V0.737Gy =  52.83%, V1.47Gy
 =  37.99%, V2.21Gy =  21.44%, V2.95Gy =  11.87%, V3.68Gy =   0.00%,

  1               target - Mean dose =  3.01 Gy +/-  0.11 Gy (Max dose
 =  3.68 Gy, Min dose =  2.51 Gy)
                           D2% =  3.25 Gy, D5% =  3.18 Gy, D50% =
 3.02 Gy, D95% =  2.83 Gy, D98% =  2.76 Gy,
                           V0Gy = 100.00%, V0.737Gy = 100.00%, V1.47Gy
 = 100.00%, V2.21Gy = 100.00%, V2.95Gy =  74.41%, V3.68Gy =   0.01%,
                           CI = 0.9185, HI = 11.88 for reference dose
 of 3.0 Gy

Approximate Total calculation time: 0.04h. Estimated finish: 23-
Mar-2018 18:05:48
Progress: 100.00 %
matRad: Performing gamma index analysis with parameters2  2[
```


axial plane z = 75 [mm]
y [mm]
x [mm]
50
100
150
200
250
300
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0


axial plane z = 75 [mm]
y [mm]
x [mm]
50
100
150
200
250
300
0.6
0.5
0.4
0.3
0.2
0.1
0

Published with MATLAB® R2017a