RESEARCH ARTICLE

A comparison of global optimisation algorithms with standard benchmark functions and real-world applications using EnergyPlus

Jérôme Henri Kämpf[a]*, Michael Wetter[b] and Darren Robinson[a]

[a] *Solar Energy and Building Physics Laboratory, Ecole Polytechnique Fédérale de Lausanne, Station 18, 1015 Lausanne, Switzerland*

[b] *Simulation Research Group, Environmental Energy Technologies Division, Lawrence Berkeley National Laboratory, Berkeley, California, United States of America*

There is an increasing interest in the use of computer algorithms to identify combinations of parameters which optimise the energy performance of buildings. For such problems, the objective function can be multi-modal and needs to be approximated numerically using building energy simulation programs. As these programs contain iterative solution algorithms, they introduce discontinuities in the numerical approximation to the objective function. Metaheuristics often work well for such problems, but their convergence to a global optimum cannot be established formally. Moreover, different algorithms tend to be suited to particular classes of optimization problems.
To shed light on this issue we compared the performance of two metaheuristics, the hybrid CMA-ES/HDE and the hybrid PSO/HJ, in minimizing standard benchmark functions and real-world building energy optimization problems of varying complexity. From this we find that the CMA-ES/HDE performs well on more complex objective functions, but that the PSO/HJ more consistently identifies the global minimum for simpler objective functions. Both identified similar values in the objective functions arising from energy simulations, but with different combinations of model parameters. This may suggest that the objective function is multi-modal. The algorithms also correctly identified some non-intuitive parameter combinations that were caused by a simplified control sequence of the building energy system that does not represent actual practice, further reinforcing their utility.

Keywords: optimisation; algorithm; application using EnergyPlus; Covariance Matrix Adaptation Evolution Strategy Algorithm (CMA-ES) and Hybrid Differential Evolution (HDE); Particle Swarm Optimisation (PSO) and Hooke-Jeeves (HJ); building energy minimisation

1. Introduction

Detailed simulation programs are increasingly used to asses the energy performance of buildings. In Switzerland, the law requires every new construction or refurbishment to meet energy consumption standards such as SIA 380.1, Minergie or Minergie P. To meet the charge of the U.S. Energy Independence and Security Act of 2007, the "Zero-Net-Energy Commercial Buildings Initiative" of the U.S. Department of Energy has as its goal that any commercial building newly constructed in the United States from 2030 will be a Zero-Net-Energy building. Similarly, the Integrated Energy Policy Report 2007 of the California Energy Commission recommends that efficiency standards for buildings be increased so that, when combined with on-site generation, newly constructed buildings can be Zero-Net-Energy by 2020 for residences and by 2030 for commercial buildings. These performance standards along with the raising energy costs are beginning to stimulate increased demand for the development and usage of building simulations

*Corresponding author. Email: jerome.kaempf@epfl.ch

programs such as ESP-r (Clarke, 2001), TRNSYS (Bradley & Kummert, 2005) and EnergyPlus (ENE, 2009). When it comes to the energy performance optimisation of buildings, the end-users of the programs may experience difficulties in identifying the best performing solution to a given problem as in general the design parameters are numerous and the final response is generally non-linear and non-convex. Moreover building simulation programs contain numerical solvers that may iterate until a convergence criterion is met, resulting in the energy function being discontinuous with respect to the input parameters (see for example Wetter & Polak (2004)). To tackle such energy optimisation problems, metaheuristics are of interest, as they do not rely on the smoothness of the function. Moreover, metaheuristics can handle non-convex functions and black-box optimisation problems. While convergence to a global optimum can often not be formally proven for such algorithms, they usually produce a good solution. Many metaheuristics are in use today and some experiments are generally required to discover which algorithm performs best on a selected family of problems.
Caldas & Norford (2002) have used Genetic Algorithms (GAs) to look for optimized design solutions in terms of thermal and lighting performance in a building. His study addressed the placing and sizing of windows in a simple office building. The evaluation criteria (or fitness function) was annual energy spent for heating, cooling and lighting. The author outlined that different runs of the optimisation tool may lead to different solutions with similar performance, and this to be an advantage for the designer to make a choice between available alternatives. Wright et al. (2002) applied multi-objective optimisation using GAs (MOGA) for the identification of the pay-off characteristic or Pareto front between the energy cost of a building and the occupant thermal discomfort. Wetter & Wright (2004) compared the performance of deterministic and probabilistic optimisation algorithms in minimizing the building energy consumption (lighting, fan, cooling and heating). The authors concluded that depending on the smoothness of the objective function, one algorithm performs better than another.
The two metaheuristics chosen are recent hybrid algorithms: the Covariance Matrix Adaptation Evolution Strategy and Hybrid Differential Evolution (CMA-ES/HDE) and the hybrid Particle Swarm Optimisation and Hooke-Jeeves (PSO/HJ). The hybrid PSO/HJ was ranked best amongst nine optimisation algorithms tested with EnergyPlus, using two sets of parameters (Wetter & Wright, 2004). Moreover it is part of a Generic Optimisation Program "GenOpt" that is freely available and customisable (Wetter, 2004). The hybrid CMA-ES/HDE has been used in irradiation maximisation on building envelopes and compared with a multi-objective optimiser (Kämpf & Robinson, 2009). This paper compares the novel CMA-ES/HDE with the established performance of the PSO/HJ on problems with benchmark functions and with EnergyPlus.

2. Methodology

We start by defining the optimisation problem in general, summarising the principles of the two algorithms selected and then describing the parameters used by the algorithms. We continue by explaining the method used for handling constraints in the algorithms and then present a selection of benchmark functions that vary from convex to highly multi-modal. Those functions have an analytical form and are therefore inexpensive to compute, so that we can use statistical measures of the performance of the algorithms. Results from this may give some insights into which algorithm is best suited to a particular response function, and furthermore into the selection of the algorithms parameters that are most adapted to those functions. Finally, we use the two algorithms and their selected parameters on a more computationally expensive real-world problem that involves the minimisation of the annual primary energy consumption of a building. In order to vary the complexity of the corresponding objective function, we defined two different buildings (with one thermal zone and several thermal zones) at three locations representing three different climates. The building performance is simulated with EnergyPlus version 2.2. As our cost function evaluations are computationally expensive and the available computing resources

limited, we compare the algorithms performance for a prescribed number of objective function evaluations.

2.1 Optimisation problems and algorithms

The manual way of identifying the preferred parameter values for a given problem is to decompose the parameter space in discrete values and exhaustively compare the simulated performance for all possible parameter combinations. Unfortunately when the dimensionality of the parameter space becomes large, this method is impractical because the number of evaluations grows exponentially with the number of parameters. To overcome the problem that an exhaustive search is computationally impracticable, more advanced optimisation algorithms have been developed.
Optimisation algorithms search for a minimum[1] (or minima) of a function f that depends on n independent decision variables. Formally, the algorithm searches for:

$$\vec{x}_{min} \in \arg\ \min\{f(\vec{x}) \mid \vec{x} \in M \subseteq \mathbb{R}^n\}, \tag{1}$$

where $n \in \mathbb{N}^\star$ is the dimension of the problem, $f : M \to \mathbb{R}$ is the objective function, $M = \{\vec{x} \in \mathbb{R}^n \mid g_j(\vec{x}) \leq 0, \forall j \in \{1, ..., m\}\}$, $M \neq \emptyset$ is the feasible region and $m \in \mathbb{N}$ is the number of constraints. The set of inequality constraints $g_j : \mathbb{R}^n \to \mathbb{R}, \quad \forall j \in \{1, ..., m\}$ includes a special case of constraints due to the domain boundaries $l_i \leq x_i \leq h_i$, where $l_i, h_i \in \mathbb{R}$ and $i = 1..n$. The symbol l_i refers to the lower bound and h_i to the upper bound of the domain.
The function f is generally non-linear, multi-modal, discontinuous and hence non-differentiable in simulation-based building energy optimisation (see Wetter & Polak (2004)). We therefore selected global search algorithms that do not require smoothness of the objective function, but instead use probabilistic operators to search for an improvement in the objective function. Recall that for such problems, one cannot guarantee that the global optimum will be found with a finite number of simulations.

2.1.1 Hybrid CMA-ES and HDE

The CMA-ES and HDE are population-based Evolutionary Algorithms. Evolutionary Algorithms are inspired by principles of biological evolution theory. Each potential solution ($\vec{x} \in \mathbb{R}^n$) of the optimisation problem is considered as an individual (parent or child) with variables and the corresponding objective function value is considered as its fitness. The algorithm goes through the phases of recombination, mutation and selection of the individuals.
The (μ_I, λ)-CMA-ES starts with a population of μ parents randomly drawn within the domain boundaries. The λ children are created using a global weighted intermediate recombination method in conjunction with the sorted parent population. In the implementation used, μ and λ are adapted to the problem size n according to $\mu = 2 + \lfloor 1.5 \cdot \log(n) \rfloor$ and $\lambda = 4 + \lfloor 3 \cdot \log(n) \rfloor$. The main mechanism of the mutation changes the variables of the children by adding random noise drawn from a normal distribution. An elitist selection is made to keep the μ best individuals amongst the children, which will become the new parents. Finally an adaptation of the mutation parameters is made, taking into account the progress made by the selected individuals. A detailed description about CMA-ES algorithm can be found in Hansen & Ostermeier (2001).
The HDE $(NP, F, Cr, \epsilon_2, \vec{\epsilon_3})$ is a modified form of the original DE by the addition of a migration phase after the selection phase. The population of NP individuals is randomly chosen within the domain boundaries. For each member of the population, a trial individual is generated by the addition of a base vector and a differentiation vector that is scaled by F. The choice of these two vectors come from a strategy that combines individuals of the parent population. A crossover is then carried out between the trial and corresponding parent, with a probability of selecting each parent allele equal to Cr. The selection is finally realised between each parent

[1]The optimisation may also be a maximisation by reversing the sign of the objective function

and the corresponding trial, keeping the best ones that are fitted to the problem. The migration technique moves the variables of the individuals towards the domain boundaries if the diversity of the population is too low. The diversity is a measure of how close the individuals are clustered around the best individual. For the diversity calculation, we use ϵ_2 and $\vec{\epsilon_3} \in \mathbb{R}_+^n$ (one for each variable) as relative and absolute precisions for the problem solved. This prevents stagnation around a local minimum when all individuals are close together. A detailed description about the Differential Evolution (DE) can be found in Feoktistov (2006) and about its modified HDE form in Chang et al. (2007).

