Optimization tools for Radiance

KHALED NASSAR

MOHAMED ALY

THE AMERICAN UNIVERSITY IN CAIRO
الجـامـعـة الأمـريكـيـة بالقـاهـرة

- Lets start with some examples of design optimizations in real world problems. I will present two problems

Why optimization

Problem: Design the roof shading to maximize the shading on the surface and at the same time provide for enough ventilation for the roof top equipment.

A Quick and "dirty" formulation was needed due to the time constraints for the project

We try to find some design parameters related to the spacing, size, angle as even material of the sun shades. The objective will be a function with time and complexity required to solve the problem. One can even use rcontrib to calculate annual contribution of the sky/sun on the roof surface given a certain configuration
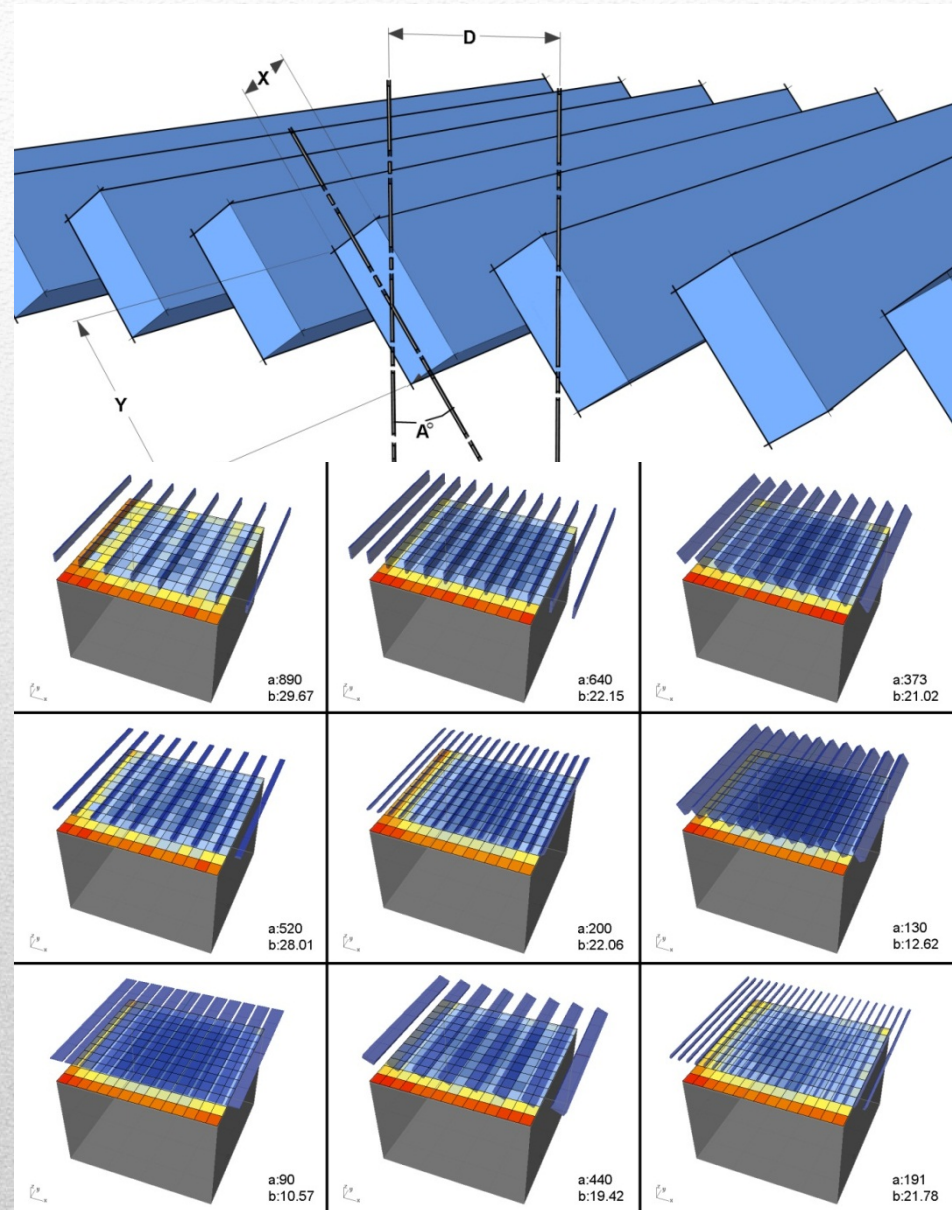
Problem 2 involves design a light pocket to maximize lighting levels on the inside rooms. You will be able to vary the dimensions of the pocket within constraints as well as the materials used.

Under clear sky conditions the solution is not always obvious

You could use trial and error but with just 5 variables each with 10 different options you end up with 1000s of solutions and the solution may not always be clear
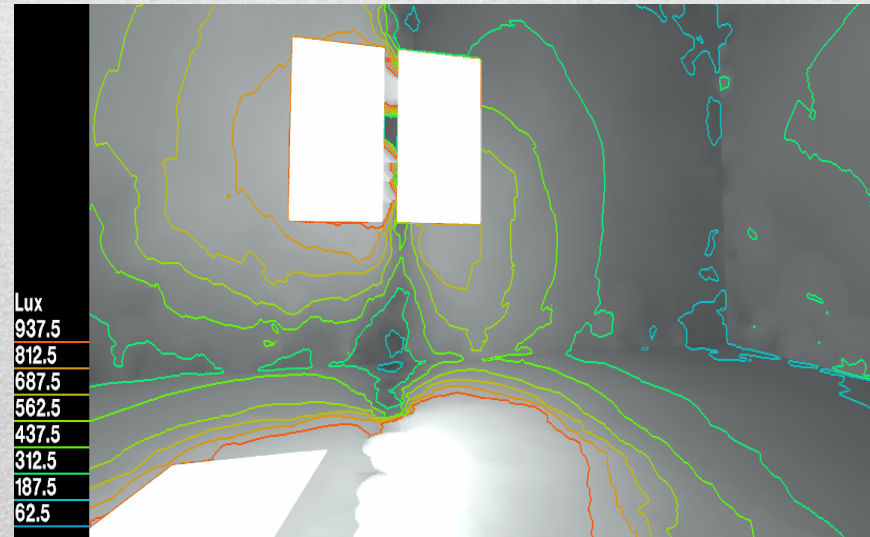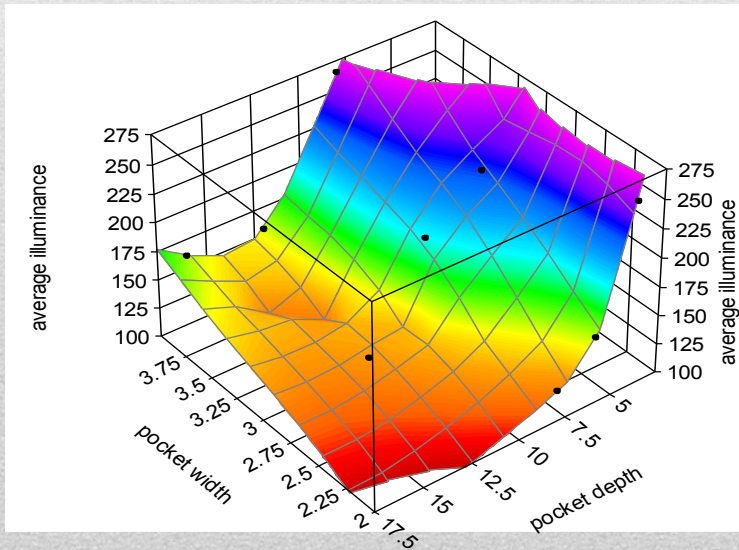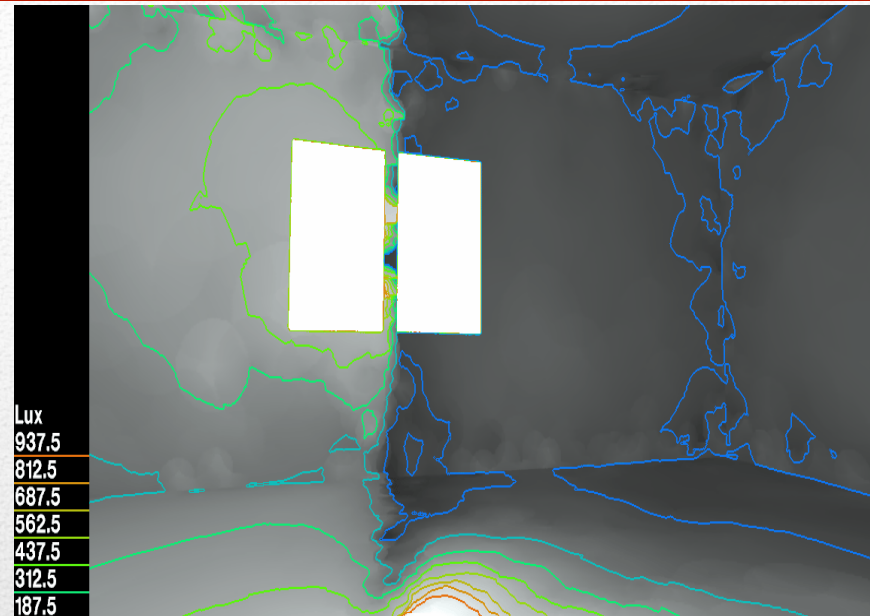
- This tutorial will demonstrate how to do optimization using radiance. It only assumes that you are familiar with radiance. Other software will be introduced and we will show how to use these tools to do optimization but other functions not related to the optimization will not be introduced.

