Studies in Computational Intelligence 829

Christophe Sabourin
Juan Julian Merelo
Kurosh Madani
Kevin Warwick *Editors*

Computational Intelligence

9th International Joint Conference, IJCCI 2017 Funchal-Madeira, Portugal, November 1–3, 2017 Revised Selected Papers

Springer

Optimizing Costs and Quality of Interior Lighting by Genetic Algorithm

Alice Plebe, Vincenzo Cutello and Mario Pavone

Abstract This paper proposes the use of multi-objective optimization to help in the design of interior lighting. The optimization provides an approximation of the inverse lighting problem, the determination of potential light sources satisfying a set of given illumination requirements, for which there are no analytic solutions in real instances. In order to find acceptable solutions we use the metaphor of genetic evolution, where individuals are lists of possible light sources, their positions and lighting levels. We group the many, and often not explicit, requirements for a good lighting, into two competing groups, pertaining to the quality and the costs of a lighting solution. The cost group includes both energy consumption and the electrical wiring required for the light installation. Objectives inside each group are blended with weights, and the two groups are treated as multi-objectives. The architectural space to be lighted is reproduced with 3D graphic software Blender, used to simulate the effect of illumination. The final Pareto set resulting from the genetic algorithm is further processed with clustering, in order to extract a very small set of candidate solutions, to be evaluated by the architect.

Keywords Lighting design · Genetic algorithm · Decision maker · Blender

1 Introduction

In this paper we explore the possibility to support the task of designing light in architecture, by a formulation of a multi-objective optimization problem, to be solved using evolutionary algorithms, followed by a clustering technique to reduce the final

A. Plebe
Department of Information Engineering and Computer Science,
University of Trento, Trento, Italy
e-mail: alice.plebe@unitn.it

V. Cutello · M. Pavone (✉)
Department of Mathematics and Computer Science, University of Catania, Catania, Italy
e-mail: mpavone@dmi.unict.it

V. Cutello
e-mail: cutello@dmi.unict.it

C. Sabourin et al. (eds.), *Computational Intelligence*, Studies
in Computational Intelligence 829, https://doi.org/10.1007/978-3-030-16469-0_2

Pareto set. The design of interior lighting is the crucial and complex process of integrating luminaries into the fabric of architecture [1, 2]. Humans, like most primates and several mammals, are predominantly visual creatures. Forms of artificial lighting have been introduced since antiquity, to make visual perception possible when and where sunlight lacks [3]. In most of the contemporary world a considerable amount of time is spent indoors and with insufficient daylight illumination. Our ability to move toward interior environments, orientate ourselves, and go about our business relying on the perceptions we form of the surrounding objects, reckons on the level and quality of ambient illumination. Contemporary lighting design has the goal of selecting the lighting equipment and their placement in the interior environment that result in a comfortable and pleasant visual experience. The design process should take into account several aspects, such as the type of occupants and the type of activities in the given space, or the interior surface finishes and furnishings.

Unlike most multi-objective optimization problem in other domains, such as industrial engineering [4], in lighting design there is rarely an explicit formulation of the requirements for the optimal solution. However, it is often possible to identify at least two types of objectives for a lighting system: on one side the properties that enhance the quality and the pleasantness of the interior light, on the other side the costs involved in providing the chosen lighting solution. The first component of costs is related to the realization of the lighting plant. In addition, in the last decades increasing attention has been paid to the issue of energy savings. In U.S. the energy consumed for lighting accounts for about 30% of the total energy consumed by commercial buildings [5], and in the European Union the yearly consumption is over 170 TWh [6]. Therefore, the concept of *sustainable* lighting design has become central in architectural strategies [7].

In the approach here proposed we group objectives that belong to the quality or the cost categories by weighted summations, and then treat the two combined values as contrasting multi-objectives. It should be highlighted that lighting design is a process that encompasses strict physical evaluations with aesthetic and stylistic evaluations, upon the premise that the lighting condition has enormous emotional, psychological, and physiological impact on people. It is not possible, nor even desirable, for a computerized optimization to provide a single, deterministic solution. The great help of a system like the one here proposed would be to shortlist a manageable small set of lighting solution for the architect to be evaluated with her subjective expertise and sensibility. As with most real problem solved with multi-objective optimization, the final Pareto set is far too large for a visual evaluation. We apply a technique to reduce it, even in the absence of known preferences of the decision maker, typically assumed in literature [8, 9]. We partition the whole Pareto set into a small number of clusters, and in each cluster we pick a representative solution.

