Applied Soft Computing 9 (2009) 738–745

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

A hybrid CMA-ES and HDE optimisation algorithm with application to solar energy potential

Jérôme Henri Kämpf *, Darren Robinson

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

ARTICLE INFO

Article history:
Received 17 August 2007
Received in revised form 4 September 2008
Accepted 28 September 2008
Available online 17 October 2008

Keywords:
Optimisation
Covariance matrix adaptation evolution strategy (CMA-ES)
Differential evolution (DE)
Hybrid differential evolution (HDE)
Hybrid evolutionary algorithm
Hybrid CMA-ES/HDE
Urban planning
Solar energy potential

ABSTRACT

This paper describes the results of initial experiments to apply computational algorithms to explore a large parameter space containing many variables in the search for an optimal solution for the sustainable design of an urban development using a potentially complicated fitness function. This initial work concentrates on varying the placement of buildings to optimise solar irradiation availability. For this we propose a hybrid of the covariance matrix adaptation evolution strategy (CMA-ES) and hybrid differential evolution (HDE) algorithms coupled with an efficient backwards ray tracing technique. In this paper we concentrate on the formulation of the new hybrid algorithm and its testing using standard benchmarks as well as a solar optimisation problem. The new algorithm outperforms both the standalone CMA-ES and HDE algorithms in benchmark tests and an alternative multi-objective optimisation tool in the case of the solar optimisation problem.

1. Introduction

With increasing pressure to reduce the risks of climate change due to fossil fuel combustion, there is growing interest in optimizing the design of urban settlements for the exploitation of solar radiation both to reduce energy demands for heating and artificial lighting and to convert into heat (for hot water) and electricity. But past optimisation studies [1–7] have been based on manual trial and error, testing a limited number of subjectively conceived propositions using either physical or numerical tools. Far more powerful would be the coupling of computational algorithms with numerical ray tracing techniques to search the available parameter space for an optimum. Moreover the ray tracing tool should be capable of efficiently predicting the integral of solar irradiance received on built surfaces throughout a given time period (seasonal or annual). In this paper we describe how we have achieved these objectives. First we discuss the principles of optimisation algorithms and briefly review some of the different approaches that have thus far been explored, before selecting two specific algorithms (covariance matrix adaptation algorithm (CMA-ES) and hybrid differential evolution (HDE)) which we proposed to hybridise. We then describe the procedure we have followed in performing this hybridisation before finally testing the new algorithm using standard benchmark tests (the Ackley and Rastrigin functions) and a solar energy potential optimisation problem. As part of this latter we describe the method employed to efficiently solve for long-term solar irradiation simulation.

1.1. Optimisation algorithms

In all generality, we want to find the global maximum (or maxima) of a function f that depends on n independent decision variables. In formal terms, we are looking for the supremum (set of variables that maximises the function) as in the following equation:

$$sup\ \{f(\vec{x})|\vec{x}\in M\subseteq\mathbb{R}^n\} \quad (1)$$

with:

$n\in\mathbb{N}$
dimension of the problem
$f: M\rightarrow\mathbb{R}$
objective function
$M=\{\vec{x}\in\mathbb{R}^n|g_j(\vec{x})\geq 0,\ \forall\, j\in\{1,\ldots,m\}\}, M\neq\varnothing$

* Corresponding author. Tel.: +41 21 693 45 47; fax: +41 21 693 27 22.
E-mail address: jerome.kaempf@epfl.ch (J.H. Kämpf).

1568-4946/$ – see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.asoc.2008.09.009

feasible region

$m \in \mathbb{N}$

number of constraints

The set of inequality restrictions $g_j : \mathbb{R}^n \to \mathbb{R}, \forall j \in \{1, \ldots, m\}$ includes a special case of constraints due to the domain boundaries $L_i \leq x_i \leq H_i, \forall i = 1, \ldots, n$. $\vec{L}$ is named the lower bound and $\vec{H}$ the upper bound of the domain.

When the function to optimise exhibits a non-differentiable, non-linear behaviour and is multi-modal, it justifies the choice of heuristic methods such as evolutionary algorithms (EA). EA is a family of optimisation methods based upon the principles of Darwinian natural selection [8–10]. They are population-based heuristic algorithms, where each individual represents a potential solution of the function to optimise. The operators of recombination, mutation and selection are applied to get closer to a best solution through population generations.

The first evolutionary algorithms proposals come from the mid-60s with genetic algorithms (GAs) from Holland [11] at the University of Michigan, evolutionary programming (EP) from Fogel [12] at the University of California in San Diego and independently evolution strategies (ES) from Rechenberg [13] at the Technical University of Berlin. Their work brought a wide class of optimisation methods for difficult problems where few is known about the underlying search space. Koza [14], with the development of genetic programming (GP) beginning of 1990s, enriched the trend of evolutionary algorithms.

Facing a multitude of methods and approaches, we have made an arbitrary choice of focusing our attention to the evolution strategies branch of evolutionary algorithms. Evolution strategies were first developed by Rechenberg [13] and Schwefel [15] and have evolved into the cumulative step-path adaptation algorithm (CSA-ES) [16,17] and the CMA-ES [18,19]. The variables of the function to optimise are coded using a floating-point representation and are associated in phenotypes with standard deviations for mutation purpose. CMA-ES have been used to solve many optimisation problems [19] and are regarded as one of the best algorithm for real-value coded variables. In [19] Hansen and Kern concludes that CMA-ES is outperformed by differential evolution (DE) only if the function to optimise is additively separable.

Differential evolution was developed by Storn and Price in 1996 [20] and has proved to be another good candidate for real-value optimisation problem solving. DE is very simple to implement and relies only on variables with a floating point representation. The method is based on stochastic search; one of its drawbacks is the need for a large population to overcome local optima. Chang et al. [21] developed a hybrid algorithm of differential evolution to allow the use of a smaller population. However, for all kinds of DE, the results are very sensitive to the algorithm control parameters [22]. They must be well chosen for a good performance.

From Hansen and Kern's conclusion [19] we considered that a hybrid CMA-ES/HDE algorithm might combine the advantages of the two optimisation methods, since in real life applications we tend to face optimisation problems where the dependence of the function on its variable is unknown. It might be a good compromise in terms of robustness and convergence speed, as it should perform well on additively and non-additively separable functions.