What this tutorial is about

- Optimization algorithms for functions: $y = x^2 + 2x$
- Optimizing functions without a closed form for the gradient such as the Radiance. Functions may be non-smooth, discontinuous and thus non-differentiable or we are not able to calculate a gradient because it is a result of a numerical method such as Monte-Carlo Sampling
- Evolutionary techniques and non gradient based techniques
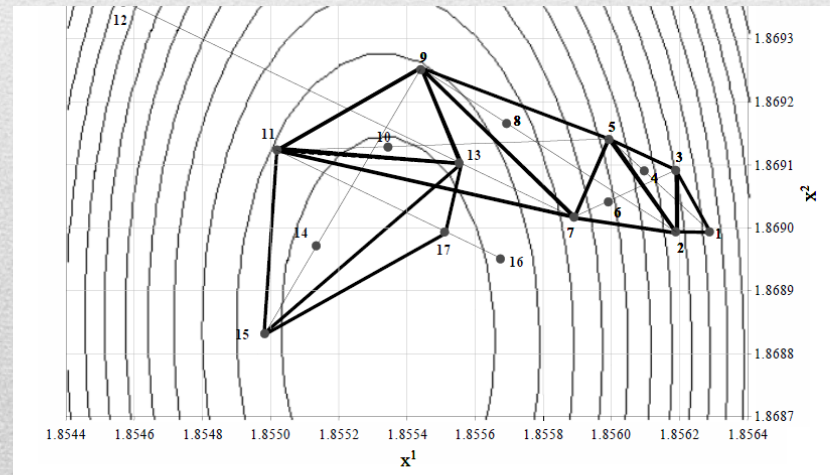- Overview of gradient-based techniques

Optimization Techniques

The Discrete Armijo Gradient algorithm approximates gradients by finite differences. Since the Discrete Armijo Gradient algorithm is sensitive to discontinuities in the cost function, LBL recommends not to use this algorithm if the simulation program contains adaptive solvers.
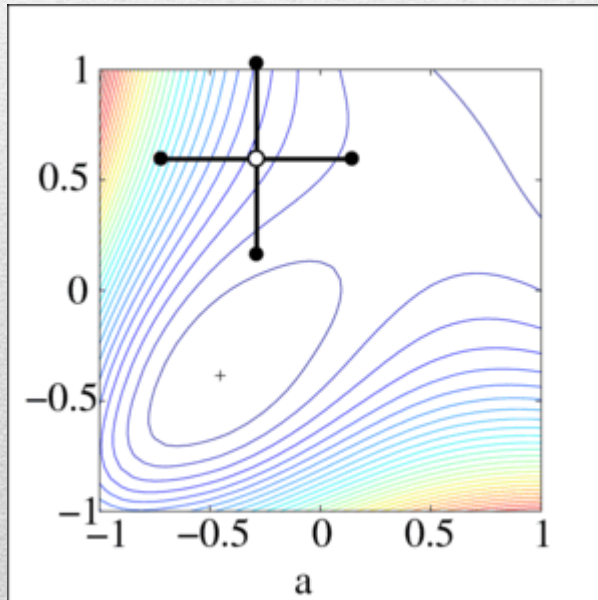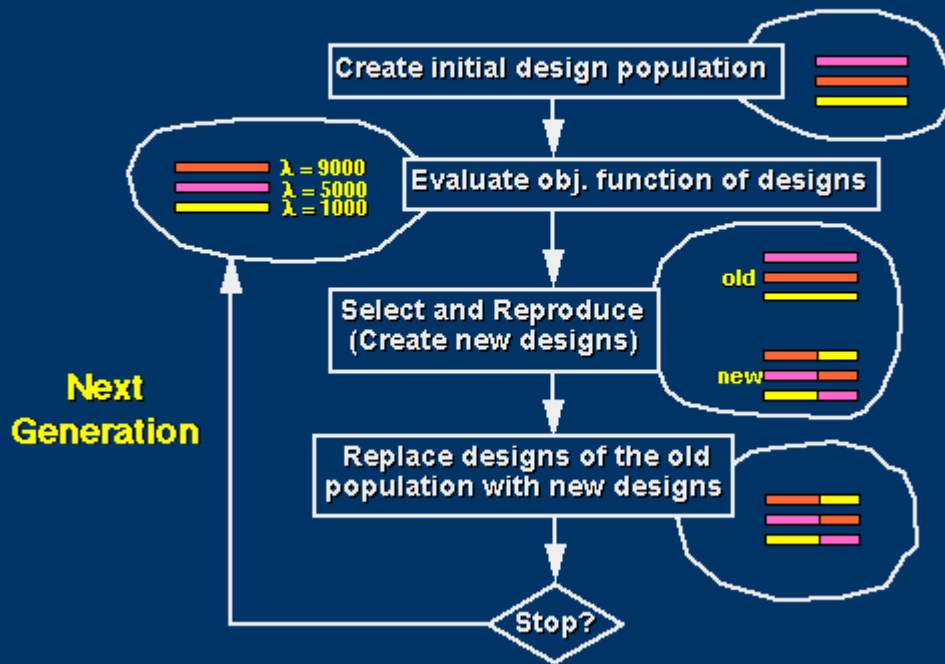
The Simplex algorithm constructs an n-dimensional simplex in the space that is spanned by the independent parameters. At each of the (n+1) vertices of the simplex, the value of the cost function is evaluated. In each iteration step, the point with the highest value of the cost function is replaced by another point. The algorithm consists of three main operations: (a) point reflection, (b) contraction of the simplex and (c) expansion of the simplex.

Gradient based techniques

GA Flowchart

Create initial design population

Evaluate obj. function of designs

$\lambda = 9000$
$\lambda = 5000$
$\lambda = 1000$

Select and Reproduce (Create new designs)

old

new

Next Generation

Replace designs of the old population with new designs

Stop?

Derivative Free. What **GPS** algorithms have in common is that they define the construction of a mesh, which is then explored according to some rules that differ among the various members of the family of GPS algorithms. If no decrease in cost is obtained on mesh points around the current iterate, then the distance between the mesh points is reduced, and the process is repeated.

Particle Swarm

Genetic Algorithms

# Evolutionary Techniques and non gradient techniques

- Two ways to do optimization using radiance
- We will compare between two techniques on the same example
- Using Rhino + Grasshopper + Galapagos
- The example is a simple room with a window and we would like to optimize the design of the room.
- We will divide the kinds of parameters we can optimize into two categories, geometric and non-geometric

Two ways to optimize using Radiance

ALTERNATIVE 1.

OPTIMIZATION USING DIVA RHINO GRASSHOPPER AND GALAPAGOS
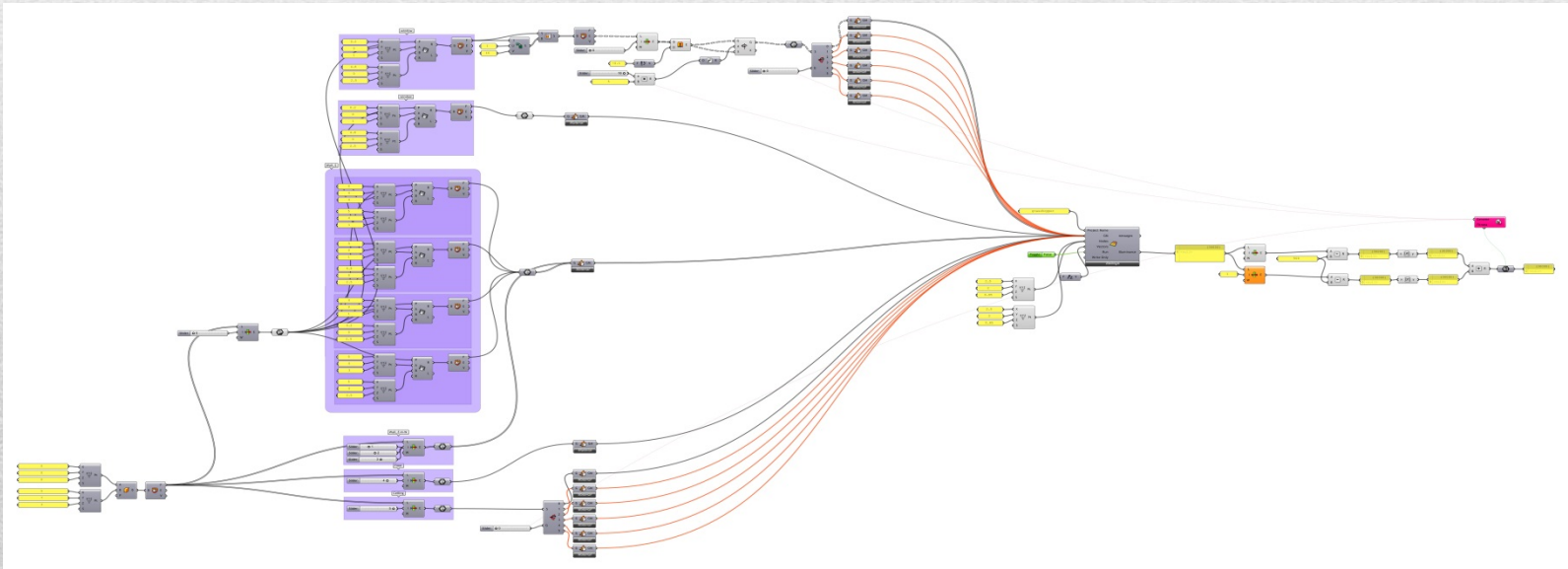
Step 0. Makes sure you have grasshoper, diva for grasshoper and galapago s installed.

Step 1. Draw or import the design into Rhino. You can draw the room in rhi no by...if you are using diva for rhino you can only do single analysis and n ot iterative or algorithmic design optimization. Therefore you need to use di va cor grasshoper. You will draw the same room algorithmiclly using grassh oper and the link it to radiance using diva for grasshopper.