The idea of the hybridisation came from the conclusion of Hansen & Kern (2004), in which the authors state that only if the function is additively separable then the CMA-ES is outperformed by the DE. As in practice we generally don't know a priori if the function is additively separable or not, using a hybrid of the two methods might be more robust than methods taken separately. In Kämpf & Robinson (2009) this hypothesis was tested with success on the Ackley (non-additively separable) and Rastrigin (additively separable) benchmark functions. Finally, the hybrid compared well with MOO (a multi-objective optimiser) on the problem of placement of buildings for solar irradiation maximisation. The proposed hybridisation is a serialisation of the two algorithms: We run the CMA-ES for 10 generations, save the covariance matrix and the global step size for the next run and then follow by 10 generations of HDE. This sequence is repeated until a user-defined maximum number of function evaluations is reached. The best individuals of the two algorithms are exchanged when switching algorithms.

For the hybrid algorithm found in Kämpf & Robinson (2009) a compromise was reached between exploration and exploitation, which is further investigated in this paper on different benchmark functions and a building optimisation problem. Therefore, we use the same parameters as those used in previous runs of the hybrid (see Table 1). For the HDE the *rand3* strategy (Feoktistov,

Table 1. Parameters used for the hybrid CMA-ES/HDE algorithm

Algorithm	Parameters			
	Population size			
CMA-ES	$\mu = 2 + \lfloor 1.5 \cdot \log(n) \rfloor$	$\lambda = 4 + \lfloor 3 \cdot \log(n) \rfloor$	$\sigma = 0.2$	
HDE	$NP = 30$	$F = 0.3$	$Cr = 0.1$	$\epsilon_2 = 10\%$

2006, pp.48-49) was used and the absolute precisions (Kämpf & Robinson, 2009) $\vec{\epsilon_3} \in \mathbb{R}_+^n$ (one for each variable) are problem dependent and selected as $\frac{1}{32}$ of the problem step sizes, which corresponds to half the finest grid size of the PSO/HJ algorithm described in the next paragraph.

2.1.2 Hybrid PSO and HJ

The Particle Swarm Optimisation algorithm was initially proposed by Eberhart & Kennedy (1995). It is a population-based algorithm, in which each individual is called a particle. Those particles evolve within generations mimicking the social behaviour of flocks of birds or schools of fish. They change their location going towards a point of lower objective function value known from previous iterations, modelling the cognitive behaviour, but also towards regions of space where other particles had a lower objective function value, modelling the social behaviour. For the cognitive behaviour, a corresponding acceleration is given as a parameter algorithm, modifying the speed of the particle proportionally to the difference vector between the particle and the local best point. For the social behaviour, another acceleration is given as parameter, modifying the speed according to the difference vector between the particle and the global best point. The global best point is being taken in the neighbourhood of the particle.

This heuristic is a global optimisation algorithm that generally finds a good solution at the expense of many function evaluations. The implemented version of the algorithm uses a constriction coefficient, which reduces the velocity of the particles. Moreover, it is put on a mesh, meaning that even though the variables are considered as continuous in the algorithm, when the evaluation of the objective function takes place, the coordinates of the nearest mesh point are used to compute the objective function. The distance between the mesh points is given by a

step size for each variable. Since the particles are put on a rectangular grid, the von Neumann neighbourhood is used.
The HJ algorithm (Hooke & Jeeves, 1961) is a member of the family of Generalised Pattern Search algorithms (Audet & Dennis, 2002). It searches along each coordinate direction for a decrease in the objective function. The initial mesh size for the search is given by a step size for each variable and when no improvement in the objective function is achieved, the step size is divided by a mesh size divider. When the local search around the current point finds a better point, the algorithm tries to make a global search move continuing in the same direction. As far as the global search finds a better point, it continues moving in the same direction until it fails, in which case the local search is restarted around the last best point. The local search and the algorithm ends once the maximal number of step reductions is attained. For this algorithm, convergence to a stationary point (a point at which the gradient is zero) is guaranteed for unconstrained, differentiable problems (Torczon, 1997; Audet & Dennis, 2002) and also for bound constrained problems (Lewis & V., 1999). However, on a multi-modal objective function the algorithm may get stuck at a local minimum.
The idea of the hybridisation is to use the PSO as a global optimisation algorithm, which gets close to the global minimum and then refine the position of the attained minimum by the HJ algorithm. Practically, the PSO algorithm is executed for a user-specified number of generations, and then the HJ uses as its initial search point the best individual obtained by the PSO algorithm. For the coupled PSO/HJ, because the PSO algorithm evaluates the cost function only a finite number of times, the proofs of (Torczon, 1997; Lewis & V., 1999; Audet & Dennis, 2002) that establish convergence to a local minimum on differentiable functions still apply. However, as for other metaheuristic algorithms, for multimodal functions convergence can only be established to a local but not a global minimum. For a detailed description of the hybrid or each algorithm separately please refer to Wetter (2004).
To optimise the benchmark functions, five algorithm parameter sets were used for the PSO algorithm. Four are taken from Wetter & Wright (2004) and the last one from Bui et al. (2007), as summarized in Table 2. The set for which the PSO algorithm performs best, in terms of

Table 2. Parameters used for the PSO algorithm within the hybrid PSO/HJ algorithm for the benchmark functions

	particles	c_1 cognitive acceleration	c_2 social acceleration	λ maximum velocity gain	κ constriction coefficient
Variant 1	16	2.8	1.3	0.5	0.5
Variant 2	16	2.8	1.3	0.5	1
Variant 3	36	2.8	1.3	0.5	0.5
Variant 4	36	2.8	1.3	0.5	1
Variant 5	100	2.05	2.05	0.2	1

identifying benchmark functions' optima, will then be used for the real-case applications with EnergyPlus. For the HJ algorithm we use as parameters $r = 2$, $s = 0$, $t = 1$ and $m = 4$, where r is the mesh size divider, s is the initial mesh exponent, t is the mesh size exponent increment and m is the number of step reductions.

2.1.3 Constraint handling

In our examples, we have analytical constraints and time consuming evaluations. We want to avoid evaluating potential solutions that do not satisfy the constraints. Therefore, we implemented the "Modification of the Selection Operation" proposed by Feoktistov (2006, pp.34-35). It redefines the "is better than" operator, by taking into account a pure Pareto dominance defined in a constraint function space. This latter operator goes as follow: $\vec{x_1}$ is better than $\vec{x_2}$ if and only if $\Phi \vee \Psi$, where $\Phi = (\forall k \in \{1, ..., m\} : g_k(\vec{x_1}) \leq 0 \wedge g_k(\vec{x_2}) \leq 0) \wedge (f(\vec{x_1}) < f(\vec{x_2}))$ and $\Psi = (\exists k \in \{1, ..., m\} : g_k(\vec{x_1}) > 0) \wedge (\forall k \in \{1, ..., m\} : \max(g_k(\vec{x_1}), 0) \leq \max(g_k(\vec{x_2}), 0))$. Please note that if $\Phi \vee \Psi$ is false, nothing is said about $\vec{x_2}$ (i.e. it doesn't always imply that $\vec{x_2}$ is better

than $\vec{x_1}$). The application of this comparison operator within the different sub-algorithms is explained below. This method of handling constraints allows individuals violating the constraints to survive in the first generations of the algorithms, and therefore bringing diversity, until the constrained domain is touched.

2.1.3.1 ***CMA-ES.*** When the mutation phase proposes a variable that is outside of the domain boundaries for that variable, it is put back at the nearest domain boundary. The mutation phase is repeated on an individual as long as it remains outside of the constrained space, but for a maximum of 10 times. The comparison operator described above is applied to the children population for sorting, before the elitist selection of the new generation's parents.

2.1.3.2 ***HDE.*** When the trial individual has a variable that is outside of the domain boundaries for that variable, it is put back randomly within the boundaries. In the selection phase, the comparison operator is used to compare the candidate with the trial (in that order). If the candidate is better than the trial, the candidate is kept, otherwise the trial is kept. This ensures that when both individuals satisfy the constraints and the objective functions are equal, the trial is preferred, bringing diversity in the population and preventing stagnation. Moreover, when both individuals do not satisfy the constraints, the candidate individual is kept only if it dominates over all constraints at the same time, allowing the trial to be selected in most cases, for the same diversity reasons.

2.1.3.3 ***PSO.*** When the particles' positions are updated, if a variable is outside of the domain boundaries, it is put back within the domain in the following way: $x_i \leftarrow 2 \cdot l_i - x_i$ if $x_i < l_i$ and $x_i \leftarrow 2 \cdot h_i - x_i$ if $x_i > h_i$. The procedure is repeated as long as the variable remains outside of the domain boundaries. The best local and global particles are selected using the comparison operator. If the best particles are better than the new proposed ones, they will remain the best ones, otherwise the new ones become the new best ones.

2.1.3.4 ***HJ.*** In the global and local search, if a proposed variable is outside of the domain boundaries, we assign $f(\vec{x}) = \infty$, thereby the iterate is rejected. Moreover, the new solution is compared to the former one using the comparison operator, keeping the new one only if it is better.