2. Hybrid CMA-ES and HDE algorithm

In this section, we start with a general description of EA, a brief introduction to the CMA-ES and the HDE algorithms, and continue by describing the proposed hybridisation of the two methods (detailed descriptions can be found for CMA-ES in [18,19], for DE in [20,23] and HDE in [21]).

2.1. Evolutionary algorithms

A population of μ individuals, each representing a potential solution within the domain boundaries, is randomly chosen as a starting point. The population goes through three operators to evolve: recombination between individuals, random mutation of their alleles and selection of the fittest. One iteration of the strategy is a step from a population P^n to P^{n+1}, where n is the generation number, and can be written as

$$P^{n+1} := o\,pt_{\mathrm{EA}}(P^n) \tag{2}$$

The optimisation of P^n is defined by the operators *sel* (selection), *mut* (mutation) and *rec* (recombination) in the following way:

$$o\,pt_{\mathrm{EA}} := sel \circ (mut \circ rec)^{\lambda} \tag{3}$$

where λ corresponds to the number of new individuals (children).

According to the type of EA, a phase of adaptation of the parameters or migration of individuals might follow from their selection. The termination criterion for each iteration is met when the maximum number of function evaluations is reached. Since this is roughly proportional to the total computing time, we are able to define an (approximate) upper limit of time required for the optimisation process. When reached, the algorithm exits and returns the best individual.

2.2. Covariance matrix adaptation evolution strategy

Each individual in the population $P = \{\vec{a}_1, \vec{a}_2, \ldots, \vec{a}_\mu\}$ referred to by an index $k = 1, \ldots, \mu$ has a phenotype $\vec{a}_k = (\vec{x}_k, \vec{z}_k)$ with $\vec{x}_k, \vec{z}_k \in \mathbb{R}^n$, where $\vec{x}_k$ is the standard ES parameter vector and $\vec{z}_k$ is the associated standard deviation vector. Three matrices are needed for the algorithm: the covariance matrix $C \in \mathbb{R}^{n \times n}$, the eigenvector matrix of C named $B \in \mathbb{R}^{n \times n}$ and the diagonal matrix of the square rooted eigenvalues of C named $D \in \mathbb{R}^{n \times n}$. The μ individuals of the initial population are randomly defined ($\vec{x}_k$ are randomly chosen within the domain boundaries of f and $\vec{z}_k$ are set to the null vector). Matrix B is set to the identity matrix, the diagonal matrix D is set to represent the domain boundaries $D_{ii} = H_i - L_i, \forall i = 1, \ldots, n$. C is calculated as the product of BD and its transpose: $BD \cdot (BD)^t$.

2.2.1. Recombination

Using the global weighted intermediate recombination method in conjunction with a sorted population (the best individual is number 1, the worst is μ), λ identical children are created with a phenotype:

$$(\vec{x}_h)_i = \sum_{k=1}^{\mu} \omega_k \cdot (\vec{x}_k)_i, \quad \forall i = 1, \ldots, n \tag{4}$$

$$(\vec{z}_h)_i = 0, \quad \forall i = 1, \ldots, n \tag{5}$$

in which the individual index h goes from $(\mu + 1)$ to $(\mu + \lambda)$ and ω_k are the weights of the recombination, which are themselves parameters of the algorithm. In this study we take $\omega_k = (\log(\mu + 1) - \log(k)) / \sum_{l=1}^{\mu} (\log(\mu + 1) - \log(l))$ from [18], which gives more weight to the best individuals of the population.

2.2.2. Mutation

The main mechanism of the implemented operator is changing the allele values by adding random noise drawn from a normal distribution. The randomness from the normal distribution is stored in the individual phenotype and used in the adaptation phase. The mutation acts on each of the λ children with a modification of their

phenotype in the following order:

$$(\vec{z}_h)_i \leftarrow N_i^h(0,1), \quad \forall i = 1,\ldots,n$$
$$(\vec{x}_h)_i \leftarrow (\vec{x}_h)_i + \sigma_F \cdot \sum_{k=1}^{n} B_{ik} \cdot D_{kk} \cdot (\vec{z}_h)_k, \quad \forall i = 1,\ldots,n, \tag{6}$$

where $N_i^h(0,1)$ is a random number drawn from a normal distribution sampled anew for each element i of each individual $h = (\mu+1),\ldots,(\mu+\lambda)$ and the symbol $\leftarrow$ means that $(\vec{z}_h)_i$ and $(\vec{x}_h)_i$ will take the values on their RHS. The global step size $\sigma_F \in \mathbb{R}_+$ is a (problem-dependant) parameter of the algorithm.

2.2.3. Selection

Elitist selection is used to retain the μ best individuals of the λ children.

2.2.4. Adaptation

The global step size $\sigma_F \in \mathbb{R}_+$ is adapted using a "conjugate" evolution path $\vec{s} \in \mathbb{R}^n$, realised in the following order:

$$\vec{s} \leftarrow (1-c_s)\vec{s} + \sqrt{\mu_{eff} \cdot c_s(2-c_s)} \cdot B \cdot \sum_{k=1}^{\mu} \omega_k \vec{z}_k$$
$$\sigma_F \leftarrow \sigma_F \cdot exp\left(\left(\frac{s}{\overline{\chi_n}} - 1\right) \cdot \frac{c_s}{d_s}\right), \tag{7}$$

where $\overline{\chi_n} = \sqrt{n}(1 - 1/4n + 1/21n^2)$, $c_s = (\mu_{eff}+2)/(n+\mu_{eff}+3)$, $d_s = 1 + 2 \cdot max\left(0, \sqrt{(\mu_{eff}-1)/(n+1)} - 1\right) + c_s$, $\mu_{eff} = 1/\sum_{k=1}^{\mu} \omega_k^2$ and s is the vector norm of $\vec{s}$. The initial conjugate evolution path is $\vec{s} = \vec{0}$.