The algorithm here presented is an extension of an earlier model [10], based on the combination of the 3D graphic software Blender and a genetic algorithm for solving the multi-objective optimization problem. Blender provides the rendering engine for a physical simulation of the effect of a lighting solution on a model of the interior environment. In the current version the problem objectives have been refined, with three components in the quality group and two components in the group related to the cost of the lighting solution. In the latter, we compute an estimate of the realization cost of each candidate lighting solution.

2 The Lighting Problem and the Optimization Solution

The design of interior lighting is the crucial and complex process of integrating luminaries into the fabric of architecture [1, 2]. The goal is to select the lighting equipment and their placement in the interior environment that result in a comfortable and pleasant visual experience. The design process should take into account several aspects, such as the type of occupants and the type of activities in the given space, or the interior surface finishes and furnishings. Since the discovery of the electric light system by Thomas Edison in 1879, lighting design has experienced several significant revolutions, such as fluorescent lamps in 1938 and, more recently, solid-state lighting. Traditionally, illumination design has been seen as a blend of art and practice, where all the challenges are left to the creativity and the experience of the design architect. Given the aesthetic nature of the task, lighting design may seem difficult to formally model. Nevertheless, in the last years the design process has been increasingly considered as a mathematical and physical problem to be solved with optimization techniques. A well established aids offered by computational tools to the designer is by photorealistic architectural rendering, simulating in computer graphics the effect of a lighting solution on a model of the interior environment [11]. Mathematically, this is the solution of the *direct lighting problem* that is the computation of radiance distribution in an environment that is completely known a priori, including its lighting parameters. The drawback of adopting direct lighting as the only aiding tool is that, if the achieved illumination is not satisfactory, it is not easy to infer which modifications to the current solution may lead to improvements. Very likely, the final solution chosen by the designer over a collection of trials will be far from optimal. A more effective assistance would be given by computational tools implementing the *inverse lighting problem* [12–15]: the determination of potential light sources satisfying a set of given illumination requirements, for a pre-defined interior space. For this problem there are no reliable analytic solutions, even for simple geometries, One of the earliest attempt to solve this problem with optimization [13] was based on the Broyden-Fletcher-Goldfarb-Shanno method [16] applied of the Hessian of the matrix derived by the objectives of lighting uniformity. The limitation of this class of methods is that the dependency from gradients leads easily to poor local minima. In very simple cases constrained least squares based optimization may work, for example when the position of the lights are fixed, and only their intensities can be varied [14].

A system facing the inverse lighting problem must provide a virtual environment able to accurately reproduce the architectural space and its spectral reflectometric properties. Moreover, a physical simulation platform must be considered as well for correct illumination calculation in sample points of the architectural space.

Several different tools can be considered for this purpose. Lightsolve [17] is an interactive dedicated environment for daylight design, with a performance-driven decision support system, however the system lacks a detailed architectural reproduction, and the inclusion of interior furniture is difficult to manage. In the work of [18] the 3D models of building facades are obtained with the simple modelling tool Google SketchUp, which offers a quick and easy way to outline an architectural space, but resulting in a low level of realism. Conversely, the popular software Radiance, widely used in the field of optimal lighting design [11, 19, 20], consists of a sophisticated physically-correct rendering engine for illumination calculation, and it allows architectural spaces reproduction at arbitrary levels of detail. Nevertheless, it is a non-interactive system composed of a collection of command-line programs, and all architectural specifications have to be coded into configuration files.