2.1.4 Implementation

An implementation in C++ of the above algorithms is available for download at `http://leso.epfl.ch/e/research_urbdev.html`.

2.2 Benchmark functions

Benchmark functions are designed to test the performance of the optimisation algorithms. In particular they are intended to represent some of the complexity that can be encountered in real-world optimisation problems. In this study, we have chosen five functions, with different degrees of complexity, to test the algorithms presented in the former section. We are interested to learn which algorithm is better suited to a certain kind of objective function shape.
The first benchmark function is the generalized Ackley function of dimension n,

$$f_n(\vec{x}) = -a \exp\left(-b\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n} \cos cx_i\right) + a + \exp(1), \tag{2}$$

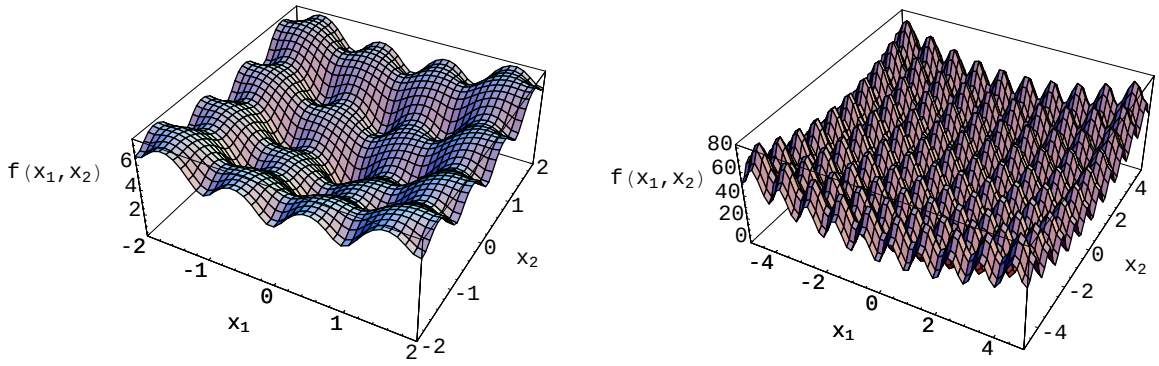


Figure 1. The two dimensional Ackley function (on the left) and the two dimensional Rastrigin function (on the right)

where $a = 20$, $b = 0.2$ and $c = 2\pi$. Its domain is $-32.768 \leq x_i \leq 32.768$, for all $i = 1..n$. This function is multi-modal with a global minimum at $f(\vec{0}) = 0$ which is surrounded by many local minima. Figure 1 shows the Ackley function in two dimensions. The step size for the optimisation algorithms was chosen to be 0.5, which corresponds to twice the frequency of the cosine perturbations. This allows for a complete sampling of the main features of the generalised Ackley function.
The second benchmark function is the generalised Rastrigin function of dimension n,

$$f_n(\vec{x}) = nA \sum_{i=1}^{n} x_i^2 - A\cos(\omega x_i), \tag{3}$$

where $A = 10$ and $\omega = 2\pi$. Its domain is $-5.12 \leq x_i \leq 5.12$, for all $i = 1..n$ and the global minimum is at $f(\vec{0}) = 0$. This function is highly multi-modal with many sub-peaks increasing in intensity when approaching the global minimum. Figure 1 shows the Rastrigin function in two dimensions. We have chosen the same step of 0.5 used for the generalised Ackley function.
The third benchmark function is the generalized Rosenbrock function for dimension n,

$$f_n(\vec{x}) = \sum_{i=1}^{n-1} 100 \cdot (x_i^2 - x_{i+1})^2 + (1 - x_i)^2. \tag{4}$$

Its domain is $-2.048 \leq x_i \leq 2.048$, for all $i = 1..n$. In two dimensions, this function is unimodal and banana shaped but slightly asymmetric, the minimum being at $\vec{x} = (x_1, x_2) = (1, 1)$. However, for dimensions higher than three, the function is no longer unimodal and has a local minimum in the neighbourhood of $\vec{x} = (-1, 1, ..., 1)$ (Shang & Qiu, 2006) in addition to the global minimum at $f(\vec{1}) = 0$. In this case, a step size of 0.1 was chosen.
The fourth benchmark function is the Sphere function of dimension n,

$$f_n(\vec{x}) = \sum_{i=1}^{n} x_i^2. \tag{5}$$

Its domain is $-1 \leq x_i \leq 1$, for all $i = 1..n$ and the global minimum is at $f(\vec{0}) = 0$. This function for which we use a step size of 0.5 is completely symmetric along every axis.
The last benchmark function has 13 variables and 9 linear constraints, it was designed to test

different constraint handling methods (Michalewicz & Schoenauer, 1996),

$$f(\vec{x}) = 5x_1 + 5x_2 + 5x_3 + 5x_4 - 5\sum_{i=1}^{4} x_i^2 - \sum_{i=5}^{13} x_i + 15, \tag{6}$$

subject to the following constraints:

$$2x_1 + 2x_2 + x_{10} + x_{11} \leq 10, \quad -8x_1 + x_{10} \leq 0, \quad -2x_4 - x_5 + x_{10} \leq 0, \tag{7}$$

$$2x_1 + 2x_3 + x_{10} + x_{11} \leq 10, \quad -8x_2 + x_{11} \leq 0, \quad -2x_6 - x_7 + x_{11} \leq 0, \tag{8}$$

$$2x_1 + 2x_3 + +x_{11} + x_{12} \leq 10, \quad -8x_3 + x_{12} \leq 0, \quad -2x_8 - x_9 + x_{12} \leq 0. \tag{9}$$

Its domain is $0 \leq x_i \leq 1$, for all $i = 1..9$ and $i = 13$ with a step size of 0.01, moreover $0 \leq x_i \leq 100$ for $i = 10, 11, 12$ with a step size of 0.1. The function is quadratic with a global minimum at $f(1, 1, 1, 1, 1, 1, 1, 1, 1, 3, 3, 3, 1) = 0$.

2.3 Real-world applications with EnergyPlus

We used the simulation software EnergyPlus version 2.2 to compute the objective function for real-world optimisation problems. The parameters varied during the optimisation were the window position, HVAC control temperatures and the temperatures used for system sizing. Two different benchmark buildings within the United States were used, each for the three locations, Chicago (IL), Miami (FL) and San Francisco (CA). In each case our aim is to minimize the primary energy consumption,

$$f(\vec{x}) = \eta_h \cdot Q_h(\vec{x}) + \eta_{el} \cdot E_{el}(\vec{x}), \tag{10}$$

where $\vec{x}$ is the vector containing the independent parameters, η_h is the source-site energy factor for the heating system primary resource, Q_h is the total annual on site energy consumption for heating and domestic hot water production (in J), η_{el} is the source-site energy factor for electricity and E_{el} is the total annual electric consumption (in J). We have taken primary source-site factors from Deru & Torcellini (2007) which are shown in Table 3.

Table 3. Source-site energy factors used in our study

	Chicago	Miami	San Francisco
Gas	1.092	1.092	1.092
Electricity	3.546	3.317	3.095

2.3.1 Small Office Building

The first set of numerical experiments uses the DOE benchmark case of a single story office building with one thermal zone. The floor area is 511 m^2 and the floor height is 3.05 m. The envelope properties vary with climate according to ASHRAE 90.1-2004. The original window to wall ratio is 18%, which corresponds to a window of height 0.55 m which spreads on the entire length of the wall. This ratio will vary throughout the study. The infiltration rate is 0.5 h^{-1} when fans are off and 0.15 h^{-1} otherwise. The HVAC system consists of packaged single zone air conditioning units and a gas furnace. Twenty people occupy the office during working hours, with an appliance load of 8.07 W/m^2. The interior lights have a peak power consumption of 10.8 W/m^2, of which 40% are in the central part of the building and controlled by the working hours schedule. The remaining 60% have a dimming control that uses two measurement points placed, respectively, at 3 m distance from the south and north windows. No shading devices or overhangs are present in the building. Figure 2 shows a projection of this small office building.

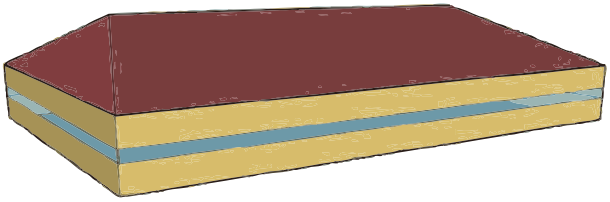


Figure 2. A projection of the Small Office Building simulated with the initial windows position

2.3.1.1 ***Parameters.*** The thirteen parameters for the study are shown in the Table 4. There are eight parameters describing the windows' lower and upper positions in each facade. The minimal allowed value for the lower window position is 0.8 m, which corresponds to a standard desk height. This choice was made as in office buildings desks often obstruct the daylight coming from the windows. The maximal allowed value for the lower window position is 1.25 m, which corresponds to the eye height of a seated person (see Watson & Crosbie (2004)) and hence permits a view to the outside world. For the upper window position, the minimal value was set to accommodate the minimal window size of 0.55 m. The maximal value for the upper window position was selected to be 2.2 m, which takes into account the slab height and the space needed for ventilation and air conditioning equipment. In addition to the window size and position, we varied the cooling supply air temperature used when sizing the system. A higher value results in larger flow rates which increases the fan energy use but reduces the chiller electricity consumption. The last four parameters are the control set points for the night and week-end temperature set-back for heating and cooling. The temperature set points during the day on

Table 4. The thirteen parameters for the Small Office Building study and the Large Office Building study