The covariance matrix $C \in \mathbb{R}^{n \times n}$ is adapted using the evolution path $\vec{c} \in \mathbb{R}^n$ in the following way:

$$\vec{c} \leftarrow (1-c_c)\vec{c} + H_s \cdot \sqrt{\mu_{eff} \cdot c_c(2-c_c)} \cdot BD \cdot \sum_{k=1}^{\mu} \omega_k \vec{z}_k$$
$$C \leftarrow (1-c_{cov})C + c_{cov} \cdot \frac{1}{\mu_{eff}} \cdot \vec{c} \cdot \vec{c}^t + c_{cov} \cdot \left(1 - \frac{1}{\mu_{eff}}\right) \cdot \sum_{k=1}^{\mu} (BD \cdot \vec{z}_k)(BD \cdot \vec{z}_k)^t \tag{8}$$

where $c_{cov} = 1/\mu_{eff} \cdot 2/(n+\sqrt{2})^2 + (1 - (1/\mu_{eff})) \cdot min(1, (2\mu_{eff} - 1)/((n+2)^2 + \mu_{eff}))$, $c_c = 4/(n+4)$, $H_s = 1$ if $s/\sqrt{1-(1-c_s)^{2(g+1)}} < (1.5 + (1/(n-0.5)))\overline{\chi_n}$ or 0 otherwise (the symbol g corresponds to the generation number). The initial evolution path is $\vec{c} = \vec{0}$.

Once adapted, the orthogonal matrix B and diagonal matrix D are obtained through principal component analysis of C (i.e. $C = BD^2B^t$).

2.3. The hybrid differential evolution algorithm

Following from [20,23,21], the individuals are coded with real-value representations. The population $P = \{\vec{a}_1, \vec{a}_2, \ldots, \vec{a}_{NP}\}$ is composed of NP individuals and their phenotype is given by $\vec{a}_k = (\vec{x}_k), \vec{x}_k \in \mathbb{R}^n$ where $\vec{x}_k$ is the parameter vector for individual $k = 1,\ldots,NP$. The initial population is randomly distributed in the domain of the function to optimise.

2.3.1. Recombination and mutation

For each member of the population $k = 1,\ldots,NP$ a trial individual $\vec{w}_k \in \mathbb{R}^n$ is generated according to

$$\vec{\omega}_k = \vec{\beta}_k + F \cdot \vec{\delta}_k, \tag{9}$$

where F is the differentiation constant (controls the amplification of the differentiation), $\vec{\beta}_k$ is the base vector and $\vec{\delta}_k$ the differentiation vector.

The differentiation strategy used in our study is *Rand*3 [23] where $\vec{\delta}_k = \vec{\xi}_1 - \vec{\xi}_2$ and $\vec{\beta}_k = \vec{\xi}_3$. $\{\vec{\xi}_1, \vec{\xi}_2, \vec{\xi}_3\}_k$ are randomly chosen individuals in the population sampled anew for each $k = 1,\ldots,NP$.

A crossover is then carried out between the trial and corresponding individual:

$$(\vec{\omega}_k)_i \leftarrow \begin{cases} (\vec{\omega}_k)_i, & \text{if } (rand_i(0,1) \geq C_r \text{ or } R = i) \\ (\vec{a}_k)_i, & \text{otherwise} \end{cases} \tag{10}$$

where R is a randomly selected allele number defined before the crossover (forces at least one allele to change), $rand_i(0,1)$ is a random number between 0 and 1 sampled anew for each allele i.

The resulting trial individuals are comprised in a set of NP candidates for the selection phase.

2.3.2. Selection

The best individual between the parent in the population and the corresponding trial is kept.

2.3.3. Migration

In order to reduce the population size and avoid stagnation in the region of a local optimum, Chang proposed a migration technique.[1] When the diversity ρ of the population is too small (i.e. $\rho < \epsilon_1$), all the individuals are modified according to the rule:

$$(\vec{x}_k)_i \leftarrow \begin{cases} (\vec{x}_k)_i + \rho_1 \cdot (L_i - (\vec{x}_b)_i), & \text{if } \rho_2 < \dfrac{(\vec{x}_k)_i - L_i}{H_i - L_i} \\ (\vec{x}_k)_i + \rho_1 \cdot (H_i - (\vec{x}_b)_i), & \text{otherwise} \end{cases} \tag{11}$$

where $i = 1,\ldots,n$, $\vec{x}_b$ is the best individual of the actual population, ρ_1 and ρ_2 are two random numbers chosen between 0 and 1 sampled anew for each element of each individual.

The diversity ρ is defined as follows:

$$\rho = \sum_{k=1}^{\mu} \sum_{i=1}^{n} \frac{\chi_{ki}}{n \cdot (\mu - 1)} \tag{12}$$

where

$$\chi_{ki} = \begin{cases} 1 & \text{if } \left|\dfrac{(\vec{x}_k)_i - (\vec{x}_b)_i}{(\vec{x}_b)_i}\right| > \epsilon_2 \text{ and } \left|(\vec{x}_k)_i - (\vec{x}_b)_i\right| > (\vec{\epsilon}_3)_i \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

with $\epsilon_2 \in \mathbb{R}_+$ and $\vec{\epsilon}_3 \in \mathbb{R}_+^n$ which are respectively the relative precision and the absolute precisions for the problem solved. We have introduced the parameter $\vec{\epsilon}_3$, which was not originally included in the HDE, to take into account the desired number of decimal places in the variables.

2.4. The hybrid algorithm (CMA-ES/HDE)

Fig. 1 depicts the schematic form of the proposed hybrid algorithm.

The HDE and CMA-ES operate in series. We distinguish two populations: popHDE and popCMA-ES that are associated with HDE and CMA-ES. We start with the CMA-ES on a random population for n_t steps (or generations), it feeds the best n_t individuals of each step to the population of HDE. The missing $(NP - n_t)$ individuals in popHDE are then randomly generated. We then continue with the HDE for n_t generations. We keep from the last generation of HDE the μ and $(NP - n_t)$ best individuals. If the termination criterion is met, the algorithm is stopped, otherwise it loops and continues with CMA-ES.

[1] Not to be confused with the migration phase of island distributed EAs [24].