In this paper we adopt the 3D graphic software Blender as a unified solution to the two requirements stated above. Firstly, Blender is the most comprehensive open-source 3D computer graphic tool available. It is particularly suitable for modelling architectural interiors, with the possibility of importing components from CAD files. Secondly, Blender provides a physically-based rendering engine, able to exhaustively evaluate lighting configurations needed for solving the inverse lighting problem. Moreover, Blender embeds a Python interpreter, which can run scripts supplied by the user, in order to extend its functionalities. Thanks to its intrinsic versatility, Blender has already been applied to a number of different problems, from the medical field [21] to industrial applications [22], and the inverse lighting problem itself [10].

2.1 Non-dominated Sorting Genetic Algorithm

Due to the clashing of the multiple factors involved in interior lighting design, the resulting problem is multi-objective in nature. In contrast to single-objective optimization problems, where there is usually a single global minimum solution to be found, the goal of multi-objective optimization is to determine the set of best tradeoffs between all the conflicting criteria, namely the Pareto-optimal set. Genetic algorithms are a popular class of computational models, which have been extensively applied in various multi-objective optimization domains over the last decades. As the name suggests, genetics algorithms mimic the working principles of natural genetics and natural selection to construct robust search algorithms that require minimal problem information. The algorithm structure is borrowed from the sexual genetic reproduction process, divided into three fundamental phases: selection (or reproduction), mutation and crossover. Starting from a random population of solutions, the algorithm iteratively computes fitness values for each in order to identify the best solutions and to converge to a Pareto-optimal set. The selection operator is used to promote the best individuals in the population, by duplicating good solutions and discarding the bad ones, while it keeps the population size constant. Crossover and mutation operators perform the creation of new solutions. The first randomly picks two solutions from the mating pool, and exchanges some portion between the two chromosomes to create two new solutions. Afterwards, the mutation operator introduces diversity in the populations by randomly mutating the chromosomes obtained after crossover.

One of the key working principles of genetic algorithms is the chromosomal representation of a solution. The algorithm works with a coding of decision variables, instead of the variable themselves, and choosing the right representation scheme is crucial to its performance [23]. The most traditional approach is to code the decision variables in a binary string of fixed length, which is a natural translation of real-life genetic chromosomes. Such strings are directly manipulated by the genetic operators, crossover and mutation, to obtain a new (and hopefully better) set of individuals. Another well established method is the floating-point representation of chromosomes, where each solution is coded as a vector of floating point numbers, and crossover and mutation operators are adapted to handle real parameter values. For the algorithm presented in this work, we adopted a novel chromosomal representation of solutions, specifically tailored for lighting design optimization. As will be further described in Sect. 3.1, each individual represents a possible illumination configuration, and it is coded as an ordered set of variable length containing lamp specifications. Special operators of crossover and mutation are implemented to handle this peculiar chromosomal representation.

The specific genetic algorithm adopted in the present paper is the Non-dominated Sorting Genetic Algorithm II (NSGA-II), introduced by Deb et al. [24], an elitist multi-objective genetic algorithm that performs well with real world problems, producing Pareto-optimal solutions to the optimization problem. The elitist approach favours the best solutions of a population by giving them an opportunity to be directly carried over to the next generation. This strategy ensures that the best fitness values do not deteriorate during the evolution, and it enhances the probability of creating better offspring.

The elitism is integrated in the algorithm by selecting the next-generation population of size N among the best individuals from the offspring and the parent population combined together (size $2N$). This selection strategy, named *crowded tournament selection*, takes into account two criteria: the non-domination and the crowding distance of the individuals. The first is the non-domination rank of the solution in the population, and it is used to classify the entire $2N$ population into non-dominated fronts. The second criterion is a measure of the search space around the solution, which is not occupied by any other solution in the population. Giving preference to solutions that are less crowded (with larger crowding distances) ensures a better spread among the solutions during the evolution. These conditions make sure that non-dominated individuals belonging to a high rank front and residing in a less crowded area are selected to reproduce more than others. The result of the algorithm is the set of non-dominated solutions of the whole final population, namely the Pareto front.

