

THE COUPLING OF ESP-R AND GENOPT: A SIMPLE CASE STUDY

Leen Peeters^{1,2}, Michael Wetter³, Alex Ferguson⁴ and William D'haeseleer¹

¹Applied Mechanics and Energy Conversion, University of Leuven (KUL), Leuven, Belgium

²Current address: Mechanical Engineering, University of Brussels (VUB), Brussels, Belgium

³Lawrence Berkeley National Laboratory, Environmental Energy Technologies Division, Building Technologies Department, Simulation Research Group, Berkeley, California

⁴CanmetENERGY, National Resources Canada, Ottawa, Canada

ABSTRACT

This paper describes and demonstrates how to use the optimization program GenOpt with the building energy simulation program ESP-r. GenOpt, a generic optimization program, minimises an objective function that is evaluated by an external simulation program. It has been developed for optimization problems that are computationally expensive and that may have non-smooth objective functions. ESP-r is a research oriented building simulation program that is well validated and has been used to conduct various building energy analysis studies. In this paper, the necessary file preparations are described and a simple optimization example is presented.

INTRODUCTION

Optimization is a technique that allows finding the optimal values of parameters that minimise an objective function such as energy use or operating costs. In building energy analysis, objective functions are generally discontinuous in the design parameters (Wetter and Polak, 2003), and their evaluation is usually undertaken by a dedicated simulation program. In these applications, the derivatives of an objective function are generally unavailable and might not even exist. While gradient-based optimization methods are not well-suited for these applications, derivative-free optimization algorithms usually perform well on such problems. GenOpt (Wetter 2009a) is an optimization program that contains libraries of such derivative-free optimization algorithms and that can be used for problems in which the cost function is evaluated using a simulation program that is external to the optimization program. GenOpt has been used to solve various building energy optimization problems. It has been coupled to different simulation programs, including TRNSYS (Thornton et al. 1997), IDA-ICE (Hassan et al. 2008), EnergyPlus (Wetter 2000) and Dymola (Wetter 2009b). While EnergyPlus and TRNSYS are commonly used in building optimization studies, there is precedence for employing ESP-r as well. Ferguson et al. (2008) used GenOpt to calibrate ESP-r models of combined heat and power units. However, extended optimization studies using ESP-r have not been

reported in the literature. Consequently, no guidelines are currently available for performing optimization studies with ESP-r. To address this gap, this paper discusses coupling GenOpt and ESP-r in detail. A simple example is given, for which the necessary files are downloadable from the ESP-r public repository (https://espr.svn.cvsdude.com/extras/examples/Optimization_with_GenOpt/).

SETTING UP AN OPTIMIZATION PROBLEM

The problem description

Solving an optimization problem requires modifying a set of independent variables to find a minimum of a function, which is called the objective function. Independent parameters may include, for example, insulation thickness, window sizes, equipment sizes and control set points. These user-selected parameters may vary continuously or in discrete steps. For practical reasons, these parameters may be constrained such that they vary over a limited range. Constraints may be handled directly by the optimization algorithms, such as for the simple case of lower and upper bounds of independent parameters. More complex constraints may be implemented by adding a penalty function to the objective function. For example, an optimization problem may require energy use to be minimized, while using a penalty function to ensure the resulting solution maintains thermal comfort. The set of independent parameters that results in a minimum value of the objective function is called a minimizer. It can be either a local or a global minimizer. For problems that are typically encountered in building energy optimization, it is in general not possible to know whether the cost function is convex. Thus, without extensive parametric studies, which are often computationally too expensive to conduct, it is not possible to know whether the global minimum has been found. Nevertheless, a local minimum is generally better compared to the non-optimized case.

Selecting the optimization algorithm

Several optimization algorithms are currently available in GenOpt's library. In addition, users can implement

their own optimization algorithms. Wetter (2009a) provides a discussion of different optimization algorithms and suggestions for selecting an appropriate algorithm. Here, only a brief overview is given of two families of algorithms that are implemented in GenOpt: the Particle Swarm Optimization algorithms (PSO algorithms) and the Generalized Pattern Search algorithms (GPS algorithms). These algorithms are most relevant for solving the example described here.