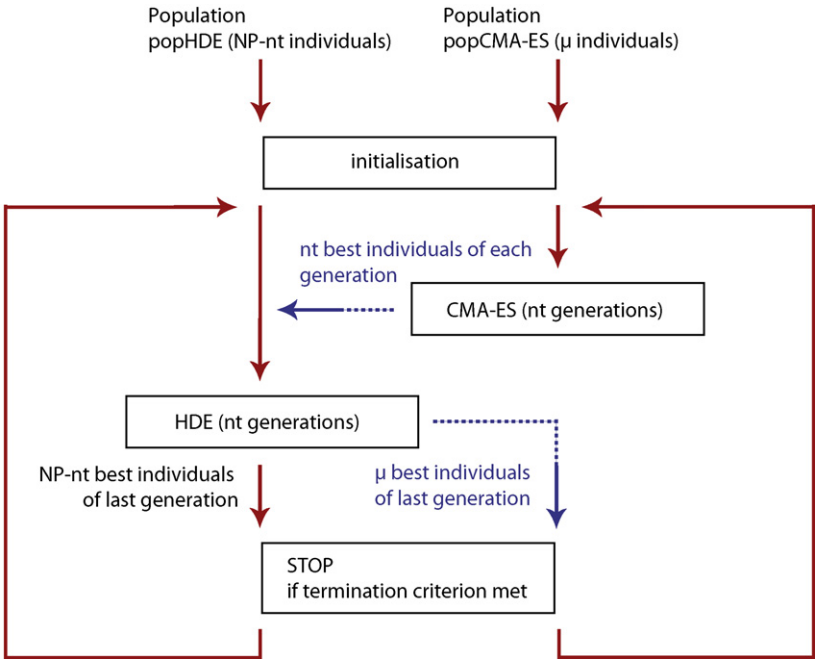

Fig. 1. The hybrid algorithm, a coupling of CMA-ES and HDE—two distinct populations popHDE and popCMA-ES go through evolution process (red solid lines) exchanging individuals (blue dashed lines). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of the article.)

3. Application examples

To illustrate the performance of the proposed hybrid method we have used two benchmark functions: those of Ackley and Rastrigin. The Ackley function is not additively separable unlike the Rastrigin function. A priori from [19] the CMA-ES alone should perform best on Ackley and the HDE alone on Rastrigin. We proceed by checking this and comparing the proposed hybrid performance to chosen methods: $(\mu/2_I, \lambda)$-ES, CSA-ES, CMA-ES, DE and HDE. We end the application with the analysis of a real-life problem: the maximisation of solar energy potential in urban areas.

3.1. Ackley function

The generalised Ackley function is defined in n dimensions by

$$f_n(\vec{x}) = -a\exp\left(-b\sqrt{\frac{1}{n}\sum_{i=1}^{n}x_i^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(c\cdot x_i)\right) + a + \exp(1) \quad (14)$$

where $\vec{x}\in\mathbb{R}^n$, $a=20$, $b=0.2$ and $c=2\pi$. Its domain is $-32.768 \le x_i \le 32.768, \forall i = 1,\ldots,n$. The global trend of this function is unimodal with a peak in $\vec{x}=\vec{0}$, and is perturbed with waves that creates many smaller peaks. We minimise the function f_n.

We have tested the Ackley function for $n=10$, repeated each case for 100 runs and compared results from the hybrid algorithm CMA-ES/HDE with DE, HDE, CSA-ES [16,17], CMA-ES and $(\mu/2_I, \lambda)$-ES [25]. Their parameters were chosen for a best performance (based on the Q-measure (Q_m)) after many experiments (available on request).

DE: $NP = 30$, $C_r = 0.1$, $F = 0.3$
HDE: $NP = 30$, $C_r = 0.1$, $F = 0.3$, $\epsilon_1 = 0.1$, $\epsilon_2 = 0.1$, $(\vec{\epsilon}_3)_{i=1,\ldots,n} = 1\times10^{-3}$
ES: $\sigma_F = 0.2$, $\mu = 15$, $\lambda = 100$, $K = 1.0$
CSA-ES and CMA-ES: $\sigma_F = 0.2$, $\mu = 5$, $\lambda = 10$
CMA-ES/HDE: combination of the parameters of CMA-ES and HDE

The results are summarised in Table 1. In this the first line contains the convergence measure C (the mean number of function evaluations to reach a fitness under 0.1) and its standard deviation. The second line contains the percentage of convergence P_C on a maximum of 80,000 function evaluations and the Q-measure which is the ratio of convergence measure by percentage of convergence. See [23, p. 84] for more details on performance measures.

We notice when looking at the results (Table 1) that as expected from [19] the CMA-ES performs better in Q-measure than the DE on this non-additively separable function. However, the DE is more robust than the CMA-ES as it achieves 100% of convergence.

The proposed hybrid algorithm benefits from both the advantages of the two algorithms. The HDE component brings the highest convergence percentage (100%) and the CMA-ES component a better performance in Q-measure than the HDE alone.

If we look closer at a typical run in Fig. 2, we notice that the CMA-ES component is more active in improving the population best fitness than the HDE component. The average number of

Table 1
Convergence comparison for the Ackley benchmark function.

	DE	HDE	ES	CSA-ES	CMA-ES	CMA-ES/HDE
C	3510 ± 151	3510 ± 151	3675 ± 201	1178 ± 137	862 ± 70	2695 ± 178
P_C (Q_m)	100% (3510)	100% (3510)	99% (3712)	99% (1190)	98% (880)	100% (2695)

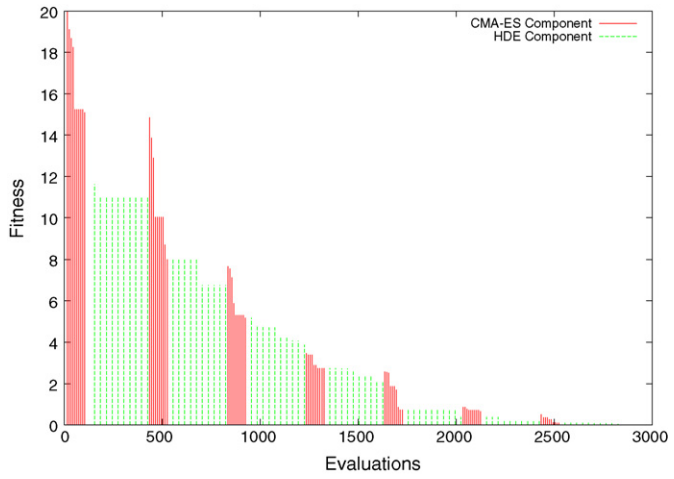

Fig. 2. A typical run of the CMA-ES/HDE algorithm on the Ackley function with dimension $n = 10$, best individual fitness versus number of function evaluations.

function evaluations needed by the CMA-ES component for the hybrid to converge to the global minimum is roughly the same as for the CMA-ES alone. The overall performance in Q-measure of the hybrid is worsened by the HDE component, but fortunately it provides a better convergence percentage.