Parameter description	Symbol and Domain
North, east, south and west window lower positions (m)	$x_1, x_3, x_5, x_7 \in [0.8, 1.25]$
North, east, south and west window upper positions (m)	$x_2, x_4, x_6, x_8 \in [1.35, 2.2]$
Cooling supply air temperature used for system sizing$(^\circ C)$	$x_9 \in [12, 18]$
Heating setback night set point temperatures for Weekdays & Saturdays $(^\circ C)$	$x_{10} \in [13, 21]$
Heating setback whole day set point temperatures for Sundays & Holidays $(^\circ C)$	$x_{11} \in [13, 21]$
Cooling setback night set point temperatures for Weekdays & Saturdays $(^\circ C)$	$x_{12} \in [24, 36]$
Cooling setback whole day set point temperatures for Sundays & Holidays $(^\circ C)$	$x_{13} \in [24, 36]$

weekdays and Saturdays in winter and summer are 21°C and 24°C. Those are included in the range of allowed values for the setback temperatures.
To automatically size the HVAC system, EnergyPlus requires both the supply air temperature and absolute humidity to be specified. As the supply air temperature is varied over a large range, we compute the absolute humidity, which is input to the sizing algorithms of EnergyPlus as follows:

$$\omega = 6.875 \cdot 10^{-4} x_9 - 2.5 \cdot 10^{-4}, \tag{11}$$

where w is the moisture content of the cooling supply air (kg/(kg dry air)). This leads to a supply air relative humidity of about 90%.
The step sizes for the optimisation algorithms were chosen to be 0.05 for the window positions x_1 to x_8 and 0.25 for the temperature set points x_9 to x_{13}.

2.3.1.2 ***Remark.*** In our early optimisation experiments, the optimiser selected in all cases the minimum supply air temperature of 12°C as the optimum, because this led to the smallest fan power consumption. The reason was that the small office building had a direct evaporating coil model in which the COP is a function of the coil supply air inlet temperature but is independent of the coil supply air outlet temperature. Therefore, this model was not detailed enough to find

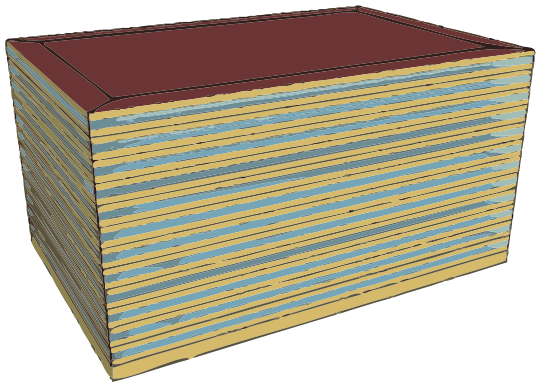


Figure 3. A projection of the Large Office Building simulated with the initial windows position

the optimum supply air temperature, as decreasing the supply air temperature does in an actual system decrease the COP. To overcome this limitation, we replaced the HVAC system of the small office building that is part of the DOE Commercial Building Benchmark files with that from the large office building. The HVAC system is described in the next section.

2.3.1.3 ***Constraints.*** There are four linear constraints to ensure a minimal window height of 55 cm, which is the standard window size in the DOE benchmark building description files. The constraints are:

(1) $g_1(\vec{x}) = x_1 - x_2 + 0.55 \leq 0$
(2) $g_2(\vec{x}) = x_3 - x_4 + 0.55 \leq 0$
(3) $g_3(\vec{x}) = x_5 - x_6 + 0.55 \leq 0$
(4) $g_4(\vec{x}) = x_7 - x_8 + 0.55 \leq 0$

2.3.2 Large Office Building

The second set of numerical experiments uses a 12 stories office building with a basement. The floor area is 42757 m^2, and its aspect ratio is 1.5. The envelope thermal properties vary with climate according to ASHRAE Standard 90.1-2004. The window to wall area ratio is 40%, which corresponds for the floor height of 3.05 m to a window size of 1.22 m. Each floor of the building, except the basement, is subdivided into five thermal zones. For each floor there are four 4.5 m deep perimeter zones with an infiltration rate of 0.3 h^{-1} when fans are off and 0.15 h^{-1} when fans are on. There is also one central zone with an infiltration rate of 0.15 h^{-1} when fans are off and 0.075 h^{-1} when fans are on. The HVAC system is a variable air volume flow system with reheat. The internal gains are composed of lights (10.8 W/m^2), electrical plug loads (8.07 W/m^2) and the heat release by the presence of 195 people. No shading devices are present, but daylighting control is used in all perimeter zones. A projection of the building is shown in Figure 3.

2.3.2.1 ***Parameters.*** The same thirteen parameters as for the Small Office Building were used (see Table 4). In the standard case defined in the DOE Benchmark Description files, the parameters for the window positions are of 0.91 m and 2.13 m respectively for the lower and upper window. The window positions x_1 to x_8 are interpreted as relative to each floor equipped with windows, which excludes the basement. In other words, all windows simulated on a facade have the same dimensions given by the first eight parameters.

2.3.2.2 ***Constraints.*** The same four linear constraints as for the Small Office Building are used for this Large Office Building case.

2.3.2.3 ***Remark.*** Even though the same parameters and linear constraints as for the Small Office Building were used, we expect a more complex objective function because it has, for each floor, separate thermal zones on each facade as well as an interior thermal zone.

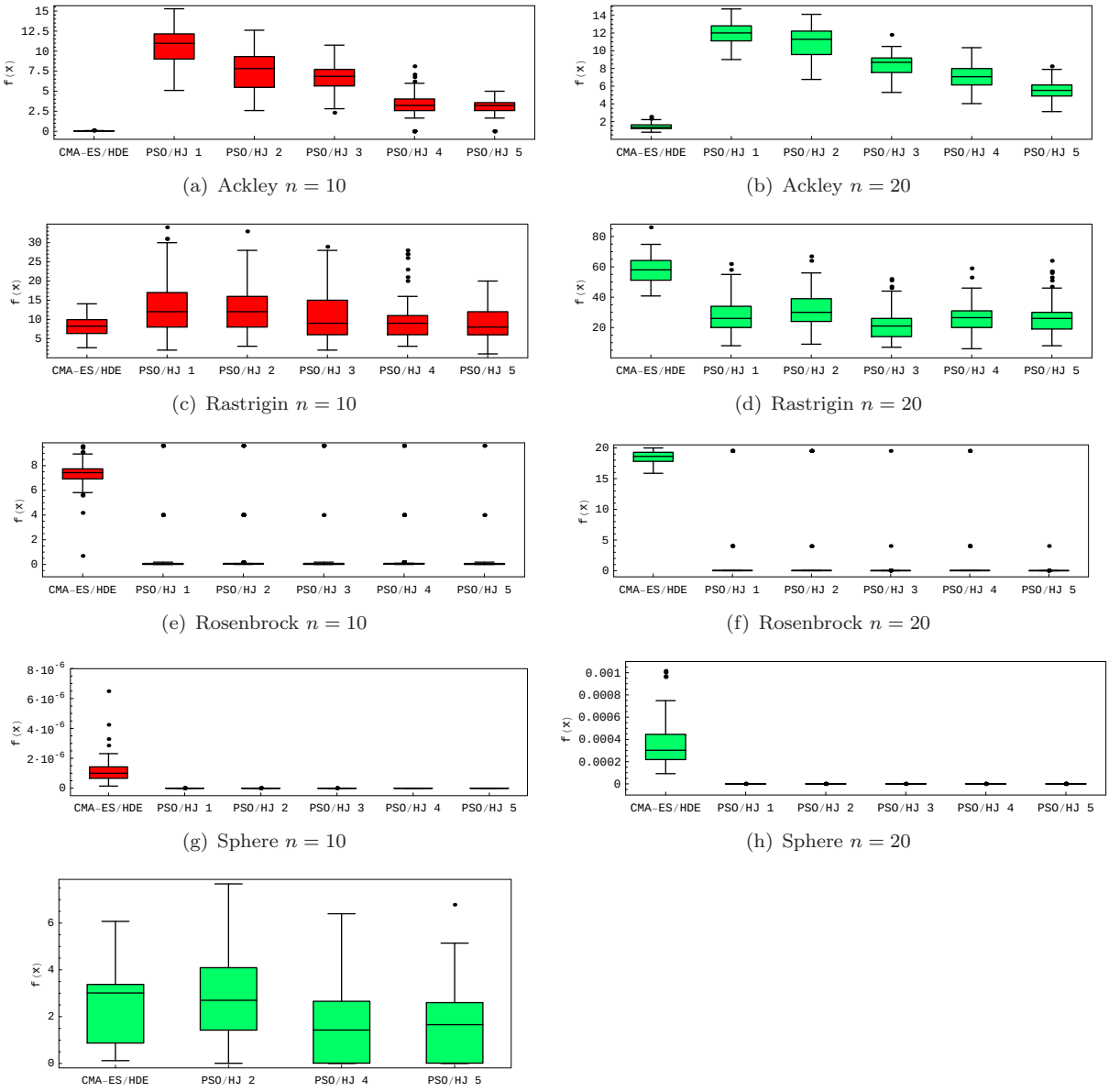
3. Results

We measured the time required for one simulation with the version 2.2 of EnergyPlus on a Linux machine equipped with a Quad-Core AMD Opteron 2.3 GHz processor and 1 GB of RAM. The Small Office Building takes about 80 seconds to complete and the Large Office Building about 900 seconds. We selected a *limit of 3000 evaluations*, leading to a reasonable 17 hours of simulations for the Small Office Building using the four cores available in parallel. For the Large Office Building, using four of such processors (16 cores available in parallel), it leads to 2 days of simulation. In comparison with this simulation time, the computing time overhead associated with the optimisation algorithm is negligible. Please note that the CMA-ES/HDE will remain close to the limit of 3000 evaluations, as it will not get interrupted during a population evaluation. However, the PSO/HJ algorithm takes the 3000 evaluations as the limit for the PSO algorithm, and then the solution is refined by the HJ algorithm which stops itself when no improvement is found with the finest grid spacing allowed. Typically, the improvement by the HJ takes only a few hundreds of evaluations and therefore the total number of evaluations is not far from the limit established.