3 The Proposed Model

The algorithm presented in this paper has been implemented in the form of a Blender script, structured in four groups of Python modules. The first group of modules, which rely on Blenders modelling features, performs the set-up of the simulation environment. The architectural interior scene of interest is represented inside the computer graphics software by means of geometric meshes and material shaders. The room structure (walls, floors, ceiling) and its furnishings are defined by the meshes, while colours, textures and reflectivity properties of the objects are specified through the shaders. Within the definition of the 3D model, the user has to provide a grid of points on the ceiling and the walls corresponding to the feasible set of coordinates for lamp positioning. This step is required because, depending on the room design, there might be some areas where the lamp placement is not allowed, for example in presence of windows, pillars, or supporting beams.

The evaluation of light quality in the 3D interior space is achieved performing individual lighting measurements over some supporting elements, called *samplers*, composed by surfaces with plain materials, which are introduced in the scene by the second group of python modules. The resulting rendered images are stored in HDR format, in order to preserve all the information of the dynamic range. Their pixel values are used, in the third group of modules, by the genetic algorithm to compute the actual fitness values of a solution, as will be detailed in Sect. 3.1. After evaluating the entire current population and selecting the mating pool, the genetic operators of crossover and mutation are applied to generate the offspring. The operators are specifically implemented for the presented case problem, as further described in Sect. 3.2, with the support of an evolutionary computation python framework named DEAP [25], which allows to freely customizing any component of the genetic algorithm workflow. At the end of the evolution, the obtained result is the Pareto front of the final population, namely the set of non-dominated solutions, each one of them representing an optimal lighting configuration for the given interior environment. A decision maker is then used to select a small number of best representative solutions, as will be described in Sect. 3.3. The Fig. 1 shows a flowchart representing the overall optimization process.

3.1 Definition of the Objectives

We can first split the requirements of our problem into two different overall goals: on the one hand the achievement of the highest quality of the lighting for the interior space for which it has to be designed, on the other hand the minimization of lighting costs. The two groups of objectives are clearly contrasting: it is easier to achieve a uniform and well adequate level of illumination with an expensive illumination plant. The objective functions related with the quality of the lighting are computed using the rendering of a solution $\mathcal{S}$ performed by Blender, and measuring the illumination

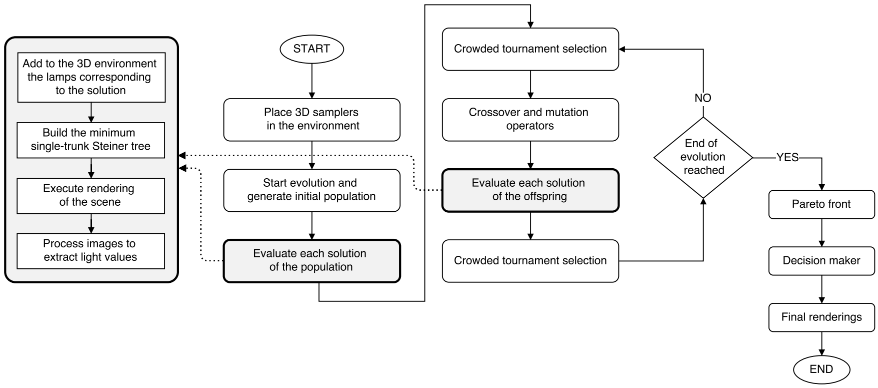

Fig. 1 Flowchart summarizing the proposed optimization model

on a set of samplers placed in the interior space. A solution $\mathcal{S}$ is, similarly to [10], the coding for a set of lamps with their specifications and placement:

$$\mathcal{S} = \langle \mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_L \rangle, \tag{1}$$

$$\mathcal{L} = \langle d, \{C, W\}, \mathbf{v}, l, w, k \rangle. \tag{2}$$

The genetic code of a solution $\mathcal{S}$ is an ordered set of lamp descriptions $\mathcal{L}$, in which d is a code identifying the type of commercial lamp, the second parameters specify the type of placement: C for ceiling and W for wall. The vector $\mathbf{v}$ specifies the 3D coordinates of the lamp placement, l is the intensity of the lamp in lumen, w its electrical efficiency in lumen/watt, and k the color temperature in kelvin degrees. The many possible combinations of parameters in a lamp description are restricted to a feasible subset accordingly to a list of predefined lamp definitions. The lamp code d specifies a real commercial light fixture, and the possible combinations of intensity l, consumption w, color temperature k and type of placement are extracted from the specification sheets provided by the manufacture.