After a first run of the hybrid algorithm, the CMA-ES component obtains the μ best individuals from the HDE, and performs a global intermediate recombination of those followed by mutation. This operation can result in worse individuals than the initial ones as the elitist selection phase forgets about the parents. This effect is very noticeable after the first HDE component run in Fig. 2: the best individual of the following first CMA-ES generation is worse than the last best one.

Finally, we have tested the performance of our algorithm against the dimension number n of the generalised Ackley function (Table 2). The CMA-ES population size is adapted according to: $(\mu = 2 + \lfloor 1.5 \cdot \log(n) \rfloor,\ \lambda = 4 + \lfloor 3 \cdot \log(n) \rfloor)$ [18] keeping the population size of HDE constant ($NP = 30$). The termination criterion is satisfied when the function returns a value under $1e-6$, and the precision parameters for HDE are set to: $\epsilon_2 = 0.1$, $(\vec{\epsilon}_3)_{i=1,\ldots,n} = 5e-7$. From this we conclude that our algorithm can adapt to reasonable increases in problem dimension for the Ackley function.

3.2. Rastrigin function

The generalised Rastrigin function is defined in n dimensions by

$$f_n(\vec{x}) = nA\sum_{i=1}^{n} x_i^2 - A\cos(\omega x_i) \tag{15}$$

where $\vec{x} \in \mathbb{R}^n$, $A = 10$, and $\omega = 2\pi$. Its domain is $-5.12 \le x_i \le 5.12$, $\forall i = 1, \ldots, n$ and the global minimum is in $f_n(\vec{0}) = 0$. This function is highly multi-modal with many sub-peaks increasing in intensity when approaching the global minimum. We minimise the function f_n.

We tested the Rastrigin function for $n = 10$. The parameters are the same as for the Ackley function, except for the $(\mu/2_I, \lambda)$-ES

Table 2
Convergence for Ackley benchmark function with different dimension values.

	n			
	10	20	30	40
C	7717 ± 264	12018 ± 279	15542 ± 306	18765 ± 269
P_C	100%	100%	100%	100%

[25] where the population was increased to (32,200) for performance. The population size of CSA-ES and CMA-ES should have been further increased for good convergence [19], however we decided to keep it small to facilitate a better performance of the hybrid CMA-ES/HDE. The corresponding results are summarised in Table 3.

Both CSA-ES and CMA-ES have too small population to converge to the global minimum. We also note that, thanks to its migration procedure, HDE is more robust than DE. As expected from Hansen and Kern's conclusion [19], the DE outperforms the CMA-ES on this additively separable function.

The hybrid appears in the statistics to be just slightly better in Q-measure than HDE alone. This fact is hard to explain as the working of EA is complex and hard to understand in details. However, we noticed in the results what could only be a small part of the explanation: when approaching the vicinity of the global minimum, the CMA-ES component of the hybrid takes the lead and provides a faster convergence to the global minimum than what the HDE component does. This effect is barely visible in Fig. 3 where two typical runs of our hybrid algorithm with the Rastrigin function are shown.

In contrast to the Ackley function, the HDE component of our hybrid is considerably more active than the CMA-ES component. The best and worst run difference in number of function evaluations to reach the global minimum gives an insight on the high standard deviation of the hundred runs. In the worst case, the migration procedure of the HDE component had to act two times to overcome local minima.

We have also tested the performance of our algorithm against the dimension number n of the generalised Rastrigin function. The same algorithm parameters were used as for the Ackley function. The results are summarised in Table 4. From this we notice that even with a small adaptation of the population size for the CMA-ES, our hybrid algorithm is robust and converges in all cases. However, the variability in convergence speed (see the standard deviation) becomes large when n increases. This is due to the migration phase in the HDE and the cost for 100% of convergence.

3.3. The optimisation of solar potential

The solar energy potential is defined as the total incident irradiation available from the sun during a period of interest. Through man history it has been a major concern as a free primary energy resource on the planet. Nowadays with the increasing pressure to diminish the risks of climate change due to combustion of fossil fuels, the interest in this renewable energy is even stronger. The shortwave irradiation can be converted to heat and electricity with adequate equipment, moreover its rational use in building design can reduce demands for heating and artificial

Table 3
Convergence comparison for Rastrigin benchmark function.

	DE	HDE	ES	CSA-ES	CMA-ES	CMA-ES/HDE
C	5949 ± 607	6543 ± 1897	17432 ± 3709	–	–	6255 ± 2020
P_C (Q_m)	95% (6262)	100% (6543)	32% (54475)	0%	0%	100% (6255)

Table 4
Convergence for Rastrigin benchmark function with different dimension values.

	n			
	10	20	30	40
C	9104 ± 3455	33844 ± 15047	71605 ± 29932	139031 ± 51237
P_C	100%	100%	100%	100%

lighting. In order to evaluate the utilisation potential of different energy conversion technologies, a simulation tool named PPF [26] has been developed. PPF is based upon RADIANCE [27], a backward ray-tracing software that can compute irradiation on surfaces. This tool was used in many previous studies [1–7]. In this paper, we take an irradiation calculation analogous to PPF for the determination of the solar energy potential. The purpose of this hypothetical study is to propose an arrangement of buildings on a flat ground that maximises their solar energy potential. The problem becomes particularly interesting when the number of buildings to arrange does not allow for a regular array (prime numbers for example) and the anisotropy of the sky leads to certain orientations being favoured over others. The period of interest chosen for the study is the winter season, when energy demands for heating and artificial lighting can be reduced. We assume that buildings can be fitted with shadowing devices in summer to avoid the negative impact of overheating. The first section deals with the parameterisation of the urban environment