The PSO algorithms can be used for continuous and discrete independent parameters. PSO algorithms randomly generate a set of initial points to be evaluated. Each point is called a particle, and the set of points is called a population. The next populations are computed using a particle update equation. This equation is modelled based on the social behaviour of bird flocks or fish schools (Wetter 2009a). The particle update equation attracts particles towards the best known iterate in a way that moves the whole population towards regions where the objective function is expected to be decreasing. It also contains a term that leads to a global exploration of the search space in order to increase the chance of finding a global minimum.

The GPS algorithms can be used for problems with continuous parameters. GPS algorithms divide the search space into a mesh and search on this mesh for a decrease in the objective function. If no further decrease can be found, the mesh is refined and the search is repeated, starting from the currently best known iterate. This process ensures convergence to a

local minimum under appropriate smoothness conditions (Torczon, 1997).

In general, the PSO algorithms are well-suited for exploring large parts of the parameter space and typically come close to an optimal solution. But because they are inherently stochastic, they require many iterations to refine the solution, which can lead to a long computing time. Although GPS algorithms can contain a global search component, they are typically used to explore small regions of the parameter space. GPS algorithms determine the parameter values for the next iteration based on the local descent around the current iteration. This allows the algorithms to quickly refine a solution that has been identified by a PSO algorithm. By combining these algorithms, the entire parameter space is explored in order to increase the chance of finding the global minimum. The hybrid GPS/PSO algorithm is applicable for optimization problems with continuous parameters only, or mixed continuous and discrete parameters. It starts with a global search using the PSO algorithm in order to increase the chance of finding the global minimum. After a user-specified number of generations, the algorithm switches to a local search using the GPS algorithm to further refine the search. The GPS starts with the best iterate found by the PSO algorithm. As the GPS algorithm requires the independent parameters to be continuous, only the continuous independent parameter will be varied during the second phase of the hybrid algorithm.

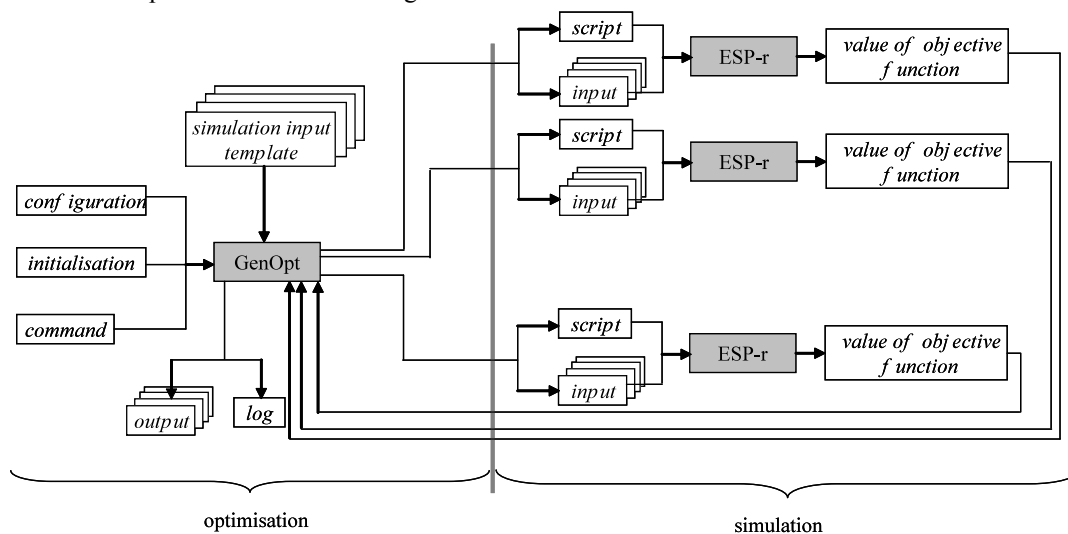


Figure 1 Data flow of GenOpt-ESP-r coupling

Preparing the files

Different files need to be prepared in order to set up the communication between GenOpt and the building energy simulation program, here ESP-r, as shown in Figure 1. An extended overview on the details concerning the file preparations can be found in Wetter (2009a). The overview here is limited to a concise description.

```
SimulationStart
{
  Command = "./script.pn" ;
  WriteInputFileExtension = false;
}
```