To draw the room algorithmically using grasshoper:

Step 2. Now we can change the various geometric parameters in grasshoper the use diva for grasshoper to link to radiance and we are able to vary the sl iders related to these variables in order to observe the effect on the radiance analysis output. In order to do optimization you need to link this grasshoper definition to an optimization tool which is galapagos. So in this step we wil l a. use the parameter sliders as inputs to galapagos, b. define an objective f unction, and c. set optimization parameters

Step 3. Running the optimization and vary the parameters to try and get a b etter result
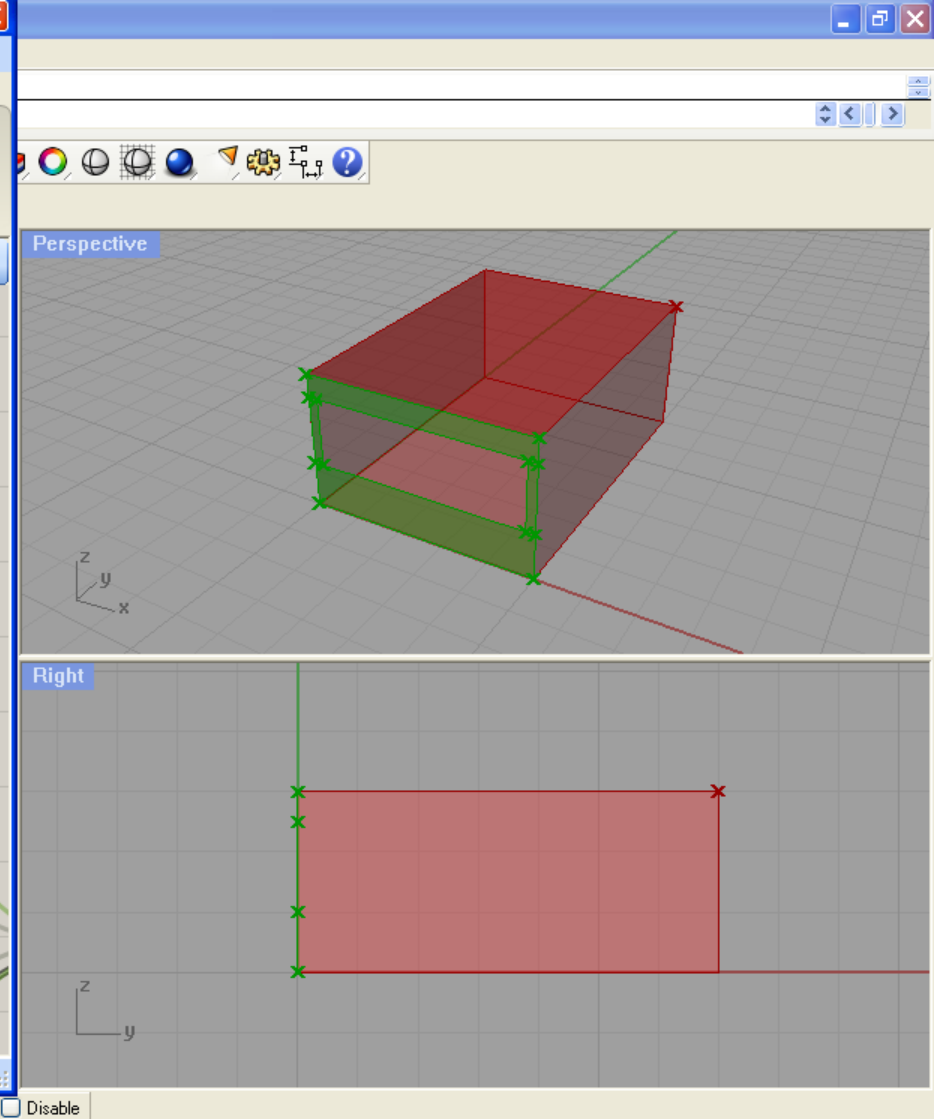
Steps Required
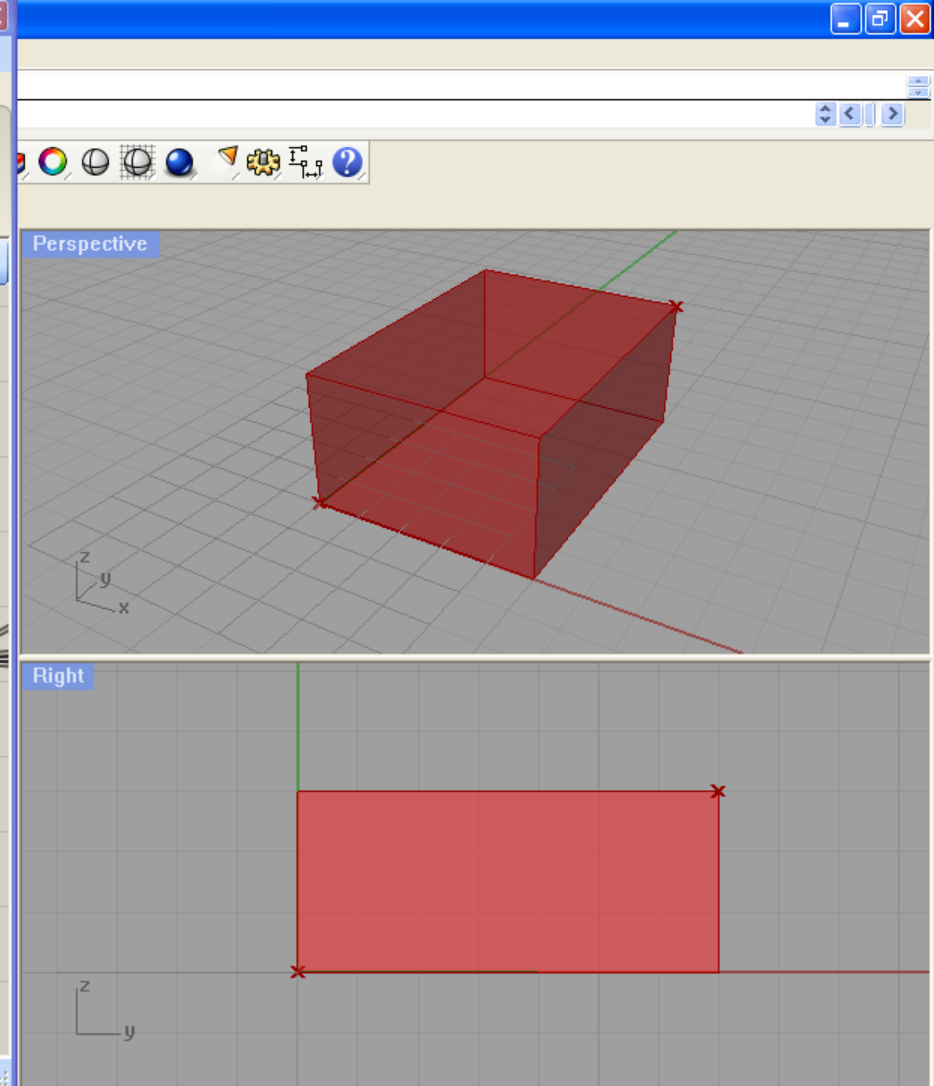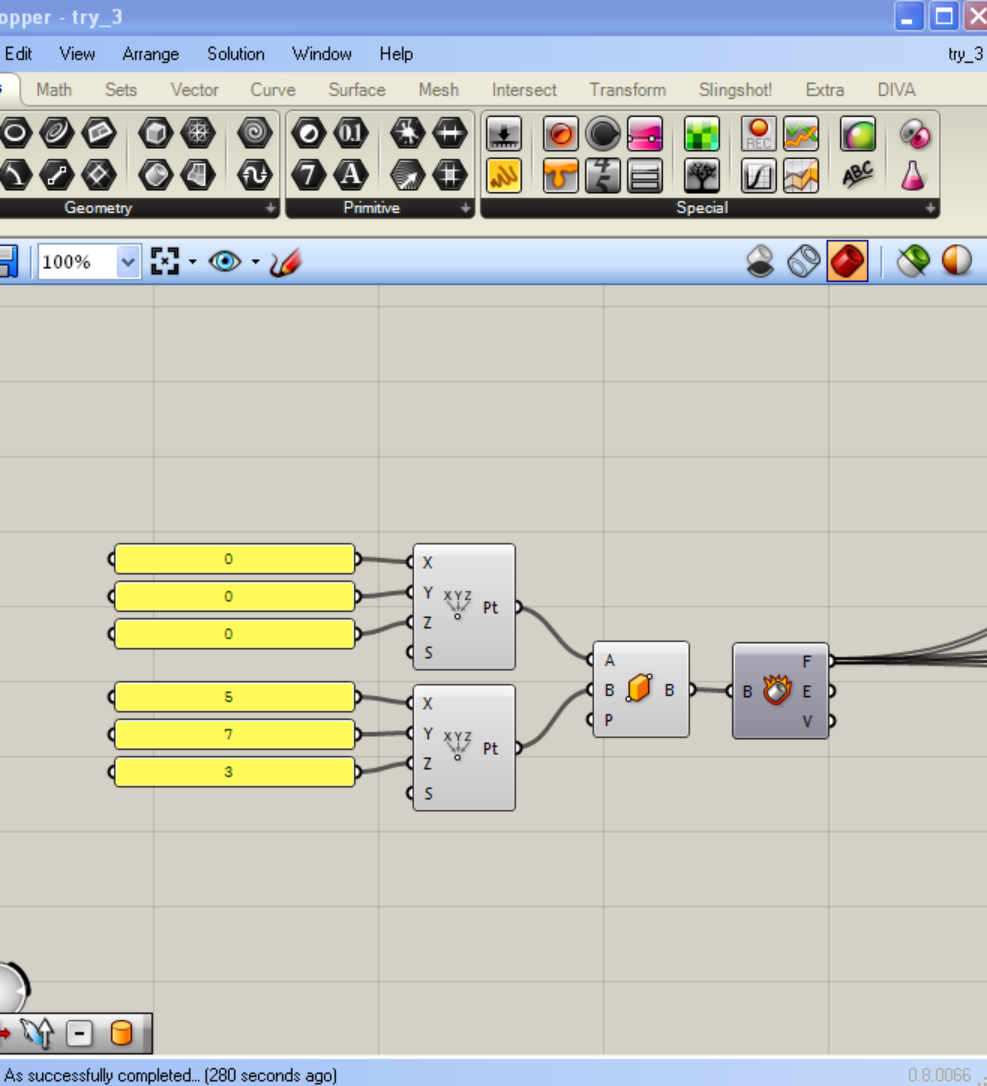
Creating a box using Two points, representing our tested