3.1 Benchmark functions

The tests with the benchmark functions were conducted with 10 and 20 problem variables, as it is in this range that our real-world experiment with EnergyPlus will be. In order to obtain useful statistics for the different algorithms, hundred runs were carried out for each objective function, in which we varied the seed of the pseudo-random number generator. Figure 4 presents the lowest objective function value found by the hundred runs of the algorithms in the form of Box-Whisker plots. The median of the set is represented by the solid horizontal line in the box, the box itself includes the values lying between the first and third quartiles of the set (50% of the values) and outliers values included in the mustaches are within 1.5 of the inter-quartile distance. The remaining outliers that are not within 1.5 of the inter-quartile distance are represented by dots.

For the Ackley function, the CMA-ES/HDE algorithm consistently performs best by getting closer to the global minimum compared to the PSO/HJ algorithm even considering all outliers. For all the runs and all parameters, the PSO remains stuck in the neighbourhood of a local minima. The HJ then refines the position of the local minimum and obtains its exact position, which improves the function value, but only insignificantly. Therefore, the different objective function values arrived at by the PSO/HJ algorithm correspond to local minima. This is not the case for the CMA-ES/HDE algorithm, for which there is no guarantee to find a stationary point. We notice that increasing the number of particles for the PSO (going from PSO/HJ 1 to PSO/HJ 5) improves the probability of finding a better local minimum and indeed of approaching the global minimum as the parameter range for this function is large. We note that our choice of limiting the number of function evaluations to 3000, in view of the typically rather large computing time of a building simulation, leads to a limited performance of the PSO algorithm, which usually requires more function evaluations to provide good performance (Parsopoulos & Vrahatis, 2002). On the contrary for the CMA-ES/HDE, 100% of convergence (considered by a fitness below 0.1) of the Ackley function with 10 variables was obtained with



(a) Ackley $n = 10$ (b) Ackley $n = 20$

(c) Rastrigin $n = 10$ (d) Rastrigin $n = 20$

(e) Rosenbrock $n = 10$ (f) Rosenbrock $n = 20$

(g) Sphere $n = 10$ (h) Sphere $n = 20$

(i) Constrained $n = 13$

Figure 4. Comparison of different algorithms with benchmark test functions, Box-Whisker plots of hundred best candidates shown by their objective function value $f(\vec{x})$.

2695±178 evaluations by Kämpf & Robinson (2009). This performance was repeated in this experiment.

In the case of the Rastrigin function, the CMA-ES/HDE performs best for dimension 10. The median value is comparable to that of PSO/HJ 5, but its spread is smaller as the algorithm gets more often closer to the global minimum. However, from Kämpf & Robinson (2009), the CMA-ES/HDE algorithm would have needed 6255±2020 evaluations for 100% of convergence with 10 variables. For dimension 20, the CMA-ES/HDE suffers even more from the low number of total evaluations and the PSO/HJ performs better, particulary the PSO/HJ 3.

For the Rosenbrock function, the CMA-ES algorithm is always inferior to the PSO/HJ algorithm. The HJ part of the PSO/HJ algorithm can almost always improve the solution and get to the global minimum with an initial step size of 0.1 and 4 step reductions, thereby the box representing 50% of the runs collapses to a line where the objective function value is zero. However, had we selected more step reductions, then the step size would have been sufficiently small for the

algorithm to converge to a minimum, as can formally be proven (Torczon, 1997; Lewis & V., 1999). We note that for the PSO/HJ 5, the PSO brings the solution sufficiently close to a minimum for the HJ to always find the local minimum around $f(\vec{x}) = 4$, or the global minimum around $f(\vec{x}) = 0$.
For the Sphere function, both hybrid algorithms perform well. The PSO/HJ algorithm always converges exactly to the global minimum. Indeed, as this function is uni-modal and the global minimum on the grid defined for the algorithm, the HJ always converges. The CMA-ES/HDE algorithm gets very close to the global minimum, but turns around it due to the nature of its recombination and mutation operators.
For the constrained function, the variants 1 and 3 of the PSO/HJ algorithm, with a constriction coefficient $\kappa = 0.5$, were not able to get inside the constrained space during our tests. The PSO was concentrating rapidly on a region outside of the constrained space and got stuck there as the speed of particles reduced quickly with generations. The other variants of the PSO algorithm could at least find one point in the constrained space that attracted the particles. When particles could get in the vicinity of the global minimum, the HJ algorithm lead to its exact position, as it is on the grid defined by the algorithm. In this constrained example, the global minimum sits at the domain boundaries, and even though it is not a stationary point of the function, the HJ algorithm can lead to it thanks to its constraint handling procedure. The CMA-ES/HDE provided a less good performance than variants 4 and 5 of the PSO/HJ. Due to its recombination and mutation operators, it tends to explore the interior of the domain boundaries, however thanks to the constraint handling procedure within the mutation phase of the CMA-ES algorithm, it can also touch the border of the domain.
Even though the number of function evaluations is not the same for the two hybrid algorithms, as the HJ algorithm stops itself when no further improvement is found, clear trends can be identified. In particular it is possible to identify which algorithm is best suited to a certain kind of objective function. For example, the CMA-ES/HDE algorithm performs best on highly multi-modal functions such as Ackley and Rastrigin, as the algorithm was designed for just these kinds of functions. On the other hand, for functions with one or two minima such as Rosenbrock or Sphere, the PSO/HJ very frequently converges to the global minimum. In those cases, the PSO algorithm explores the parameter space and when the best particle approaches the global minimum, the HJ algorithm is able to find its exact position. The real-world experiment in building performance optimisation presented in this paper should favour one algorithm over the other according to the shape of the objective function.

3.2 Real-world application with EnergyPlus

From the results obtained for the benchmark test functions, the parameters of variant number 5 were adopted for the PSO/HJ algorithm.

3.2.1 Small Office Building

The results for the objective function minimisation (see Equation 10) are summarized in Figure 5. In each case the primary energy consumption has been normalised per unit floor area to allow for an easier comparison between the different buildings. For illustration purposes, the Box-Whisker plot presented were made with 5 runs at each location. Please note that the quartiles are not statistically significant in this latter plot. For the Chicago climate, the PSO/HJ performs the best, however from a Wilcoxon rank sum test (Mann & Whitney, 1947) the difference between the two is slightly significant (W=23, p-value=0.016). For the Miami climate, the difference between the two algorithms is not significant (W=14, p-value=0.42). Finally for the San Francisco climate, likewise the Miami climate, the difference is not significant (W=12, p-value=0.58). Moreover, from an energy point of view, the energy difference between the extreme cases corresponds to less or about 1% of the median value, which is negligible.
The PSO/HJ algorithm at the Chicago location stopped after a mean 3247 evaluations (stan-

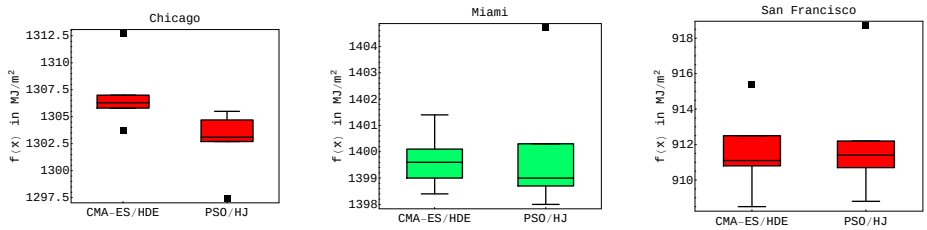


Figure 5. Primary energy consumption (in MJ/m^2) for the optimised Small Office Building, Box-Whisker plots for 5 runs at each location.

Table 5. Results for the Small Office building optimisation, worst cases for Chicago, best cases for Miami, close to median cases for San Francisco[a]

	x_1	$x_2 - x_1$	x_3	$x_4 - x_3$	x_5	$x_6 - x_5$	x_7	$x_8 - x_7$	x_9	x_{10}	x_{11}	x_{12}	x_{13}
	North window		East window		South window		West window						
Initial parameters	1.25	0.55	1.25	0.55	1.25	0.55	1.25	0.55	14.0	13.0	13.0	33.0	33.0
						Chicago (IL)							
CMA-ES/HDE	0.90	0.60	1.03	0.61	1.09	0.60	1.18	0.56	15.1	17.5	13.0	26.1	34.6
PSO/HJ	0.95	0.55	1.05	0.55	1.11	0.55	1.20	0.55	15.3	18.0	13.0	25.8	30.7
						Miami (FL)							
CMA-ES/HDE	0.86	0.73	0.84	1.36	0.84	1.36	1.09	1.11	12.0	18.6	16.8	27.1	31.6
PSO/HJ	0.88	0.73	0.80	1.40	0.85	1.35	1.05	1.05	12.0	15.2	19.8	27.0	34.8
						San Francisco (CA)							
CMA-ES/HDE	0.91	0.65	1.21	0.77	0.89	0.58	1.23	0.59	12.0	18.5	13.3	24.0	29.9
PSO/HJ	1.07	0.56	1.25	0.67	0.90	0.55	1.25	0.55	12.0	18.5	14.8	24.0	29.0

[a]Please refer to Table 4 for details about the parameters

dard deviation 52). At the Miami location, it stopped after a mean 3430 evaluations (standard deviation 89). Finally at the San Francisco location, it stopped after a mean 3307 evaluations (standard deviation 51).