Figure 2 Extract of configuration file pointing to the script

```
Simulation{
// Files used during optimization
Files{
  // Name of file containing input template
  Template{
    File1 = "office_template.geo";
    File2 = "reception_template.geo";
  }
  // Name of target file
  Input{
    File1 = "office.geo";
    File2 = "reception.geo";
  }
}
```

Figure 3 Extract of initialization file pointing to different files containing independent parameters

A second file, the *initialization file*, contains the names and paths of all files involved in or affected by the optimization. Figure 3 shows part of an initialization file. In this case, the templates for two ESP-r geometry files are named in the *Template{}* section. During each iteration, GenOpt will modify the data in these files according to the selected optimization algorithm, and will write new files using the names given in the *Input{}* section. These new files will be parsed by ESP-r at the start of the simulation. The ‘//’ characters denote comments in GenOpt input files; these characters and the text that follows them are ignored by GenOpt. The *command file* defines the feasible set, initial values and scaling information for all independent parameters (Figure 4). The file further contains criteria for convergence and specific parameters related to the selected optimization algorithm. In this case, ‘offz1’ and ‘offz2’ define the dimensions of a window. The variable *off_z1* varies continuously between 0.2 and 1.0 m. The value *Step*=0.05 causes the algorithm to search on an initial mesh with a spacing of 0.05 m, but this spacing will be

refined during the course of the optimization. The variable *off_z2* varies in discrete steps between 1.3 m and 2.2 m. We could have changed both values using a continuous parameter, but for illustration, we varied one parameter continuously and the other discretely.

```
Vary{
  Parameter{ // off_z1
    Name = off_z1;
    Min = 0.2;
    Ini = 0.5;
    Max = 1.0;
    Step = 0.05;
  }
  Parameter{ // off_z2
    Name = off_z2;
    Ini = 4;
    Values = "1.3, 1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2.0, 2.1, 2.2";
    Type = SET;
  }
}
...
Algorithm{
  Main = GPSPSOCCHJ;
}
...
```

Figure 4 extract of command file with variation range for a continuous and discrete parameter and indication of selected algorithm (in this case a combination of GPS and PSO)

The *simulation input template files* are the ESP-r input files in which all independent parameters have been replaced by the user with a variable name that is enclosed by percentage characters. Before evaluating the objective function, GenOpt will make a copy of this file and replace in the copied file all variable names with their numerical value as determined by the optimization algorithm. Figure 5 presents an extract of a simulation template file describing building geometry. In this case, the ESP-r keyword **vertex* indicates that the subsequent data describe a coordinate point in the ESP-r building geometry. The ‘#’ character denotes a comment in ESP-r input files; this character and the text that follows it are ignored by ESP-r. In this example, vertices 13–16 denote the coordinates of the four points defining a window. The keywords ‘%off_z1%’ and ‘%off_z2%’ indicate the locations where GenOpt will save the parameter values computed by the optimization algorithm. For instance, GenOpt will replace the ‘%off_z1%’ keyword with the value of the *off_z1* variable. Thus, GenOpt can manipulate complex components of a building simulation program’s inputs without understanding its underlying data model. This methodology ensures that GenOpt’s algorithms can be applied to any aspect of a

building simulation model, and that it is compatible with any building simulation program that uses ASCII input files. Once GenOpt has prepared the new ESP-r input files, it invokes the script `script.py` to start the building simulation. This script calls ESP-r and post-processes the ESP-r results. It also writes a file that contains the new value of the objective function and it writes additional output files that store more information on the simulation runs. GenOpt will then read the objective function value from the output file. Next, it displays the results to the user and generates files that document the progress of the optimization. A more detailed discussion about the data post-processing can be found in Peeters (2009).

```
*vertex,5.00000,6.00000,2.50000 # Vertex 12
*vertex,1.00000,8.00000,%off_z1% # Vertex 13
*vertex,1.00000,6.00000,%off_z1% # Vertex 14
*vertex,1.00000,6.00000,%off_z2% # Vertex 15
*vertex,1.00000,8.00000,%off_z2% # Vertex 16
```

Figure 5 Extract of simulation template file, in this case the file called `office_template.geo`

Running the optimization

The optimization can be run either with a graphical interface, or as a console application. The graphical interface shows the optimization progress of the objective function and of user-selectable parameters. The console application allows running GenOpt as a batch job for several sequential optimizations or starting GenOpt over a remote connection. On computers with multiple processors, GenOpt may run multiple ESP-r simulations in parallel. How many simulations are run in parallel depends on the number of processors, on the optimization algorithm and its configuration.