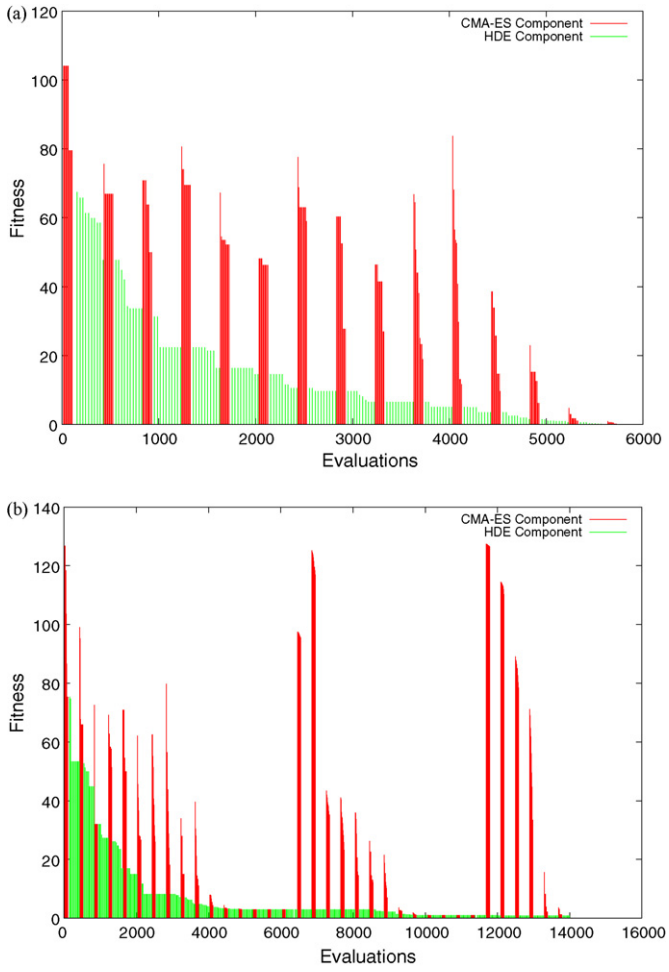

Fig. 3. Typical runs of the CMA-ES/HDE algorithm on the Rastrigin function with dimension $n = 10$. (a) Best run, migration procedure not necessary; (b) worst run, migration procedure needed two times to converge to the global minimum.

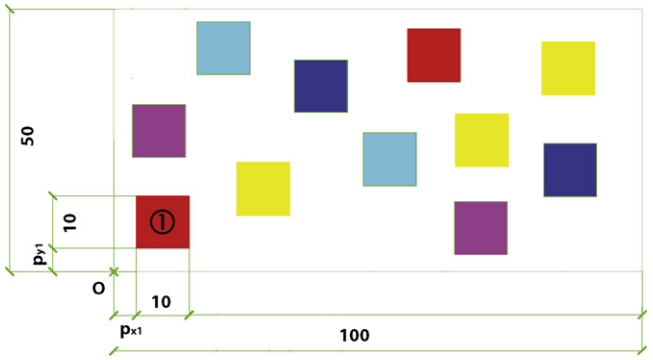

Fig. 4. The urban environment—11 buildings (view from top) disposed on the field, each building is parameterised by its position relative to the origin in Cartesian coordinates.

and the second section with solar potential determination using RADIANCE. The last section shows results of the solar potential maximisation obtained using our hybrid algorithm.

3.3.1. Parameterisation

The usual way of finding the best urban configuration(s) (for solar energy utilisation) is to subjectively define (and test) a subset of the possible combination of geometric parameters that influence the incident irradiation; selecting the configuration which leads to the best result [7]. Alternatively the entire set of possible configurations may be tested if the number of parameters to modify is small and the granularity of discrete values that these may take is crude. But when the parameter space to explore becomes large to very large, the task is preferably left to optimisation algorithms, as in our case. The parameter space is defined by the position of 11 buildings on a limited ground space of 50 m by 100 m. Each building is a parallelepiped of 10 m by 10 m on the ground and 20 m high. Fig. 4 shows a random configuration of eleven buildings viewed from the top. We use 22 variables to define the positions of the buildings on the ground:

$$\vec{x} = (p_{x1}, p_{y1}, \ldots, p_{x11}, p_{y11}), \tag{16}$$

where $p_{xi} \in [0, 90]$ is the position in x-coordinate and $p_{yi} \in [0, 40]$ in y-coordinate of the lower left corner of ith building.

The problem has no other constraints than the domain boundaries. In fact, it is a self-constrained problem. Indeed, when the buildings intersect they can intercept less rays from the sun

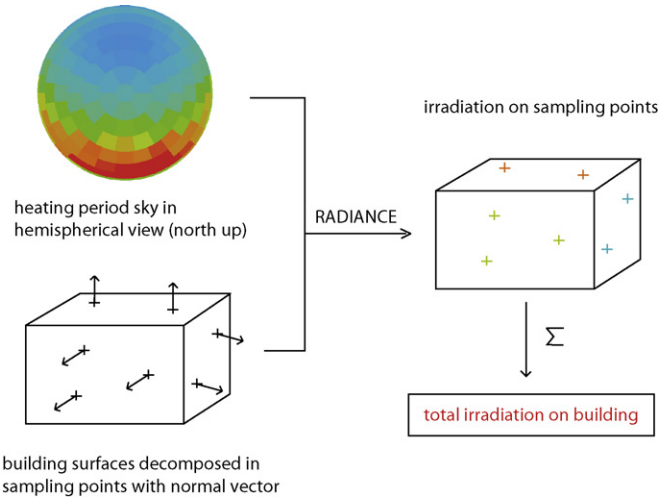

Fig. 5. The irradiation calculation using RADIANCE.

```
Total number of evaluations: 6016 Best fitness: 1.62907e+09
Simulation time: 177646 s Evaluation time: 342757 s
Acceleration factor: 1.92944
```

Fig. 6. The output of the hybrid algorithm.

and sky compared to when they are separated. By maximising the solar energy potential, the buildings tend to use at most the available space.