We define $\mathcal{P}$ the ordered set of vectors $\mathbf{p}_i$, each of them made by N pixel values measured on one of the M samplers in the scene:

$$\mathcal{P}(\mathcal{S}) = \langle \mathbf{p}_1(\mathcal{S}), \mathbf{p}_2(\mathcal{S}), \ldots \mathbf{p}_M(\mathcal{S}) \rangle. \tag{3}$$

The desired level of illumination in the environment is specified with the target value $\hat{t}$, and in each sampler the deviation from the target is averaged:

$$\Delta\mathcal{P}(\mathcal{S}) = \left\{ \frac{1}{N} \sum_{p \in \mathbf{p}} |p - \hat{t}|,\ \mathbf{p} \in \mathcal{P}(\mathcal{S}) \right\}. \tag{4}$$

The quality of the lighting solution $\mathcal{S}$ is evaluated with the following three objective functions:

$$q_1(\mathcal{S}) = \frac{1}{M} \sum_{d \in \Delta\mathcal{P}(\mathcal{S})} d, \tag{5}$$

$$q_2(\mathcal{S}) = \max \Delta\mathcal{P}(\mathcal{S}), \tag{6}$$

$$q_3(\mathcal{S}) = \sqrt{\frac{1}{NM} \sum_{\mathbf{p} \in \mathcal{P}(\mathcal{S})} \sum_{p \in \mathbf{p}} (p - \overline{p})^2}, \tag{7}$$

where $\overline{s}$ is the average value of the pixels in the sampler. The first two functions, in Eqs. (5) and (6), evaluate the compliance with the target level of light, respectively in the average and in the worst case among the samplers. Function q_3 in Eq. (7) is an evaluation of the overall uniformity of lighting. Treating those three functions as separate fitness in multi-objective optimization would be incorrect, because they are not conflicting. It can be easily verified in the limit case of an individual $\hat{\mathcal{S}}$ that illuminates all samplers exactly at target level $\hat{t}$, we obtain $q_1(\hat{\mathcal{S}}) = q_2(\hat{\mathcal{S}}) = q_3(\hat{\mathcal{S}}) = 0$. Therefore, the fitness function for lighting quality is combined from the three objective functions:

$$f_q(\mathcal{S}) = w_1^{(q)} q_1(\mathcal{S}) + w_2^{(q)} q_2(\mathcal{S}) + w_3^{(q)} q_3(\mathcal{S}), \tag{8}$$

where $w_1^{(q)} + w_2^{(q)} + w_3^{(q)} = 1$.

The second group of objective functions to minimize are related to the cost of the lighting solution, both in term of the initial cost for its realization, and the running cost when the lights are switched on.

The realization cost is dominated by the electrical wiring, and its dependence on the solution $\mathcal{S}$ is related to the placement of the lamps. We compute an estimate of this dependency using an approximate solution of the *rectilinear Steiner problem*, appropriate for the wiring pattern adopted in houses. The general *Steiner problem* asks for a minimum spanning networks connecting a given set of points, allowing for the introduction of new auxiliary points so that a spanning network of all the points will be shorter than otherwise possible [26]. In a *rectilinear Steiner tree* only horizontal or vertical line segments in a plane can connect the points [27], a situation of great importance in VLSI design [28]. Note that an efficient minimum spanning computation is out of the scope, for the purpose of evaluating the dependency of the installation cost on the lighting solution a rough approximation is enough, therefore we adopted a simple heuristic, called *refined single-trunk tree* [29]. In the "simple" single-trunk tree it is assumed that there is a main trunk that goes horizontally or vertically, and all points are connected with stems orthogonal to the trunk. In the refined version, for each point it is checked if it is shorter to connect it to the trunk, or to the nearest stem connecting another point to the trunk. If the latter holds, the point is connected to this stem. The Fig. 2 shows the different structures resulting from the two approaches in the computation of the single-trunk Steiner tree, from an example of lighting configuration.