EXAMPLE

The simulated case

Consider the optimization of the window size of a 3-zone heavyweight office building depicted in Figure 6. The geometry of all 3 windows is to be optimized. The coordinates related to the bottom side of the window are continuous, those determining the upper side are discrete. All parameters can vary over a limited interval covering the lower half of the height of the wall they are built in for the bottom side and the upper half of the height of that wall for the upper side of the window. The resulting constraining minimum and maximum window sizes are listed in Table 1.

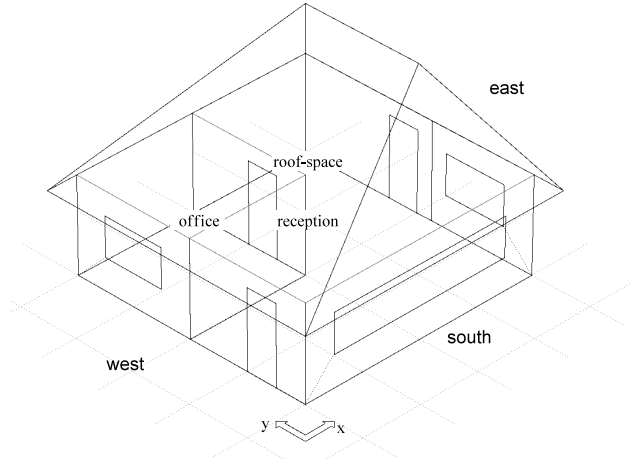


Figure 6 The 3-zone building with standard windows, south-east view

Table 1 Minimum and maximum window sizes

WINDOW	MINIMUM SIZE	MAXIMUM SIZE
East	0.6 m ²	4.0 m ²
South	1.8 m ²	12 m ²
West	0.6 m ²	4.0 m ²

The aim is to minimise the energy consumption for heating, cooling and artificial lighting. The objective function $f(x)$ (kWh) is thus defined as follows:

$$f(x) = \frac{Q_h(x)}{\eta_h} + \frac{Q_c(x)}{\eta_c} + PE \cdot EI(x)$$

where x are the independent parameters, $Q_h(\cdot)$ and $Q_c(\cdot)$ are annual heating and cooling demand (kWh) and $EI(\cdot)$ is the electricity consumption (kWh) for artificial lighting. The plant efficiencies related to heating and cooling, η_h and η_c , are set to 0.44 and 0.77 respectively (Huang 1999). PE is the conversion factor to convert site electricity to primary energy and is dependent on the country. For Belgium it is 2.5 (EURPROG 2007), for Italy 2.17 (EURPROG 2007) and for Canada 1.84 (NRCAN 2006). The optimization is performed for the same building located in three different climate zones; Brussels (Belgium), Montreal (Canada) and Palermo (Italy). Typical weather data is summarized in the graphs below. For all cities, Figure 7 shows the hours the outdoor temperature is within certain limits. Figure 8 gives for each of those temperature intervals the average total solar radiation (Wh/m²). An intermittent heating/cooling control is implemented.