3.4. Solar potential determination

In order to calculate as precisely as possible the solar potential of hypothetical buildings, we have chosen to use the well-known ray tracing program RADIANCE [27]. This backward ray-tracing tool can simulate radiant energy exchanges throughout a scene of arbitrary geometric complexity, considering obstructions both to the sun and the sky due to the surrounding (urban) landscape, as well as reflected energy from this landscape. We input to RADIANCE a virtual geometrical model along with grid-points and normal vectors on each surface. The virtual geometrical model is in a RADIANCE specific text file format [28] and contains the surfaces of elements in the scene and the light sources. The grid-points act like virtual watt-meters, measuring energy arriving throughout a hemisphere whose zenith is in the direction indicated by the normal vector. The grid-points and normal vectors are stored in a text file for an easier handling by the rtrace program (the tracing core of RADIANCE), which computes the irradiation for each grid-point and saves it in an output file. This procedure is shown in schematic form in Fig. 5. In order to compute the irradiation (Wh/m^2) as opposed to irradiance (W/m^2), we take a cumulative sky as in [29] for the period of interest (heating period) with units $Wh/(sr\ m^2)$. This procedure allows us to simulate the cumulative irradiation throughout a given period in just one RADIANCE simulation. The product of this irradiation (Wh/m^2) and the surface area covered by the grid point (m^2), for the whole set of grid points, gives us the total irradiation received by a building (Wh).

3.5. Results

Fig. 6 shows the output of our algorithm. We have set the termination criterion to be 6000 function evaluations. Since we avoided the interruption of the algorithm part way through a component iteration, it was ceased after 6016 evaluations. Each evaluation takes approximatively 57 s on an Intel Centrino Duo 2 GHz equipped with 1 GB of RAM; due to the total computing time needed only one run of the algorithm was accomplished. The multi-threading programming style allowed for a mean 1.92 evaluations during the time of one, which reduced the total computing time to about 49 h. Fig. 7 shows the evolution of the best candidate with number of function evaluations. Looking more closely at this we note that the best individual appeared after 5836 evaluations. The CMA-ES component was very helpful at the beginning providing a quick exploration of the search space, the relay was then taken by the HDE component which provided the best individual.

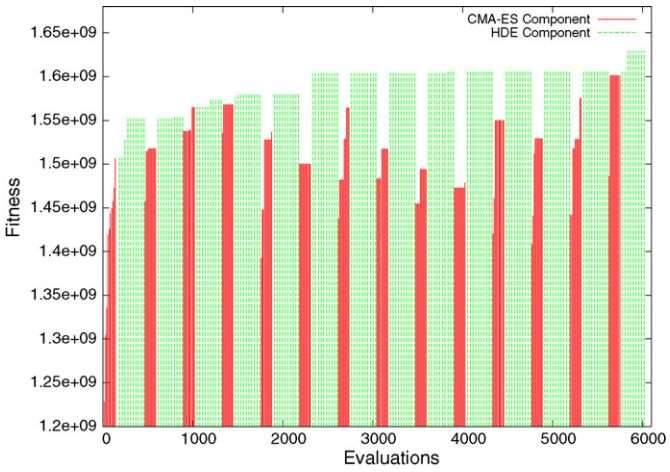

Fig. 7. The improvements in solar energy potential of the urban configuration versus the evaluations number.

The solar energy potential of the best candidate (see Fig. 8 for a RADIANCE image) is about 1.63 GWh during the heating period relating to a cumulative sky for Basel (Switzerland).

For comparison purposes we have solved the same problem using the advanced optimisation software MOO [30,31] which runs under MATLAB. With this software configuration we predict an equivalent solar energy potential (1.63 GWh) and placement of

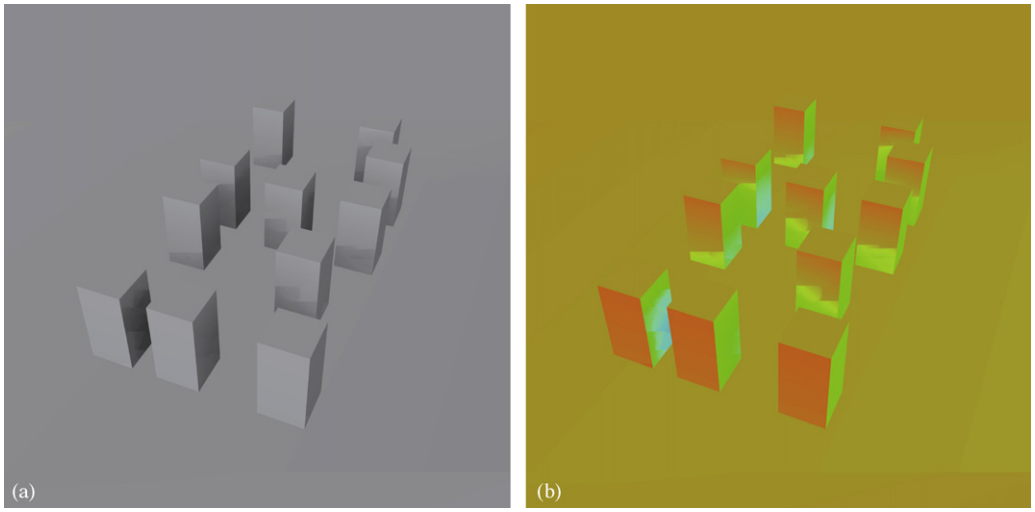

Fig. 8. The results of the optimisation of the solar energy potential for the placement of 11 buildings in a rectangular ground (color online). (a) Grey-scale image generated with RADIANCE; (b) rendering with surface falsecolored according to incident irradiation. Red corresponds to 0.08 GWh/m^2. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of the article.)

buildings after 5200 evaluations; achieved after 65 h of simulation. Although the average evaluation time of 45 s is less than that of our hybrid algorithm (due to computer resources being more available at partial processor load) the acceleration resulting from multi-threading provides a total computation time saving of about 25%.

4. Conclusion

In this paper, a hybrid CMA-ES/HDE optimisation algorithm is proposed and compared with other selected methods (DE, HDE, ES, CSA-ES and CMA-ES) on two standard benchmark functions: Ackley and Rastrigin. From our statistical results the method exhibited the advantages of the two hybridised algorithms; the HDE component brought robustness in finding the global minimum and the CMA-ES component a faster convergence than HDE alone.