Even though the annual primary energy use is similar between the algorithms, we want to see and understand the possible differences between the optimal variables values. For this, as an illustration, we have selected for each algorithm the worst cases for the Chicago climate, the best cases for the Miami climate and close to median cases for the San Francisco climate, and reported the corresponding variables in Table 5. In addition to that, we have included the variables corresponding to the standard case, which were given in the DOE benchmark description files. On closer inspection of the best candidates found by the two algorithms in Table 5, we notice that the parameters can be rather different. In order to understand this, we examine the results for each climate case in turn below.

3.2.1.1 ***Chicago (IL) case.*** For the Chicago (IL) case, the North, East and West window sizes are consistently around the minimal allowed value. Table 6 shows a breakdown of the energy consumption for the different systems in the building for both the initial and optimised cases. From this, we notice that for the optimised cases we save energy for both the cooling and ventilation compared to the initial case. However, more natural gas was consumed in heating, which may indicate that it is more beneficial to reduce electricity consumption as its source-site factor is about three times that of natural gas.

To produce Figure 6 we used the optimal parameters identified by the PSO/HJ algorithm shown in Table 5, but varied the variable x_9 to cover its domain. From this we find that the PSO/HJ algorithm correctly identified the minimum for this ninth parameter in the subspace shown by this Figure. As expected, we notice an increase in fan energy use if the supply temperature is raised, starting at the optimal value for x_9. This is because with a higher cooling supply air temperature, a higher mass flow rate is required to provide the same cooling load. Below the

Table 6. Details about the Chicago Small Office building's site energy consumption in MJ/m^2

Energy type and use (MJ/m^2)	initial	CMA-ES/HDE	PSO/HJ
Electricity			
Interior Lighting	100.7	97.28 (-3.4%)	98.10 (-2.6%)
Interior Equipment	128.8	128.8	128.8
Cooling	45.89	33.52 (-27.0%)	32.43 (-29.3%)
Natural Gas			
Heating	138.8	142.2 (+2.4%)	146.2 (+5.3%)
Electricity			
Fans	76.05	55.38 (-27.2%)	53.01 (-30.3%)
Pumps	11.3	6.46 (-42.8%)	6.11 (-45.9%)
Heat Rejection equipment (cooling tower)	8.45	4.99 (-40.9%)	4.70 (-44.4%)
Total primary energy use	1468	1313 (-10.6%)	1305 (-11.1%)

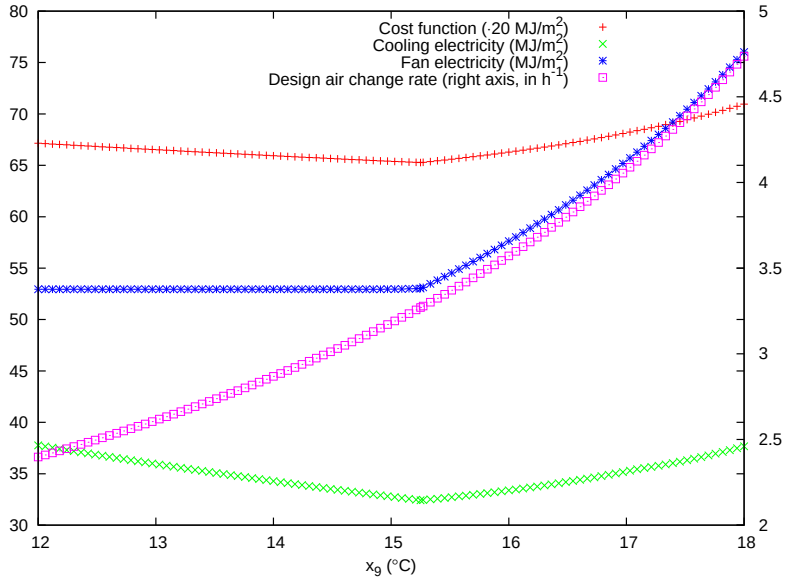


Figure 6. Variation of the fitness with x_9 around the best candidate found for the Small Office Building in Chicago with PSO/HJ

optimal value for x_9, we have a constant value for fan energy use, because for $x_9 < 15.3$°C, the sizing of the heating system determines the minimal mass flow rate. This behaviour cannot be seen for San Francisco and Miami in which the fan size was determined exclusively by the cooling load.

Looking closer at the EnergyPlus input file, we noticed that the ASHRAE 90.1-2004 regulation leads to poorly performing double glazed windows (2× 3 mm clear glass, 6 mm air gap) having a U-value of 3.26 W/(m^{2}K). To understand the implications of this, we re-run the optimisation after replacing the window construction with a better performing double glazed window. The new glazing consists of a double layer window of 6 mm clear glass, with a gap of 13 mm filled with Argon. The outer layer also has a low infrared emissivity coating. The new windows lead to a reduction in the objective function between the initial cases (without optimisation) equivalent to about 6% primary energy. Moreover, the optimised case with new glazing has larger windows which may be preferred by the occupants (see Table 7).

From Table 5, we notice that for Chicago, the optimal night set back temperature for heating was not at its lowest allowed value, which seems counter intuitive. To understand this, we examined results from two simulations. The first one was based on the optimal parameters found by the CMA-ES/HDE algorithm and the second one used the same parameters except the night set back temperature, which was set to its lowest allowed value of 13°C. This reduction in night set back temperature led to an increase in primary energy consumption of 9.7%. By way of

Table 7. Results for the Small Office building optimisation in Chicago with double glazing and low emission coating after 3000 evaluations, parameter values[b]

	x_1	$x_2 - x_1$	x_3	$x_4 - x_3$	x_5	$x_6 - x_5$	x_7	$x_8 - x_7$	x_9	x_{10}	x_{11}	x_{12}	x_{13}	$f(\vec{x})$
	North window		East window		South window		West window							MJ/m^2
Initial parameters	1.25	0.55	1.25	0.55	1.25	0.55	1.25	0.55	14.0	13.0	13.0	33.0	33.0	1381
Chicago (IL) equipped with Double Glazing and Low Emission coating														
CMA-ES/HDE	0.84	0.75	0.91	1.26	0.88	1.16	0.95	0.79	13.4	17.3	13.8	26.2	30.4	1200
PSO/HJ 5[a]	0.96	0.77	0.98	1.19	0.90	1.15	1.18	0.72	13.9	17.3	13.1	26.0	28.8	1199

[a]The parameters of the algorithm are variant 5: 100 particles, $c_1 = 2.05$, $c_2 = 2.05$, $\lambda = 0.2$, $\kappa = 1$

[b]Please refer to Table 4 for details about the parameters

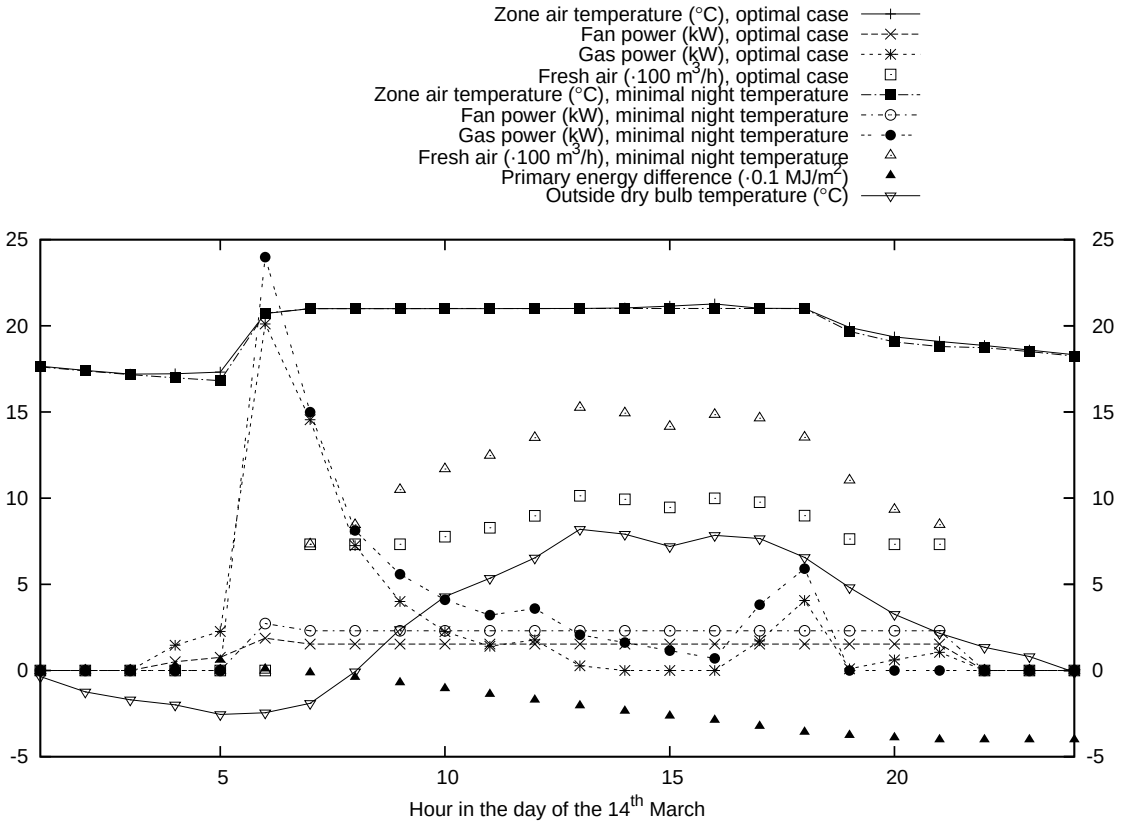


Figure 7. Comparison between the optimal point found for Chicago by the CMA-ES/HDE and the same parameters with the night set back temperature changed to 13°C. The day chosen was the 14^{th} March.

explanation, in Figure 7 we plot results for a particular day in which the performance of the optimal case was better than the one with the night set back at its lowest value. During that day, the power consumption for fans and heating were higher for the case with lower night set back temperature. This higher fan power can be explained by the auto-sizing algorithm used by EnergyPlus: during the night, the temperature of the building will decrease more, resulting in a bigger heat load for the first hour in the morning. Therefore, a larger fan is needed to provide the relatively larger amount of heating energy after the set point change.