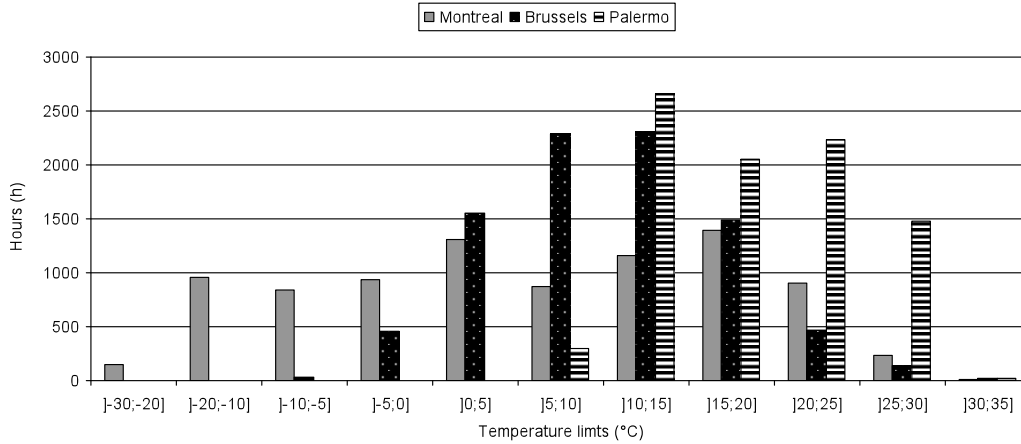


Figure 7 Yearly temperature frequency (h)

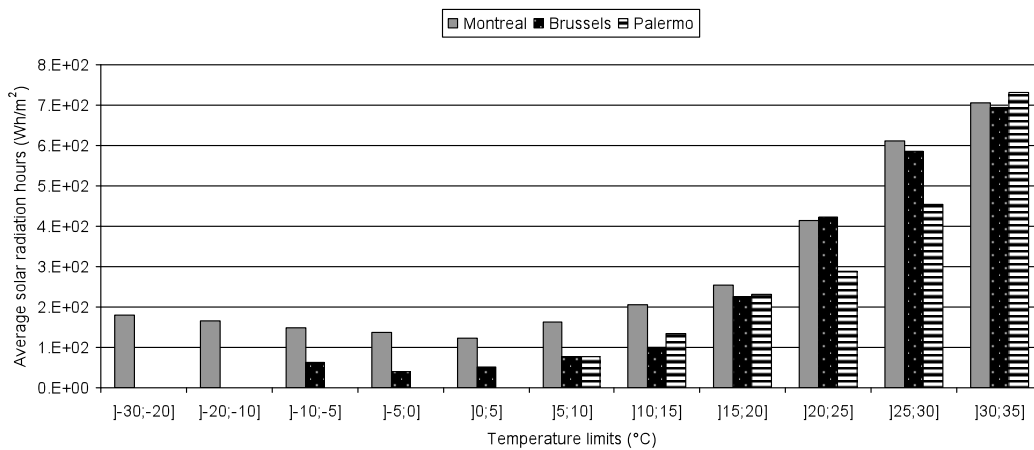


Figure 8 Average solar radiation hours (Wh/m²) for the different temperature intervals

During office-hours, the heating or cooling is delivered to the different zones by use of ESP-r's ideal zone controller. The temperature set point is 20°C for heating and 26°C for cooling. Outside this period, heating and cooling setpoints are 15°C and 28°C respectively. The maximum heating and cooling power is 3 kW. Artificial lighting is turned on when the light intensity (daylight as well as artificial lighting) is below 450 lux for the reception and 600 lux for the office. The solar heat gain coefficient is 0.61. The U-values of the heavyweight building envelope are given in Table 2. The annual simulation is performed using a 6-minute time step. Indoor heat gains are limited to office hours and are highest in the reception.

The optimization

The parameters involved are mixed discrete and continuous. Therefore, both the PSO and the combination of PSO and GPS could be applied. In this example, the latter option has been selected. The different files, including the script, can be downloaded from the ESP-r branch <https://espr.svn.cvsdude.com/extras/examples/Optimiz>

ation_with_GenOpt/. The posted files there are for a simplified optimization with shorter computing time.

Table 2 U-values of different parts composing the building envelope.