Exploding the geometry into surfaces to be assigned with the proper material

12.09.2012 11 International Radiance Workshop Copenhagen Denmark

Meshing and assigning materials to different surfaces

12.09.2012 11 International Radiance Workshop Copenhagen Denmark

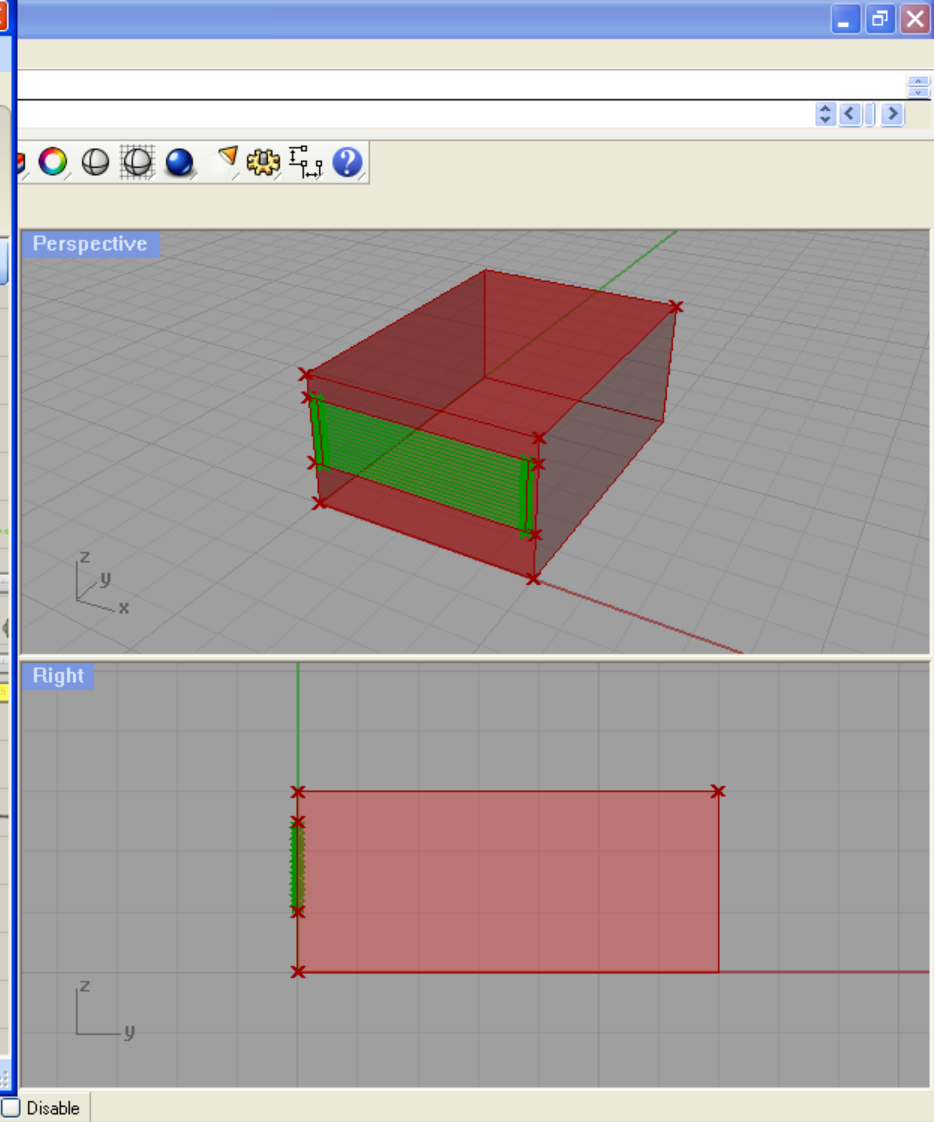As for the south wall recreating the wall into 4 surfaces

12.09.2012 11 International Radiance Workshop Copenhagen Denmark
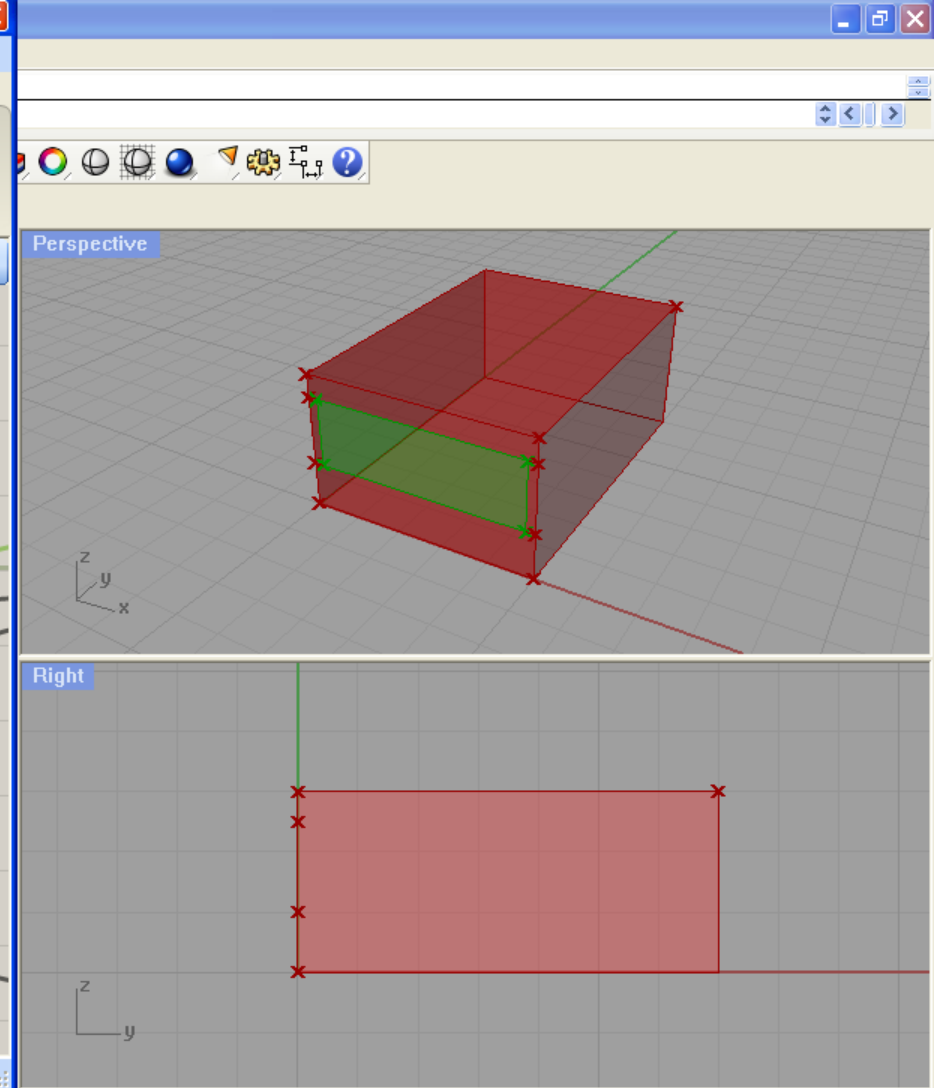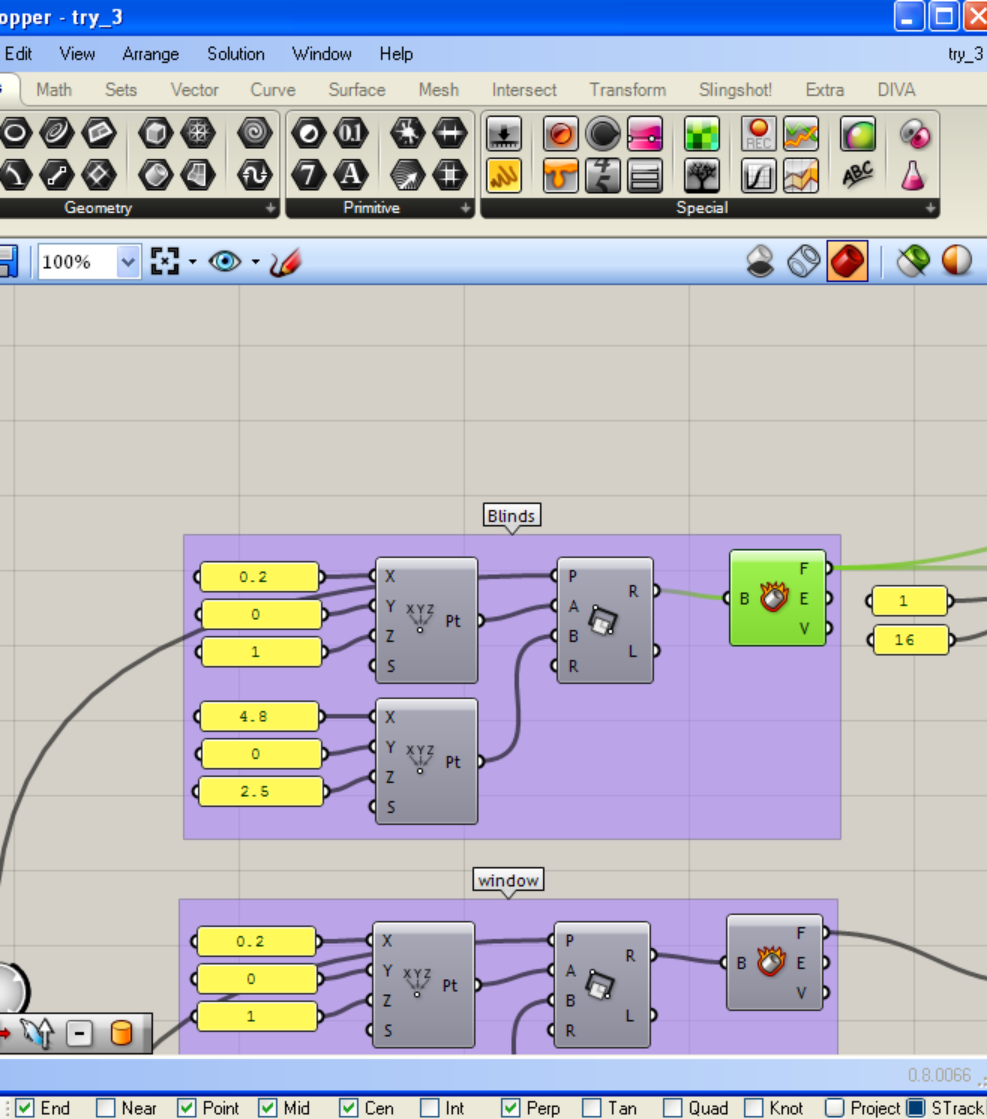
Creating the window plane