Summarising results, we showed that the proposed hybrid is robust on the two standard benchmark functions tested. In actual optimisation applications robustness is an important issue, thereby if the function dependence on its parameters has similarities with Ackley or Rastrigin, the proposed hybrid should be robust in finding a good solution. Nevertheless, we do not know if it would also be the case for functions with similarities to other benchmark functions. The application of the hybrid to a real-life problem of solar energy potential maximisation showed good results.

The proposed hybrid algorithm has been developed within the context of research to simulate and optimise the sustainability of urban developments. During this initial stage we apply the algorithm to optimise the geometric form of a hypothetical urban development according to a simple fitness function—solar radiation availability. In the future we will apply this approach to a problem of considerably greater complexity, both in terms of the parameter space under investigation (geometry, construction and occupation of buildings, energy supply technologies, etc.) and the function to be optimised (e.g. cost, energy consumption, and CO_2 emissions).

Acknowledgements

The financial support 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. The authors would also like to thank Prof. Marco Tomassini for several fruitful discussions.

References

[1] M. Montavon, J.-L. Scartezzini, R. Compagnon, Comparison of the Solar Energy Utilisation Potential of Different Urban Environments, PLEA2004, Eindhoven, The Netherlands, 2004.
[2] M. Montavon, J.-L. Scartezzini, R. Compagnon, Solar Energy Utilization Potential of Three Different Swiss Urban Sites, Energie und Umweltforschung im Bauwesen, Zurich, 2004.
[3] J.-L. Scartezzini, M. Montavon, R. Compagnon, Computer Evaluation of the Solar Energy Potential in an Urban Environment, EuroSun, Bologna, 2002.
[4] K. Steemers, D. Raydan, C. Ratti, D. Robinson, Precis: assessing the potential for renewable energy in cities, Final report, Cambridge, 2000
[5] M. Montavon, K. Steemers, V. Cheng, R. Compagnon, 'la ville radieuse' by le Corbusier Once Again a Case Study, PLEA, Geneva, Switzerland, 2006.
[6] J.H. Kämpf, M. Montavon, J. Bunyesc, R. Bolliger, D. Robinson, Optimisation of Buildings Daylight Availability, CISBAT, Lausanne, Switzerland, 2007.
[7] V. Cheng, K. Steemers, M. Montavon, R. Compagnon, Urban Form, Density and Solar Potential, PLEA, Geneva, Switzerland, 2006.
[8] D.B. Fogel, Evolutionary Computation: Toward a New Philosophy of Machine Intelligence, Wiley, 2006.
[9] D.E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley, 1992.
[10] M. Mitchell, An Introduction to Genetic Algorithms, The MIT Press, 1998.
[11] J.H. Holland, Adaptation in Natural and Artificial Systems, The University of Michigan Press, Ann Arbor, MI, 1975.
[12] L.J. Fogel, A.J. Owens, M.J. Walsh, Artificial Intelligence through Simulated Evolution, John Wiley and Sons, New York, 1966.
[13] I. Rechenberg, Evolutionsstrategie: Optimierung technischer Systeme nach Principen der biologischen Evolution, Fromman-Holzboog Verlag, Stuttgart, 1973.
[14] J.R. Koza, Genetic Programming, The MIT Press, Cambridge, MA, 1992.
[15] H.-P. Schwefel, Evolution and Optimum Seeking, Wiley-Interscience, 1995.
[16] H.-G. Beyer, H.-P. Schwefel, Evolution strategies: a comprehensive introduction, Natural Computing 1 (2002) 3–52.
[17] V.A. Dirk, M. Alexander, Hierarchically organised evolution strategies on the parabolic ridge, in: GECCO'06, Seattle, WA, USA, 2006.
[18] N. Hansen, A. Ostermeier, Completely derandomized self-adaptation in evolution strategies, Evolutionary Computation 9 (2) (2001) 159–195.
[19] N. Hansen, S. Kern, Evaluating the CMA evolution strategy on multimodal test functions, in: Parallel Problem Solving from Nature, PPSN 2004, Springer-Verlag, 2004.
[20] R. Storn, K. Price, Minimizing the real functions of the ICEC'96 contest by differential evolution, in: Proceedings of the IEEE Conference on Evolutionary Computation, 1996, pp. 842–844.
[21] C.F. Chang, J.J. Wong, J.P. Chiou, C.T. Su, Robust searching hybrid differential evolution method for optimal reactive power planning in large-scale distribution systems, Electric Power Systems Research 77 (5/6) (2007) 430–437.
[22] A. Nobakhti, H. Wang, A simple self-adaptive differential evolution algorithm with application on the alstom gasifier, Applied Soft Computing Journal 8 (1) (2008) 350–370.
[23] V. Feoktistov, Differential Evolution Search of Solutions, Springer, 2006.
[24] M. Tomassini, Spatially Structured Evolutionary Algorithms: Artificial Evolution in Space and Time, Natural Computing, Springer-Verlag New York Inc., 2005, p. 1051297.
[25] D. Quagliarella, J. Périaux, C. Poloni, G. Winter, Genetic algorithms and evolution strategies in engineering and computer science, in: Recent Advances and Industrial Applications, John Wiley and Sons, 1998.
[26] R. Compagnon, Precis: assessing the potential for renewable energy in cities, annexe3: solar and daylight availability in urban areas, Tech. rep., University of Applied Sciences, July 2000.
[27] G. Ward Larson, R. Shakespeare, Rendering with Radiance: The Art and Science of Lighting Visualization, Morgan-Kaufmann, San Francisco, 1998.
[28] G. W. Larson, Radiance file formats, http://radsite.lbl.gov/radiance/refer/filefmts.pdf.
[29] D. Robinson, A. Stone, Irradiation Modelling Made Simple: The Cumulative Sky Approach and its Applications, PLEA2004, Eindhoven, The Netherlands, 2004.
[30] G.B. Leyland, Multi-objective optimisation applied to industrial energy problems, Ph.D. thesis, EPFL, 2002.
[31] A. Molyneaux, A practical evolutionary method for the multi-objective optimisation of complex energy systems, including vehicle drivetrains, Ph.D. thesis, EPFL, 2002.