Part of building envelope	U-value (W/m²K)
External wall	0.5
Ground floor	0.5
Roof	1.1
Window	1.24
Door	1.5

Results analysis

The optimizations performed for the current example are for a whole year and the savings achieved are thus for both heating and cooling. Compared to the initial case values as given in Table 3, the optimized window sizes resulted in savings of 8.5% for Brussels, 4.8% for Montreal and even 21.3% for Palermo. The convergence process is shown in Figures 9 to 12. Figure 9 shows the value of the objective function, i.e. the primary energy delivered, as a function of the

iteration number throughout the optimization process. Figures 10 to 12 show the corresponding dimensions of the windows. To display the results, we did not plot the individual window heights and widths, as specified in Figure 4. Instead, we plotted the resulting window areas normalized to the maximum window area as listed in Table 1. The points that are spread out on the left of the Figures 10 to 12 are iterates of the PSO algorithm. The points on the right part of the figures that appear to follow a more smooth line are the main iterates of the GPS, here the Hooke-Jeeves, algorithm.

Considering the Brussels case, the optimum for the east window shows to be as large as possible. This is the result of the best trade-off between heat gains and losses and lighting electricity savings due to use of daylight. The weight of the latter consumption is relatively high due to the conversion factor of 2.5 for the Belgian electricity generation mix. However, as can be seen in Table 4 below, the overall energy use for lighting is small compared to the energy use for cooling. The optimal size of the south window tends towards minimal, the main reason being the higher

intensity of the solar radiation on this window. This high solar intensity increases the cooling energy consumption. This increase cannot be compensated by a reduction in energy for artificial lighting and a possible decrease of the heating load due to solar heat gains in winter. For Montreal, all optimal window areas are smaller than the initial values (Figure 11). The values given by Table 4 show a large reduction in the cooling energy consumption when compared to the initial case scenario. The reduction of the window sizes has only a limited effect on the heating energy consumption, i.e. an increase by 6.4%, while the increase in primary energy consumption for lighting is close to 87%. The absolute increase for lighting energy, however, is small compared to the primary energy for heating and cooling. The results are thus mainly explained by a reduction in solar gains entering the building and thus a way to decrease overheating. Palermo shows a markedly different climate. Figure 7 shows moderate to high temperatures throughout the whole year for Palermo. Figure 8 shows that the solar radiation during moderate to high outdoor

Table 3 Initial and optimized parameter values

Parameter	Initial value			Optimized value		
	Brussels	Montreal	Palermo	Brussels	Montreal	Palermo
Lower height of east window (m)	0.5	0.5	0.5	0.2	1.0	0.8
Upper height of east window (m)	1.6	1.6	1.6	2.2	1.3	2.2
Lower height of south window (m)	0.5	0.5	0.5	0.95	0.8	0.2
Upper height of south window (m)	1.6	1.6	1.6	1.3	1.6	1.3
Lower height of west window (m)	0.5	0.5	0.5	0.7	0.7	0.8
Upper height of west window (m)	1.6	1.6	1.6	1.3	1.3	1.3

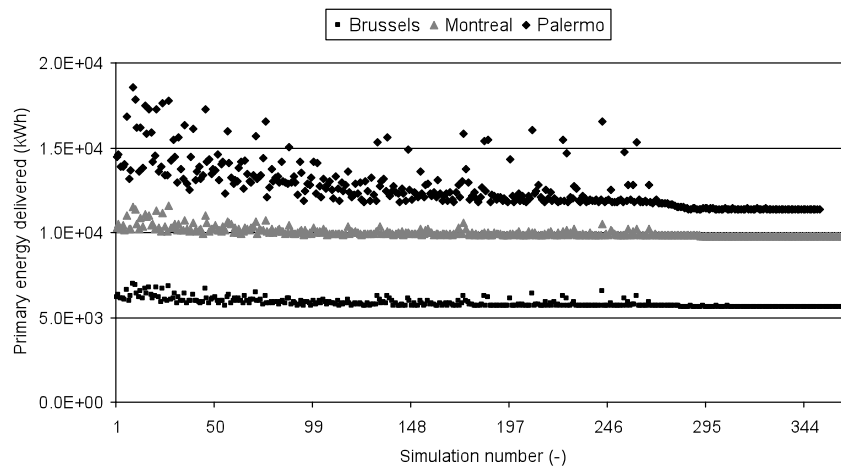


Figure 9 Primary energy delivered as a function of the simulation number in the optimization process. Results are shown for the buildings located in Brussels, Montreal and Palermo