Recreating the blinds plane and subdividing into 16 horizontal surface
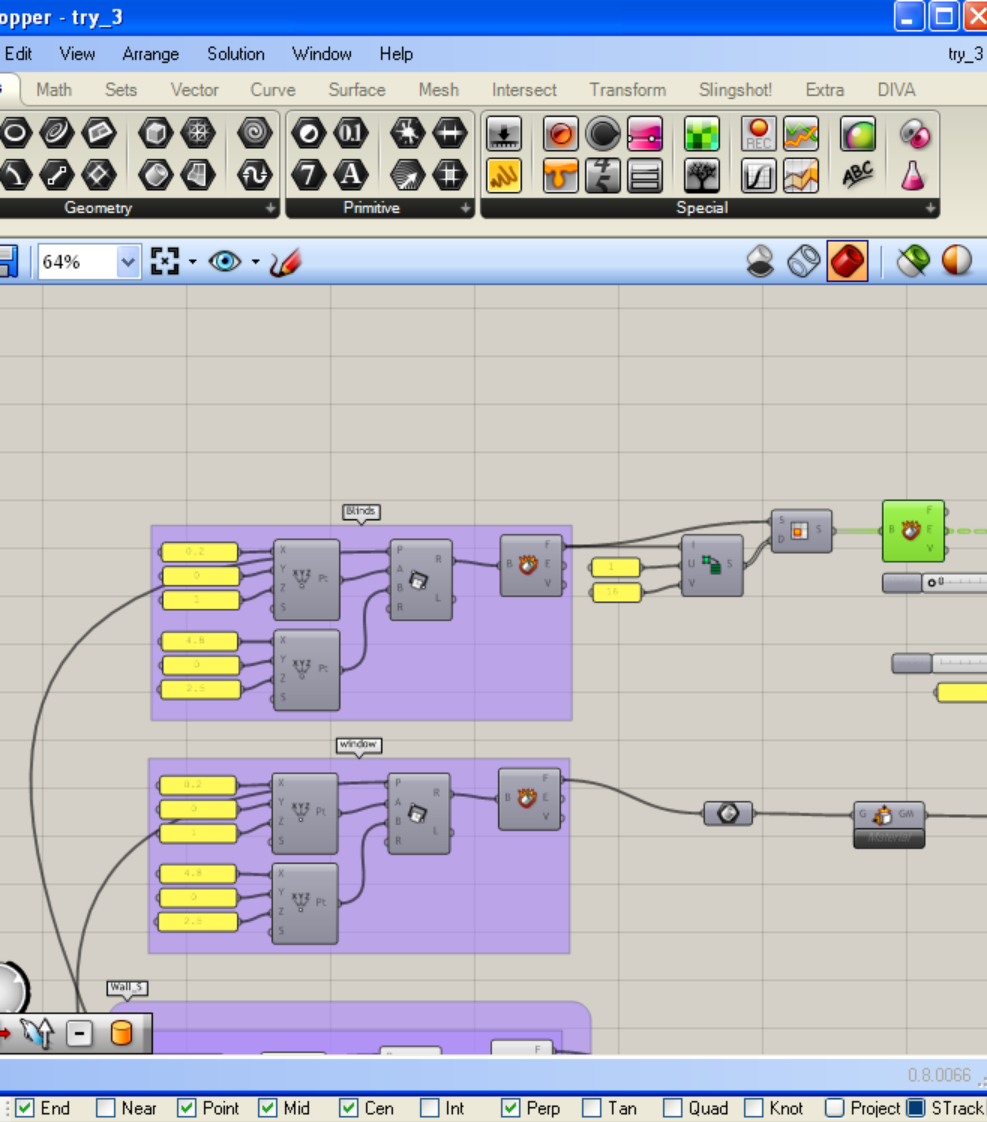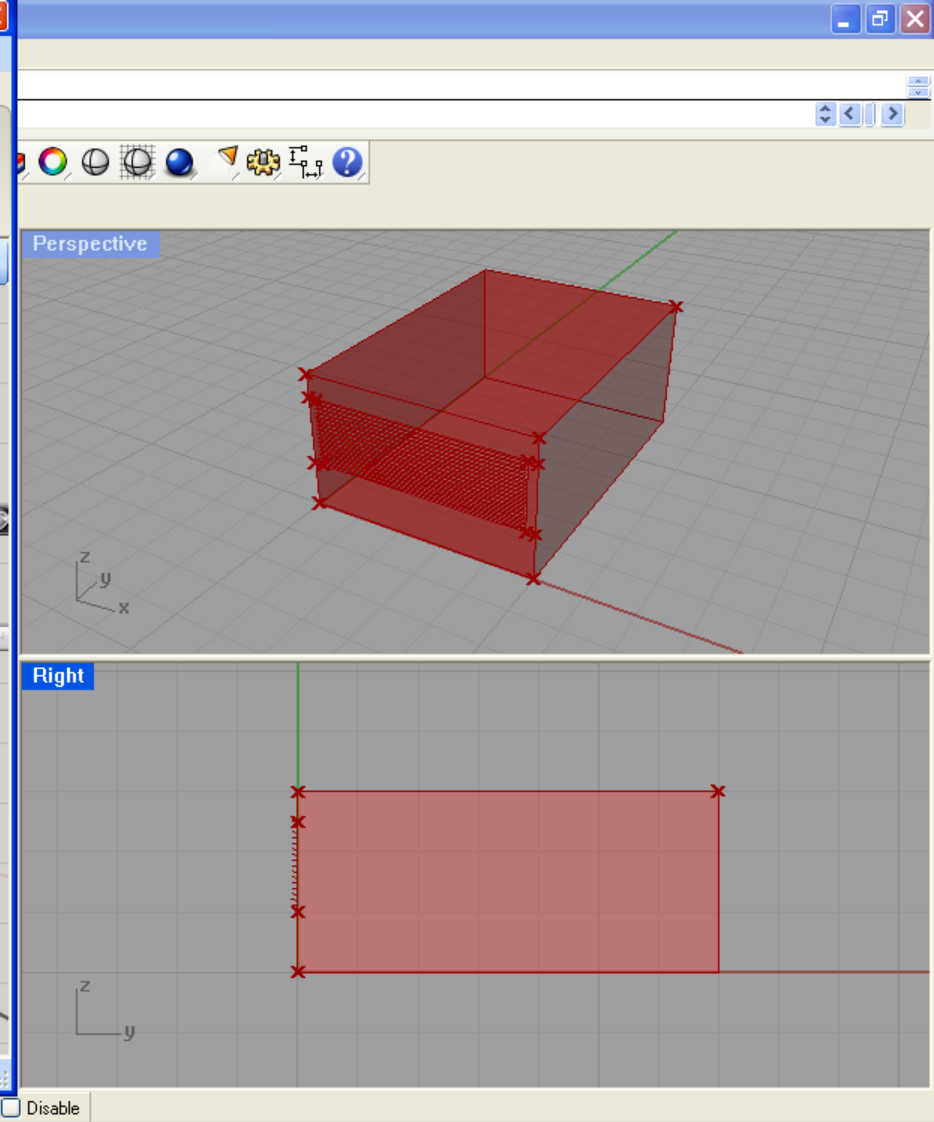
Adding the angle as a parameter defining the blinds