The higher gas consumption was at first surprising, because the minimum outside air mass flow rate for ventilation was fixed according to the expected occupancy, and independent of the fan size. Furthermore, a larger fan adds more heat to the air stream. However, the EnergyPlus input file is such that the economizer control attempts to maintain a mixed air temperature of 16°C until the 31^{st} March and starting from the 1^{st} October. Since in the case of the lowest night temperature there is a higher air flow rate (due to a larger fan), more outside air is needed to bring the mixed air temperature down to 16°C. Consequently more gas is needed to reheat the

Table 8. Details about the Miami Small Office building's site energy consumption in MJ/m^2

Energy type and use (MJ/m^2)	Miami (FL) initial	CMA-ES/HDE	PSO/HJ
Electricity			
Interior Lighting	102.4	88.60 (-13.5%)	88.75 (-13.3%)
Interior Equipment	128.8	128.8	128.8
Cooling	120.1	110.5 (-8.0%)	110.5 (-8.0%)
Natural Gas			
Heating	0.96	1.41 (+46.9%)	1.39 (+44.8%)
Electricity			
Fans	85.56	63.60 (-25.7%)	63.50 (-25.8%)
Pumps	19.5	15.7 (-19.5%)	15.7 (-19.5%)
Heat Rejection equipment (cooling tower)	17.1	13.9 (-18.7%)	13.8 (-19.3%)
Total primary energy use	1571	1398 (-11.0%)	1398 (-11.0%)

mixed air to the building's supply air temperature.
From the above analysis, we conclude that the way the economizer was modeled in the DOE benchmark files used for this study was oversimplified for the heating season. However, for the summer, a mixed air temperature of 13°C was used, which is correct to take advantage of night cooling. Moreover, the temperature schedule used for the auto-sizing algorithms should ramp up to the daytime set point, instead of having a change that looks like a step function. If the room air set point temperature for heating had been ramped up over a period of 3 hours, then compared to a constant night set back temperature of 13°C, a reduction in the objective function of 9.4% would have been achieved, which is substantial.

3.2.1.2 ***Miami (FL) case.*** In the Miami case, the optimised cases have large windows (Table 5) especially for the East, South and West facades. The cooling supply air temperature is set to the lowest bound (12°C). The winter heating set back temperatures are close to the initial set point of 21°C, but these seem non-intuitive for the PSO/HJ algorithm as the set-back temperature for Sundays and Holidays is higher than that for weekdays and Saturdays. To understand this, we analysed the office temperatures for these two periods. During Sundays and Holidays, the minimal achieved room temperature was 19.6°C. For Weekdays and Saturdays during night time, the minimal temperature was 19.3°C. Therefore, any heating temperature set point below these temperatures will have the same effect, which explains the non-intuitive values. The summer cooling set back temperatures, on the other hand, do follow our intuition. The maximal room temperature at night time during weekdays and Saturdays achieves the set point temperature. However the maximal temperature during Sundays and Holidays was 30.4°C. Therefore any set point above this temperature leads to equivalent energy use, which explains the differences between the set points found by the CMA-ES/HDE and the PSO/HJ.
Table 8 shows the energy details of the different systems in the building for the optimised cases and the initial case. For the optimised cases energy for cooling is reduced but natural gas consumption is slightly increased, which is, however, negligible for the Miami climate. Fan electrical consumption shows the biggest savings. By reducing the cooling supply air temperature to its minimal value, the optimiser reduced the necessary mass flow rate for an equivalent cooling load.

3.2.1.3 ***San Francisco (CA) case.*** As with the Chicago case, for the San Francisco case the windows are set around the minimal allowed height. The cooling supply air temperature is also at its lowest value (12°C), allowing for a reduction in fan power. Table 9 shows the energy details of the different systems in the building. For the optimised cases we save about half the energy for cooling compared to the initial case. Perhaps of more interest is that for a similar reduction in total primary energy consumption, the CMA-ES/HDE results in larger windows and therefore reduces lighting energy use, but it also results in an increase of all other energy end uses compared to the solution found by the PSO/HJ algorithm. This indicates what may be

Table 9. Details about the San Francisco Small Office building's site energy consumption in MJ/m^2

Energy type and use (MJ/m^2)	San Francisco (CA) initial	CMA-ES/HDE	PSO/HJ
Electricity			
Interior Lighting	99.78	93.56 (-6.2%)	95.83 (-4.0%)
Interior Equipment	128.8	128.8	128.8
Cooling	40.14	19.55 (-51.3%)	19.06 (-52.5%)
Natural Gas			
Heating	35.03	26.52 (-24.3%)	24.15 (-31.1%)
Electricity			
Fans	47.36	32.86 (-30.6%)	32.25 (-31.9%)
Pumps	13.0	5.48 (-57.8%)	5.36 (-58.8%)
Heat Rejection equipment (cooling tower)	11.1	4.76 (-57.1%)	4.66 (-58.0%)
Total primary energy use	1091	911 (-16.5%)	911 (-16.5%)

Table 10. Results for the optimisation of the Large Office building in different locations after 3000 evaluations, yearly primary energy consumption in MJ/m^2

	Chicago (IL)	Miami (FL)	San Francisco (CA)
Standard parameter set	1689	1718	1440
Best candidate with CMA-ES/HDE	1347.7 (-20.2%)	1596.2 (-7.1%)	1014.5 (-29.6%)
Best candidate with PSO/HJ 5[a]	1348.2 (-20.2%)	1596.0 (-7.1%)	1015.2 (-29.5%)

[a]The parameters of the algorithm are variant 5: 100 particles, $c_1 = 2.05$, $c_2 = 2.05$, $\lambda = 0.2$, $\kappa = 1$

a multi-modal objective function with two almost equivalent minimum objective function values but different independent parameters.

3.2.2 Large Office Building

For the Large Office Building, considering the large simulation time of about 800 seconds for each function evaluation and the similar reduction in primary energy consumption obtained by different runs, we chose to do the optimisation only once for each algorithm. The results for the objective function minimisation are summarized in Table 10. We have added a further digit to the yearly energy consumption to help display the actual difference between the two algorithms, however this difference is insignificant in terms of energy consumption. As for the Small Office Building, the most significant improvement was achieved for San Francisco. For the Large Office Building, we expected a more complex objective function because it has, for each floor, separate thermal zones on each facade as well as an interior thermal zone, and therefore would favour the CMA-ES/HDE like the Ackley or Rastrigin benchmark functions. However, there is no discernable difference between the two tested optimisation methods as far as only one run of the algorithms is concerned.

Table 11 shows that both algorithms found similar parameters. We show in Figure 8 the evolution of the objective function with the number of evaluations for San Francisco. This Figure shows that the HDE component of the CMA-ES/HDE is most effective in improving the solution, as is the PSO component in the PSO/HJ. With both algorithms, a plateau has been reached around 2200 evaluations. The apparent divergent behaviour of the CMA-ES algorithm was already observed on the Rastrigin function in Kämpf & Robinson (2009). At each generation, the CMA-ES algorithm selects the new parents amongst the children, and those children may have a worse fitness than the actual parents. It appears that such a small populated CMA-ES algorithm with $\mu = 5$ and $\lambda = 11$ is not suited for this kind of function, which justifies the hybrid approach, as this knowledge comes after the optimisation. Moreover, this behaviour of the CMA-ES algorithm does not impact on the robustness of the hybrid, but rather brings diversity to the population of the HDE during the exchange of individuals.

Table 11. Results for the Large Office building optimisation in different locations after 3000 evaluations, parameter values[b]

	x_1	$x_2 - x_1$	x_3	$x_4 - x_3$	x_5	$x_6 - x_5$	x_7	$x_8 - x_7$	x_9	x_{10}	x_{11}	x_{12}	x_{13}
	North window		East window		South window		West window						
Standard	0.91	1.22	0.91	1.22	0.91	1.22	0.91	1.22	14.0	13.0	13.0	33.0	33.0
Chicago (IL)													
CMA-ES/HDE	1.22	0.55	1.01	0.56	1.15	0.55	1.22	0.59	12.1	14.6	13.0	24.8	34.4
PSO/HJ 5[a]	1.21	0.55	1.00	0.55	1.00	0.55	1.15	0.55	12.8	14.6	13.0	24.5	33.5
Miami (FL)													
CMA-ES/HDE	1.22	0.55	1.18	0.75	1.17	1.02	1.24	0.70	12.0	17.3	16.2	24.3	35.8
PSO/HJ 5[a]	1.18	0.56	1.25	0.73	1.23	0.97	1.25	0.68	12.0	17.6	16.0	24.3	36.0
San Francisco (CA)													
CMA-ES/HDE	1.20	0.69	1.17	0.69	1.20	0.55	1.19	0.58	12.0	14.4	13.0	24.5	34.9
PSO/HJ 5[a]	1.25	0.78	1.25	0.68	0.95	0.55	1.25	0.56	12.0	13.0	13.0	24.5	35.8

[a]The parameters of the algorithm are variant 5 with 100 particles, $c_1 = 2.05$, $c_2 = 2.05$, $\lambda = 0.2$ and $\kappa = 1$