For simplicity's sake, let $\mathbf{v}_{\mathcal{L}_i} = [x_i, y_i]^{\mathrm{T}}$ be the 2D position of the lamp $\mathcal{L}_i \in \mathcal{S}$ on the ceiling plane, and y_0 the coordinate of the trunk, supposed to run horizontally

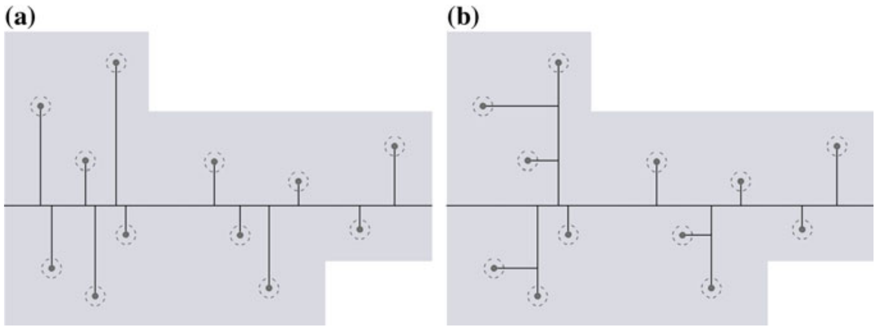

Fig. 2 Difference between a simple single-trunk Steiner tree (**a**) and a refined single-trunk Steiner tree (**b**), connecting all the lamps in a room

all across the room. As mentioned above, the exact computation of the optimal tree is not the aim of this work, therefore all the lamps can be considered on the same 2D plane. The contribution of lamp $\mathcal{L}_i$ to the length of the approximate rectilinear Steiner tree is the following:

$$\ell\left(\mathcal{L}_i\right) = \begin{cases} y_i - y_0 & \text{if } \mathbf{v}_{\mathcal{L}_i} \text{ closer to the trunk,} \\ x_i - x_v & \text{if } \mathbf{v}_{\mathcal{L}_i} \text{ closer to the vertical stem connecting } \mathbf{v}_{\mathcal{L}_v}, \\ y_i - y_h & \text{if } \mathbf{v}_{\mathcal{L}_i} \text{ closer to the horizontal stem connecting } \mathbf{v}_{\mathcal{L}_h}. \end{cases} \tag{9}$$

The cost function derived by the electrical wiring is computed cumulating all stem lengths:

$$c_1\left(\mathcal{S}\right) = \frac{1}{\sqrt{A}} \sum_{\mathcal{L} \in \mathcal{S}} \ell\left(\mathcal{L}\right), \tag{10}$$

where A is the area in m^2 of the interior environment to be lighted.

Energy consumption represents the second objective in the group of costs to minimize, and is quantified as the overall power consumption of the lamps (measured in Watt) divided by the volume of the room:

$$c_2\left(\mathcal{S}\right) = \frac{1}{V} \sum_{\mathcal{L} \in \mathcal{S}} C_{\mathcal{L}}, \tag{11}$$

where $C_{\mathcal{L}}$ is the amount of Watts consumed by the lamp $\mathcal{L}$ of the individual $\mathcal{S}$, V the volume of the interior space in m^3.

The fitness function related to costs is the weighted sum of the installation cost and the energy consumption costs:

$$f_c\left(\mathcal{S}\right) = w_1^{(c)} c_1\left(\mathcal{S}\right) + w_2^{(c)} c_2\left(\mathcal{S}\right), \tag{12}$$

where $w_1^{(c)} + w_2^{(c)} = 1$.

Finally, the bi-dimensional fitness function used in the genetic algorithm is the following:

$$\mathbf{f}(\mathcal{S}) = \begin{bmatrix} f_q(\mathcal{S}) \\ f_c(\mathcal{S}) \end{bmatrix}. \tag{13}$$