12.09.2012 11 International Radiance Workshop Copenhagen Denmark

The fitness function

12.09.2012 11 International Radiance Workshop Copenhagen Denmark
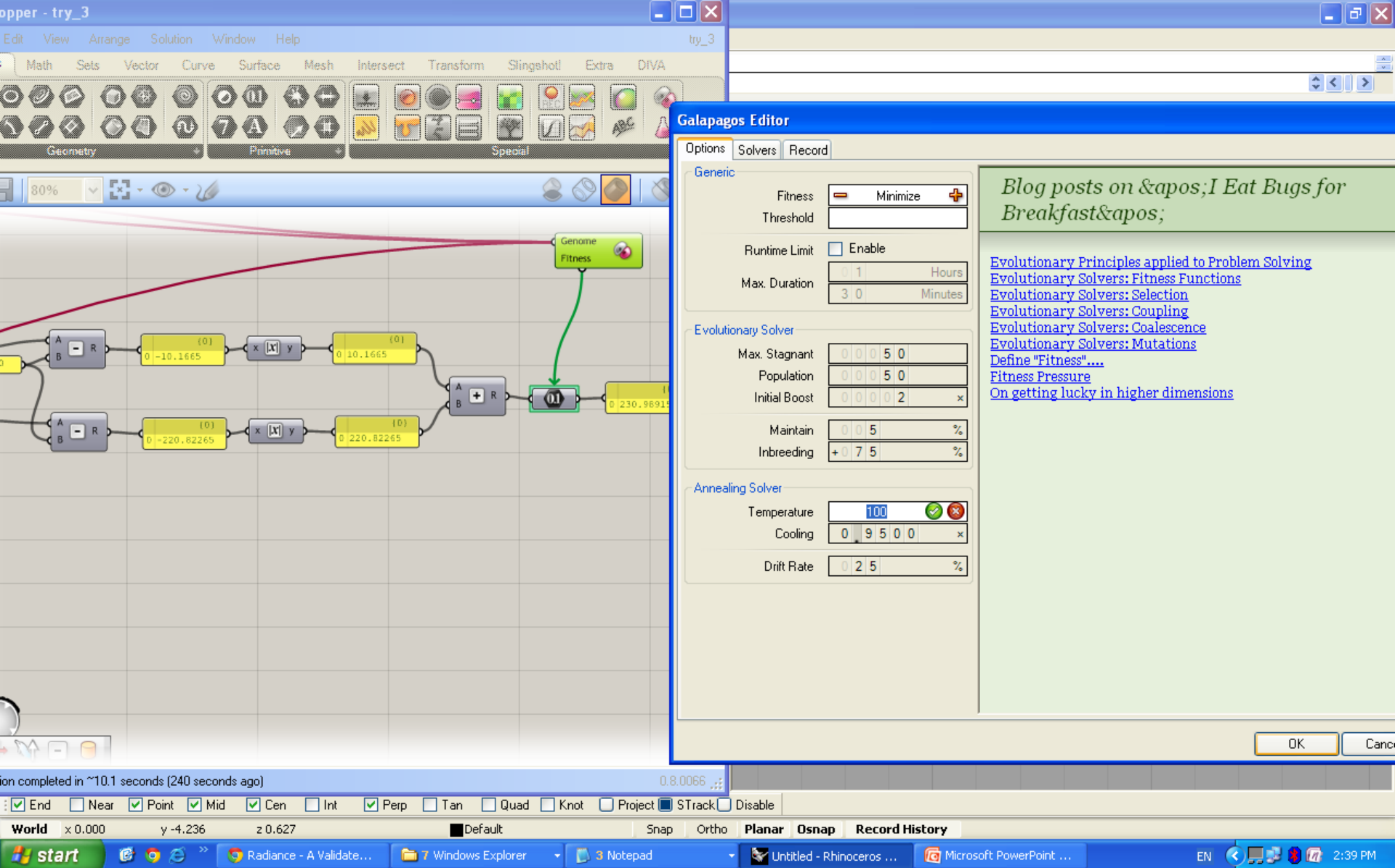
Start galapagos optimization

12.09.2012 11 International Radiance Workshop Copenhagen Denmark
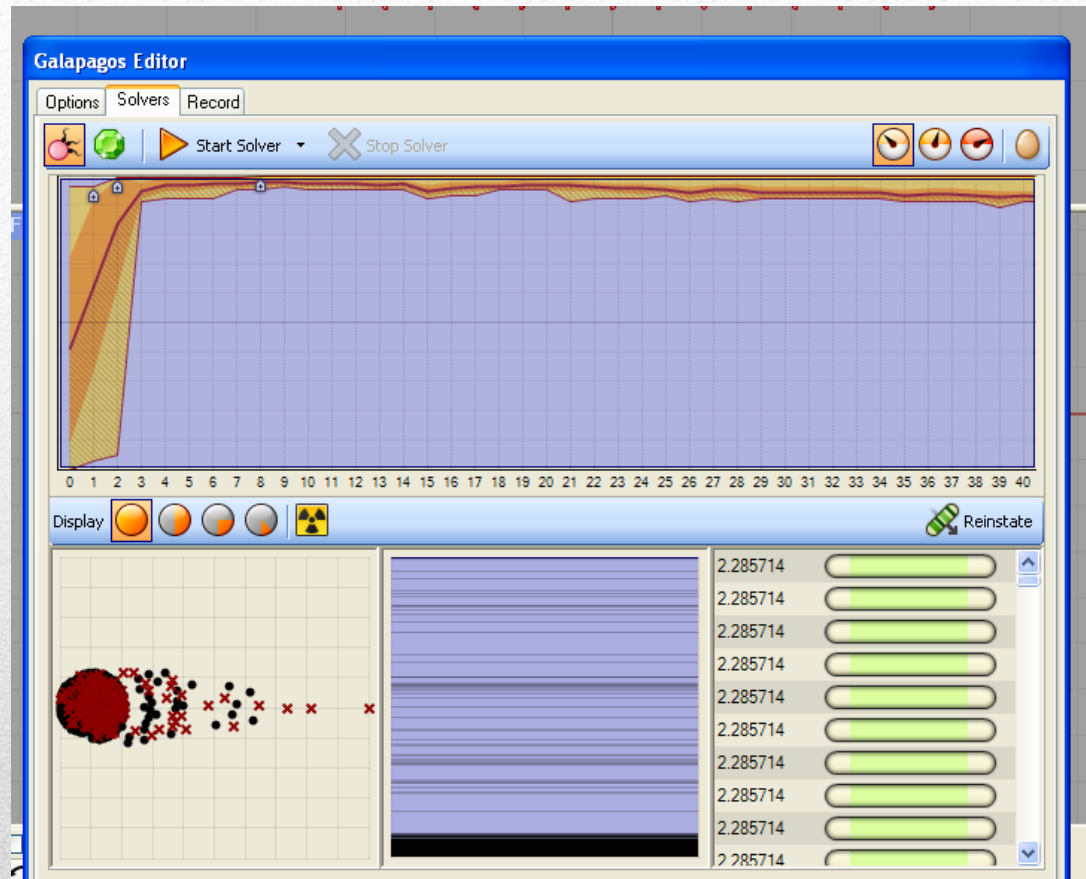
Results