[b]Please refer to Table 4 for details about the parameters

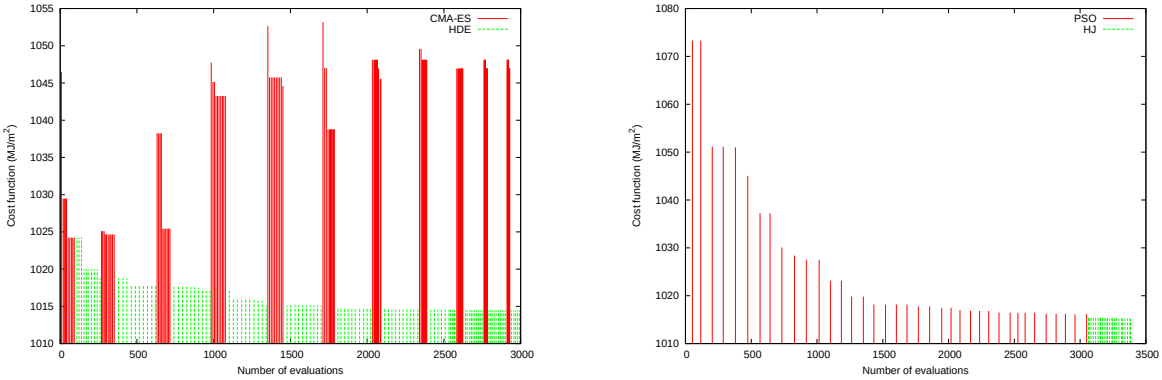


Figure 8. The improvement in cost function as a function of the number of evaluations for the Large Office in San Francisco, on the left for the CMA-ES/HDE and on the right for the PSO/HJ

4. Conclusion

The novel hybrid evolutionary algorithm CMA-ES/HDE was compared with the established performance of the PSO/HJ. The first set of numerical experiments involved benchmark functions with different complexities (Ackley, Rastrigin, Rosenbrock, Sphere and Constrained). The CMA-ES/HDE performed better than the PSO/HJ on difficult multi-modal functions such as those of Ackley and Rastrigin for a parameter space of dimension 10 and within a limit of 3000 evaluations. However, when the problem dimension was increased to 20, the CMA-ES/HDE performed less well than the PSO/HJ algorithm for the Rastrigin function, which indicates a limit of the algorithm for complex functions when using low number of function evaluations. For the non-convex Rosenbrock function in dimensions 10 and 20, the PSO/HJ performs best because the HJ algorithm finds very often the global minimum once the PSO algorithm has reached a basin of attraction of the global minimum. The uni-modal Sphere function is best solved by the HJ part of the PSO/HJ, which always converges exactly to the global minimum. Even though the CMA-ES/HDE algorithm does not get the exact position of the global minimum, it gets very close to it. The uni-modal constrained function favours some parameter variants of PSO/HJ and therefore indicates which algorithm parameters should be preferred.

We then tested the algorithms' performance for minimising the energy use of small and large office buildings simulated by EnergyPlus. Thirteen parameters were varied, representing window positions, HVAC system design variables and control set-points. The optimal configuration led to a reduction of primary energy consumption of up to 30%. The optimal values represented a trade-off that one can obtain by sizing the windows properly in order to save electricity for

artificial lighting, and by setting the HVAC sizing and control temperatures to reduce energy for fans, pumps and air conditioning systems. A quasi similar performance was obtained by the two algorithms CMA-ES/HDE and PSO/HJ for the small office building. For one climate only the PSO/HJ provided better results that were slightly significant. Moreover, similar performance was obtained for the large office building, as far as only one run of each algorithm is concerned. However, the resultant parameter sets were different, indicating that the objective function is multi-modal or locally just a flat landscape. We can conclude that even though the performance of the optimisation algorithms tested was significantly different on benchmark functions, it was not the case for the experiments with EnergyPlus. Even though the total number of function evaluations was not the same between the two hybrid algorithms, as the HJ algorithm stops itself when no further improvement is found in the objective function, clear trends were identified regarding the performance of the hybrid algorithms with benchmark functions. Highly multi-modal objective functions were best solved by the CMA-ES/HDE algorithm and objective functions with one or two minima by the PSO/HJ algorithm. The objective function of the building performance optimisation problem seemed to lie somewhere between these extremes, as no algorithm was significantly favoured over the other. Nevertheless, when we have no a-priori knowledge of the nature of the objective function that we wish to evaluate, it would be prudent to use the hybrid CMA-ES/HDE algorithm, which is able to handle highly multi-modal functions that may be found in building and urban optimisation problems. To this end work is under way to apply the hybrid CMA-ES/HDE algorithm in conjunction with an urban scale resource flow modelling tool, called CitySim. It is anticipated that results from application of this work will be reported in the near future.

Acknowledgements

The financial support received for this work from the Swiss National Science Foundation, under the auspices of National Research Programme 54 "Sustainable Development of the Built Environment" is gratefully acknowledged.
This research was also 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.

5. Nomenclature

f	objective function
$\vec{x}$	variables
n	problem dimension
m	constraints number
g_j	jth constraint
l_i	ith lower bound
u_i	ith upper bound
μ	CMA-ES number of parents
σ	CMA-ES global step size
NP	HDE population size
F	HDE constant of differentiation
Cr	HDE constant of crossover
ϵ_2	HDE relative precision for migration
$\vec{\epsilon_3}$	HDE absolute precision for migration
c_1	PSO cognitive acceleration constant
c_2	PSO social acceleration constant
λ	PSO maximum velocity gain

κ	PSO constriction coefficient
r	HJ mesh size divider
s	HJ initial mesh size exponent
t	HJ mesh size exponent increment
m	HJ number of step reductions
η_h	source-site energy factor for the heating primary ressource
Q_h	total annual on-site energy consumption for heating and domestic hot water production (J)
η_{el}	source-site energy factor for electricity
E_{el}	total annual on-site electric consumption (J)

References

EnergyPlus Engineering Reference: The Reference to EnergyPlus Calculations. (April 2009). , Technical report, U.S. Department of Energy.

Audet, C., & Dennis, J.E. (2002). Analysis of Generalized Pattern Searches. *SIAM Journal on Optimization*, *13*(3), 889–903.

Bradley, D., & Kummert, M. (2005). New Evolutions In Trnsys A Selection Of Version 16 Features. August 12-15 (pp. 107–114)., Montreal, Canada.

Bui, L.T., Soliman, O., & Abbass, H.A. (2007). A modified strategy for the constriction factor in particle swarm optimization. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*Vol. 4828 LNAI, pp. 333–344). .

Caldas, L.G., & Norford, L.K. (2002). A design optimization tool based on a genetic algorithm. *Automation in Construction*, *11*(2), 173–184.

Chang, C.F., Wong, J.J., Chiou, J.P., & Su, C.T. (2007). Robust searching hybrid differential evolution method for optimal reactive power planning in large-scale distribution systems. *Electric Power Systems Research*, *77*(5-6), 430–437.

Clarke, J. (2001). *Energy Simulation in Building Design*. Oxford: Butterworth-Heinemann.

Deru, M., & Torcellini, P., Source Energy and Emission Factors for Energy Use in Buildings. (2007). , Nrel/tp-550-38617.

Eberhart, R.C., & Kennedy, J. (1995). A new Optimizer using Particle Swarm Theory. Oct. (pp. 39–43)., Nagoya, Japan.

Feoktistov, V. (2006). *Differential Evolution: In Search of Solutions*. Springer.

Hansen, N., & Ostermeier, A. (2001). Completely Derandomized Self-Adaptation in Evolution Strategies. *Evolutionary Computation*, *9*(2), pp. 159–195.

Hansen, N., & Kern, S. (2004). Evaluating the CMA Evolution Strategy on Multimodal Test Functions. Springer-Verlag.

Hooke, R., & Jeeves, T. (1961). Direct search solutions of numerical and statistical problems. *Journal of the Association for Computing Machinery*, *8*, 212–229.

Kämpf, J.H., & Robinson, D. (2009). A hybrid CMA-ES and HDE optimisation algorithm with application to solar energy potential. *Applied Soft Computing*, *9*, 738–745.

Lewis, R.M., & V., T. (1999). Pattern search algorithms for bound constrained minimization. *SIAM Journal on Optimization*, (9), 10821099.

Mann, H.B., & Whitney, D.R. (1947). On a test of whether one of two random variables is stochastically larger than the other. *Annals of Mathematical Statistics*, *18*, 50–60.

Michalewicz, Z., & Schoenauer, M. (1996). Evolutionary Algorithms for Constrained Parameter Optimization Problems. *Evolutionary Computation*, *4*(1), 1–32.

Parsopoulos, K.E., & Vrahatis, M.N. (2002). Recent approaches to global optimization problems through Particle Swarm Optimization. *Natural Computing*, *1*, 235–306.

Shang, Y.W., & Qiu, Y.H. (2006). A Note on the Extended Rosenbrock Function. *Evolutionary Computation*, *14*(1), 119–126. PMID: 16536893

Torczon, V. (1997). On the convergence of pattern search algorithms. *SIAM Journal on Optimization*, (7), 1–25.

Watson, D., & Crosbie, M.J. (2004). *Time-saver standards for Architectural Design* (Eighth edition). McGraw-Hill.

Wetter, M., & Wright, J. (2004). A comparison of deterministic and probabilistic optimization algorithms for nonsmooth simulation-based optimization. *Building and Environment*, *39*(8 SPEC. ISS.), 989–999.

Wetter, M., GenOpt, Generic Optimization Program, User Manual, Version 2.0.0. (2004). , Technical report LBNL-54199, Lawrence Berkeley National Laboratory, Berkeley, CA, USA.

Wetter, M., & Polak, E. (2004). A convergent optimization method using pattern search algorithms with adaptive precision simulation. *Building Services Engineering Research and Technology*, *25*(4), 327–338.

Wright, J.A., Loosemore, H.A., & Farmani, R. (2002). Optimization of building thermal design and control by multi-criterion genetic algorithm. *Energy and Buildings*, *34*(9), 959–972.