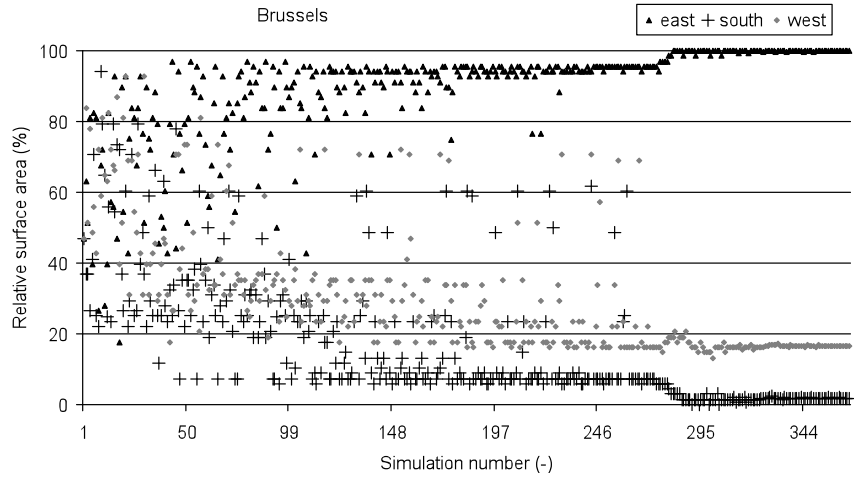


Figure 10 Relative surface area as a function of the simulation number in the optimization process. Results given for the building located in Brussels

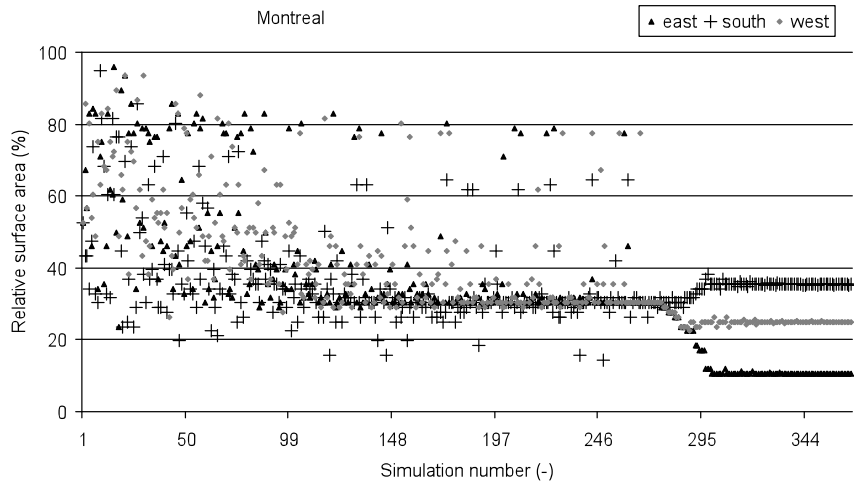


Figure 11 Relative surface area as a function of the simulation number in the optimization process. Results given for the building located in Montreal

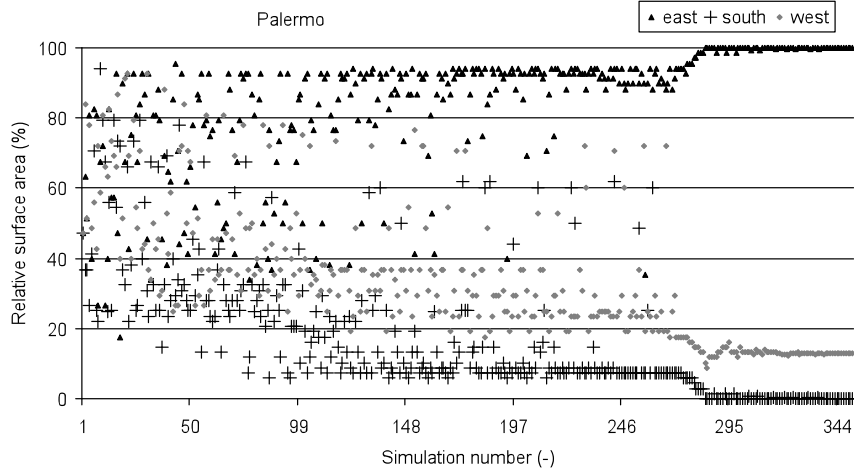


Figure 12 Relative surface area as a function of the simulation number in the optimization process. Results given for the building located in Palermo