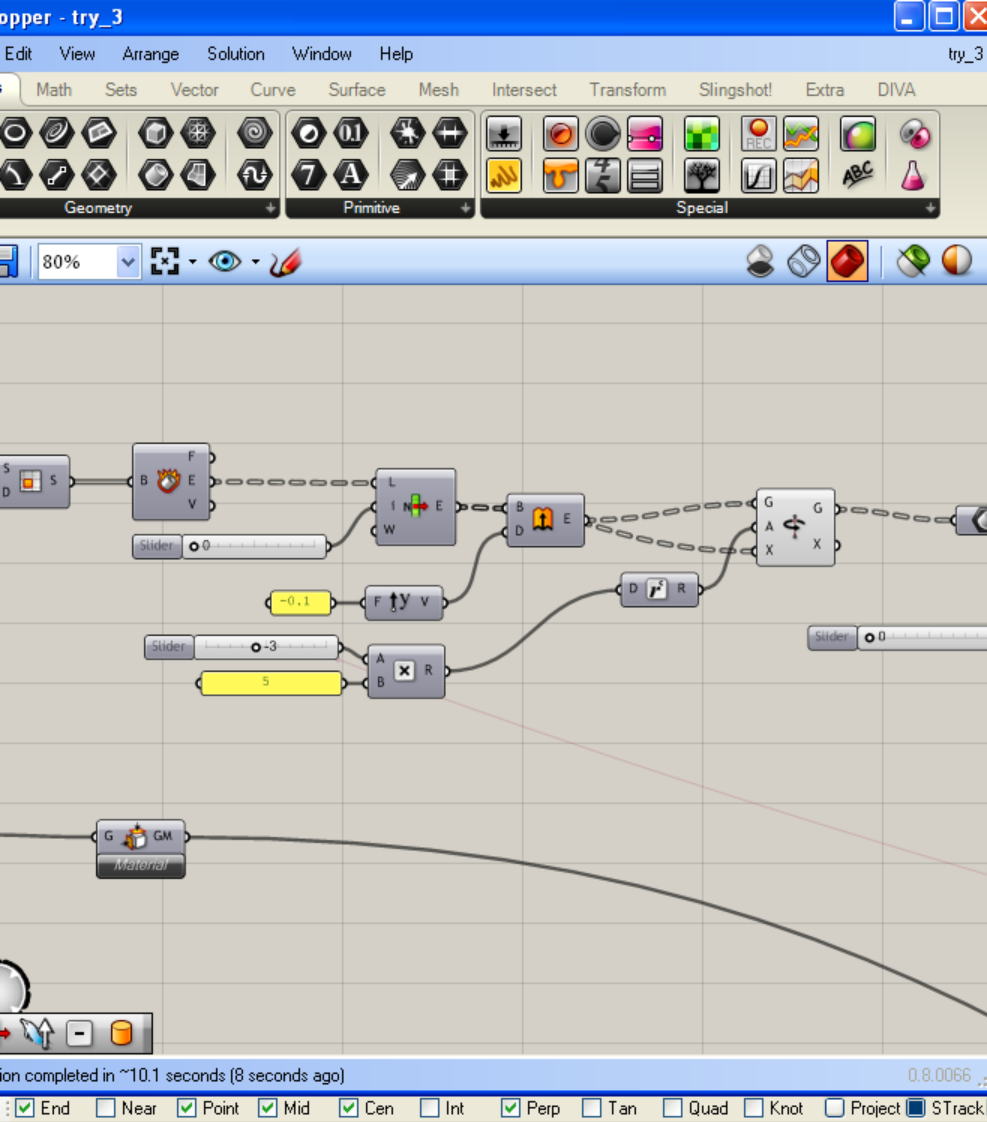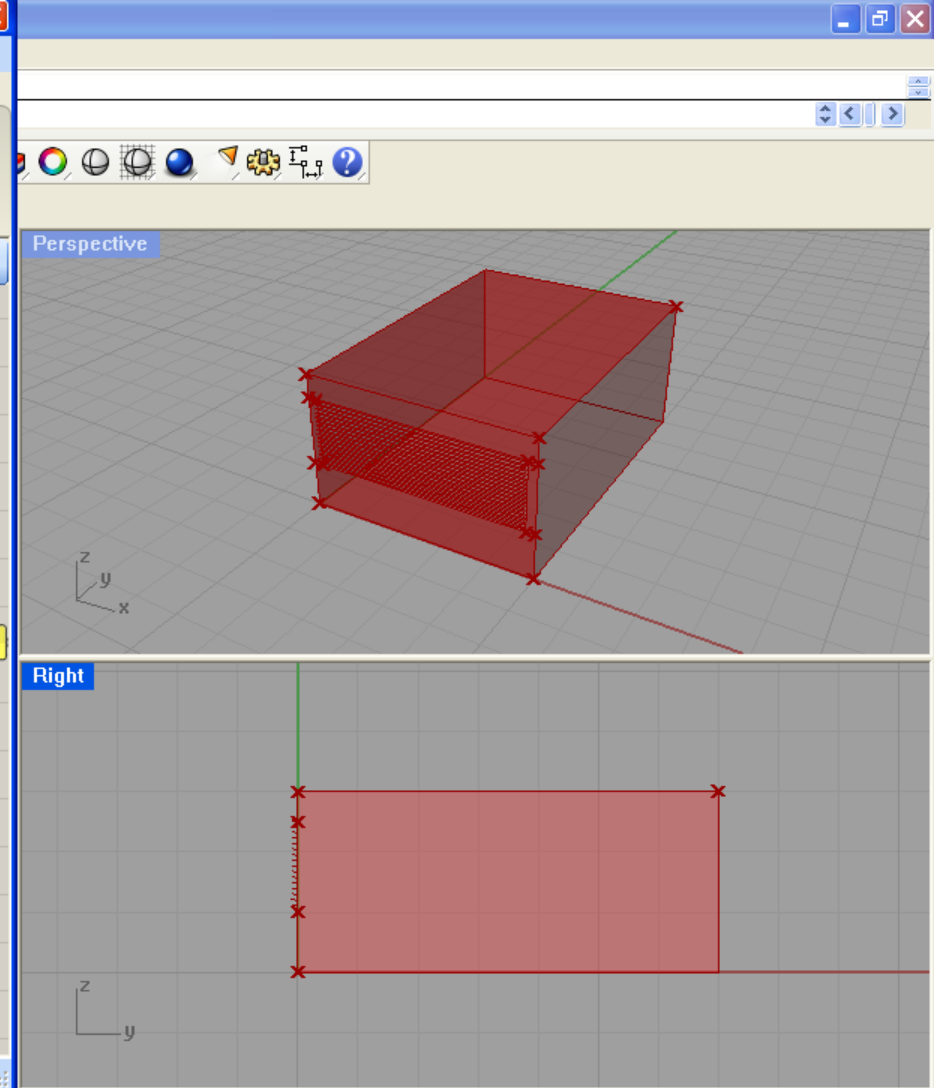# ALTERNATIVE 2.
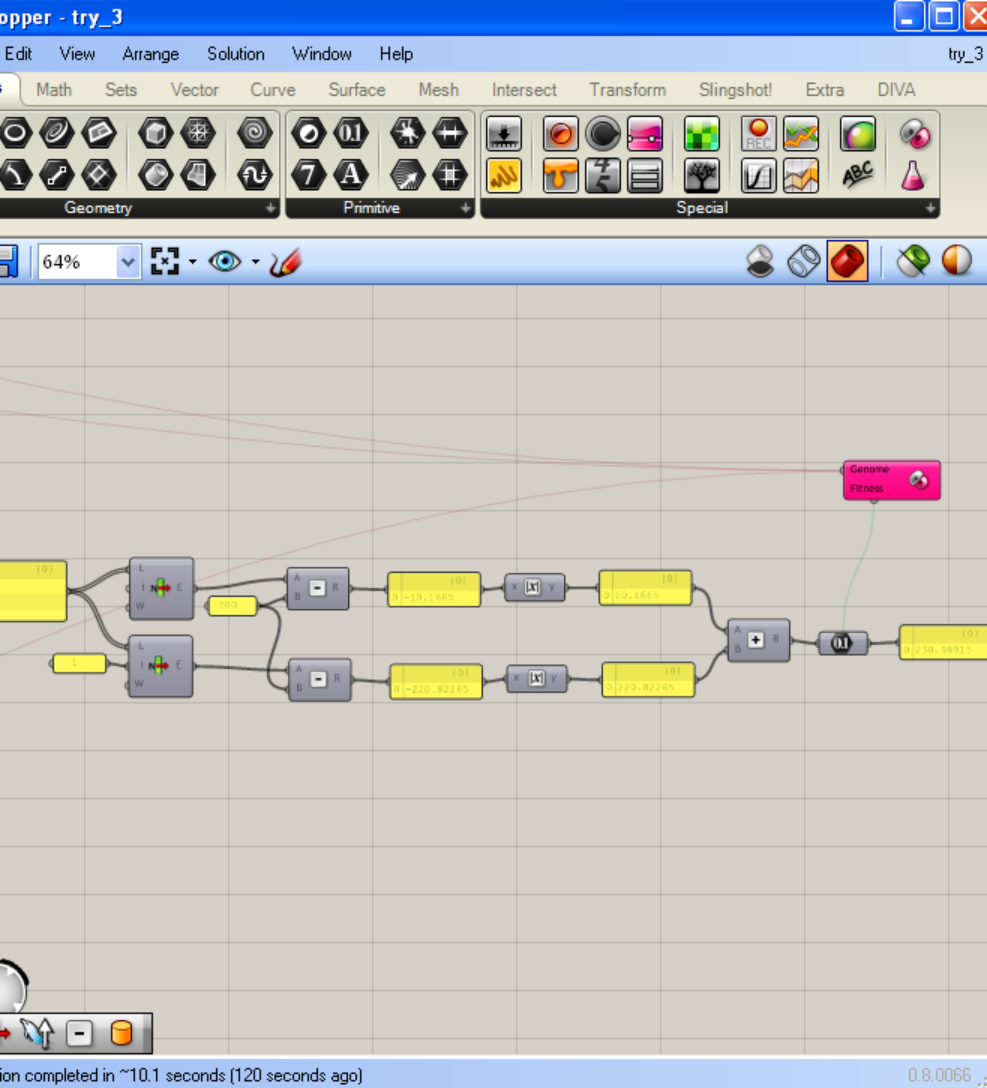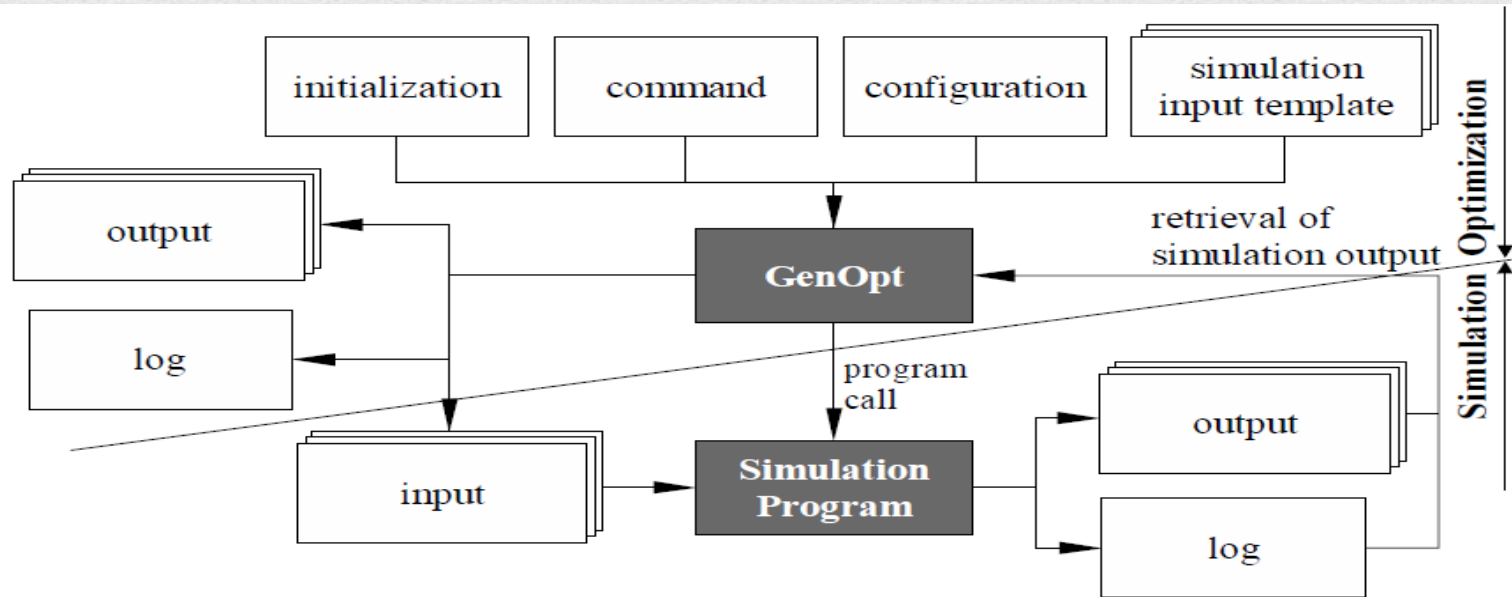
OPTIMIZATION USING GENOPT

- The purpose of GENOPT (GENeral OPTimization) is to enable an engineer to create a user-friendly system of programs for analyzing and/or optimizing anything