Table 4 Initial case and optimized case primary energy uses

Load	Initial case			Optimized case		
	Brussels	Montreal	Palermo	Brussels	Montreal	Palermo
Heating primary energy use (kWh)	1131	3806	1	1226	4051	4
Cooling primary energy use (kWh)	4136	5763	14128	3145	4423	10398
Lighting primary energy use (kWh)	889	699	344	1268	1304	738
Total primary energy use (kWh)	6156	10269	14473	5639	9777	11140

temperatures is less than for the same temperatures in Montreal. This ratio changes only for the seldom occurring temperatures above 30°C. As shown by Figure 9 and Table 4, the energy use for Palermo is higher than for the other two cases and it is dominated by the cooling energy consumption. As shown by Figure 12, both the south and west window have their optimum close to the minimal sizes, while the optimum surface area for the east window is maximum. The results are thus again explained by a reduction in solar gains. Compared to the Montreal case, however, there is a beneficial effect of solar radiation on the east window. This is due to a less intense solar radiation, negligible heat losses due to higher ambient temperatures, combined with the higher weight of electricity for artificial lighting: a conversion factor of 2.17 for Palermo, compared to 1.84 for Montreal.

The determination of the primary reasons for the observed phenomena would require a thorough sensitivity analysis, which is outside the scope of this study.

CONCLUSION

GenOpt has been used frequently with common building simulation programs such as TRNSYS, EnergyPlus and IDA-ICE. Running an ESP-r - GenOpt optimization has not been previously described in detail. The overview given in this paper shows the flow of information during such an optimization and describes the files to be prepared.

While simple, the glazing optimization presented in this paper shows the strengths of the coupling between the two programs. Since the files required to run the example are available in the public ESP-r repository, future users can build their own optimization files in an efficient way.

ACKNOWLEDGEMENT

This research was supported by the Assistant Secretary for Energy Efficiency and Renewable Energy, Office of Building Technologies of the U.S. Department of Energy, under Contract No. DE-AC02-05CH11231.

REFERENCES

EURPROG Network of Experts, 2007. Statistics and prospects for the European electricity sector.

- Ferguson, A., et al., 2008. Modelling Residential-scale, Combustion-based Cogeneration in Building Simulation. *Journal of Building Performance Simulation*. 2 (1), pp. 1-14.
- Hassan, A., Vuolle, M., and Siren, K., 2008. Minimisation of life cycle cost of a detached house using combined simulation and optimization, *Building and Environment*, 43 (12), pp.2022-2034
- Huang J., Franconi E., 1999. Commercial heating and cooling loads component analysis. Technical report LBL-37208, Lawrence Berkeley National Laboratory, EETD.
- NRCAN, 2006. Electricity Generation Sector Canada, through <http://nrcan.gc.ca> (accessed on 2009-11)
- Peeters, L., 2009. Water-based heating/cooling in residential buildings; towards optimal heat emission/absorption elements, PhD dissertation, University of Leuven (K.U.Leuven), Belgium
- Thornton, J., et al., 1997. Residential vertical geothermal heat pump system models: calibration to data, ASHRAE summer meeting
- Torczon V., 1997. On the convergence of pattern search algorithms. *SIAM Journal on Optimization*, 7(1):1-25.
- Wetter, M., 2000. Design optimization with GenOpt. *Building Energy Simulation, User News*, Lawrence Berkeley National Laboratory, vol. 21
- Wetter, M., 2009a. GenOpt, a generic optimization program, Version 3.0.0 User Manual 3.0.0, Technical Report LBNL-2077E, Berkeley, CA.
- Wetter, M., 2009b. Modelica-based Modeling and Simulation to Support Research and Development in Building Energy and Control Systems. *Journal of Building Performance Simulation*, 2(2), pp. 143-161.
- Wetter, M. and Polak, E., 2003. A convergent optimization method using pattern search algorithms with adaptive precision simulation. 8th IBPSA conference, Eindhoven, The Netherlands