Interaction between GenOpt and Radiance

Radiance Files

1

Room_Template.rad

This is the typical .rad geometry file that is used in radiance. The only difference is that file replaces key values that will be used as design parameter with parameter names instead of the actual values

2

Clear_sky_Template.rad

This is the sky definition file that includes sky definition. This file may also contain simulation parameters if need be. So you may be able to rotate the sky

3

pp1_Template.pts

This is the file that contains the point locations to calculate. In our example only two points are used but many more points can be used

4

Run_Template.bat

This is the batch file that is called from within GenOpt with every simulation run

oconv clear_sky.rad room.rad>all.oct
rtrace -faa -ov -h -I -ab 3 -ad 2000 -as 500 -ar 256 -aa 0.1 -ds .1 all.oct<pp1.pts|rcalc -e "$1=179*($1*0.265+$2*0.670+$3*0.065)">results.dat
:rview -vp 2.5 3 1 -vd -1 -1 0 -av .1 .1 .1 all.oct

luminance = 179 * (0.265*R + 0.670*G + 0.065*B)

The Radiance Batch File ①

X	Y	Z	Vx	Vy	Vz
2.50	3.00	0.85	0	0	1
2.50	2.00	0.85	0	0	1

| 2.50 | 3.00 | 0.85001 | 0 | 0 | 1 |
| 2.50 | 2.00 | 0.85002 | 0 | 0 | 1 |

The points file 2

void plastic ceiling_mat
0
0
5 %refl_ceil% %refl_ceil% %refl_ceil% 0 0
void plastic floor_mat
0
05 0.3 0.3 0.3 0 0
void plastic south_wall_mat
0
0
5 0.6 0.6 0.6 0 0
..
..
..

Scen File Template ③

..

..

south_wall_mat polygon wall_S1

0
0
12
5 0 0
0 0 0
0 0 1
5 0 1

..

..

..

If we want to define geometric variables we want to define related variables so that is one changes the other will be calculated during runtime by Radiance, e.g.

Width of wall 1 = x1 + l

Then width of

Width of wall 2 = x2 + l

Scen File Template ③

```
#
# A set of grey venetian blinds
#
void plastic blind_grey
0
0
5        %refl_blinds%        %refl_blinds%        %refl_blinds%        0.02        0
!genblinds blind_grey blind 0.1 4.6 1.5 16 %angle_blinds% | xform -rz -90 -t 0.2 -0.1 1
..
..
..
```

Scen File Template 3

gensky -ang 70 0 +s
Ground ambient level: 20.4

void light solar
0
0
3 7.14e+006 7.14e+006 7.14e+006

solar source sun
0
0
4 0.000000 -0.342020 0.939693 0.5

…

Sky Definition Template ④

GenOpt Files

1

optWinXP.ini

This is the main file GenOpt and tells its where to find the actual project files, the results and the location of the objective function in the results.
Also some post processing can be done here in the results

2

RadianceWinXP.cfg

This file is a small file that usually does not change with different projects.
The only use of this file is to tell GenOpt how to call the program from the command line and the correct syntax for that.

3

Command.txt

This file has two main functions: 1. to tell GenOpt which parameters to change and how to change them (i.e. ranges, steps).
2. To specify the optimization algorithm to be used and its parameters

This file has following structure:

- Simulation
  - Files
    - Template
    - Input
    - Log
    - OutputConfiguration
  - CallParameter
  - Objective Function Location

- Optimization
  - Files
    - Command

Initialization file **1**

Simulation {

```
Files {
   Template {
File1 = room_Template.RAD;
        File2 = clear_sky_Template.rad;
        File3 = pp1_Template.pts;
        File4 = run_Template.bat;

   }
```

Initialization File ①

```
Input {
    File1 = room.RAD;
        SavePath1 = Save;
        File2 = clear_sky.rad;
SavePath2 = Save;
        File3 = pp1.pts;
SavePath3 = Save;
        File4 = run.bat;
SavePath4 = Save;
}
```

Initialization File ①

```
Output {
    File1 = delimitedResults.dat;
    SavePath1 = Save;
  }
  Configuration {
    File1 = "RadianceWinXP.cfg";
}
```

```
CallParameter { // optional section
    // The weather file without extension
    //Suffix = Cairo_IWEC;
  }
```

Initialization File ①

```
ObjectiveFunctionLocation
{
        Name1     = fitness;
        Function1 = "add( %absPoi1% ,
%absPoi2% )";


        Name2     = absPoi2;
        Function2 = "abs( %Poi2-500% )";


        Name3     = absPoi1;
        Function3 = "abs( %Poi1-500% )";


        Name4     = Poi2-500;
        Function4 = "add( %Poi2% , -500 )";


        Name5     = Poi1-500;
        Function5 = "add( %Poi1% , -500 )";


        Name6     = Poi2;
        Delimiter6 = "001";


        Name7     = Poi1;
        Delimiter7 = "002";

}
```

This section can be easily generated by a script file if you have many sensor points We developed a simple excel sheet that outputs this section automatically by clicking and dragging

The correct way of doing this is look at the delimiter that was inserted in the batch file

1

} // End Simulation section

Optimization {

```
Files {
    Command {
      File1 = command.txt;
    }
} // end of configuration file
```

} End Optimization

Initialization File 1

```
Vary{

Parameter{

Name    = refl_ceil;
Min     = 0;
Ini     = 0.5;
Max     = 1;
Step    = 0.2;
  }
..

..
```

This part defines the parameters and how they vary

Command File ② 2

```
OptimizationSettings{
  MaxIte = 2500;
  MaxEqualResults = 1000;
  WriteStepNumber = false;
  UnitsOfExecution = 0;
}


Algorithm{ Main =
GeneticAlgorithmJGAP;
MaxGenerations = 20; // 0 <
maxGenerations

..

..
```

This part defines the parameters of the simulation

Command File

2

SimulationStart {

Command = "cmd /x /c run.bat
%Simulation.Files.Input.File1%
%Simulation.CallParameter.Suffix%\"";
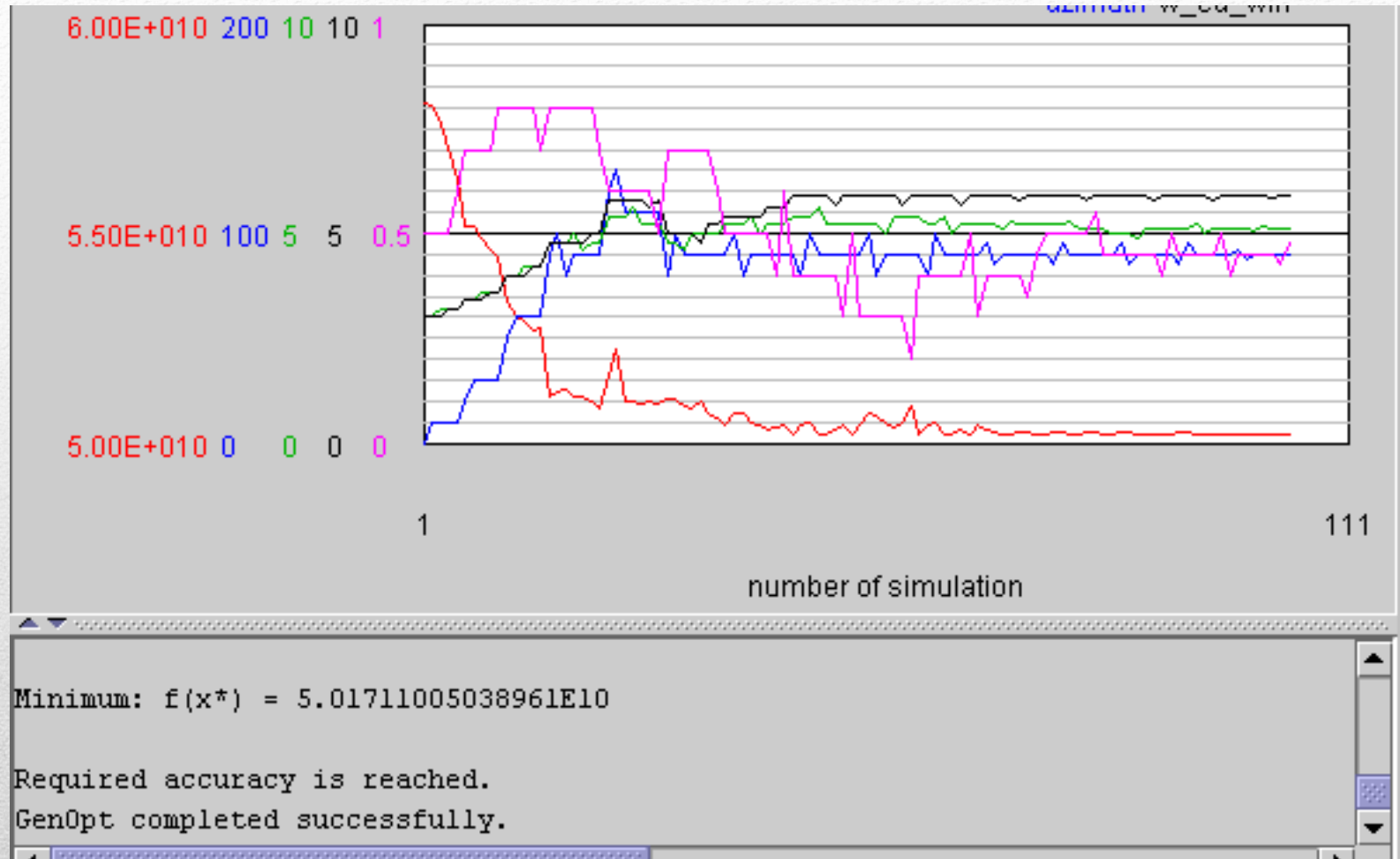
..

..

}

If you wanted access to any of the files you defined in the initialization file you can access them here

Radiance win Config ③

Results

Thank You