3.2 Genetic Operations

At each step t of evolution, there is a population $\mathcal{G}^{(t)} = \{\mathcal{S}_i\}$, which elements are individuals coding a lighting solution, as described in Eqs. (1) and (2). The population size $|\mathcal{G}| = N$ is constant during the evolution. The initial population $\mathcal{G}^{(0)}$ is generated randomly. Note that the number L of lamp descriptions in a single solution is not fixed, but constrained: $L_{\text{MIN}} \leq L \leq L_{\text{MAX}}$. The variation of the population is based on two fundamental operations: crossover and mutation. Given two individuals:

$$\mathcal{S}_1 = \left\langle \mathcal{L}_1^{(1)}, \ldots, \mathcal{L}_{L^{(1)}}^{(1)} \right\rangle, \tag{14}$$

$$\mathcal{S}_2 = \left\langle \mathcal{L}_1^{(2)}, \ldots, \mathcal{L}_{L^{(2)}}^{(2)} \right\rangle, \tag{15}$$

we define as two-points crossover the following function:

$$\begin{aligned} \chi(\langle \mathcal{S}_1, \mathcal{S}_2 \rangle) = \Big\langle & \\ & \left\langle \mathcal{L}_1^{(1)}, \ldots, \mathcal{L}_i^{(1)}, \mathcal{L}_{i+1}^{(2)}, \ldots, \mathcal{L}_j^{(2)}, \ldots, \mathcal{L}_{j+1}^{(1)}, \ldots, \mathcal{L}_{L^{(1)}}^{(1)} \right\rangle, \\ & \left\langle \mathcal{L}_1^{(2)}, \ldots, \mathcal{L}_i^{(2)}, \mathcal{L}_{i+1}^{(1)}, \ldots, \mathcal{L}_j^{(1)}, \ldots, \mathcal{L}_{j+1}^{(2)}, \ldots, \mathcal{L}_{L^{(2)}}^{(2)} \right\rangle \Big\rangle, \end{aligned} \tag{16}$$

where i and j are random integers such that $1 < i < j < \min\{L^{(1)}, L^{(2)}\}$. Note that χ takes two solutions as input and returns two modified solutions. The operator can guarantee consistent results thanks to the fact that the set of lamps in a solution is ordered by their locations $\mathbf{v}$ in the interior space. This expedient implies that the choice of i and j is tantamount to partitioning the room into simply-connected spaces. Therefore the crossover is able to propagate the topological relationship in the parent solutions through the new individuals.

The mutation function ω operates on a single individual, and it is the composition of two different levels of mutation. The upper level is that of the ordered set of lamp descriptions, and it is mutated as following:

$$\omega_U(\mathcal{S}) = \begin{cases} \mathcal{S} \setminus \mathcal{L}_i & \text{if } r < 0.5 \\ \mathcal{S} \bigcup \{\mathcal{L}_{L+1}\} & \text{if } r \geq 0.5 \end{cases} \tag{17}$$

where r, here and in all the following equations, is a random number in range $[0 \cdots 1]$, and i is a random integer in range $[1 \cdots L]$. The lamp description $\mathcal{L}_{L+1}$ is a new lamp generated randomly from the set of possible lamps. Mutation at the lower level, that of single lamp description, is given by:

$$\omega_L(\mathcal{L}) = \begin{cases} \langle d, \{C, W\}, \mathbf{v} + \Delta\mathbf{v}, l, w, k \rangle & \text{if } r > \pi_p \\ \langle d, \{C, W\}, \mathbf{v}, \widetilde{l}, w, k \rangle & \text{if } r > \pi_l \\ \langle d, \{C, W\}, \mathbf{v}, l, w, \widetilde{k} \rangle & \text{if } r > \pi_k \end{cases} \quad (18)$$

where $\widetilde{l}$ is a new level of lighting, selected randomly from the possible light intensities for the lamp of type d, similarly for $\widetilde{k}$. The displacement $\Delta\mathbf{v}$ of lamp positioning is computed in a random direction from center $\mathbf{v}$, with a random offset within a neighborhood, decreased in the course of the evolution. The parameters $\pi_{\{p,l,k\}}$ are the mutation probabilities for, respectively, lamp position, lighting level, and color temperature.

Crossover and mutation are applied to the solutions selected from the current population $\mathcal{G}^{(t)}$ using the concept of crowded tournament selection, introduced in Sect. 2.1. We compare two randomly selected individuals and keep one winner, taking care that each solution of $\mathcal{G}^{(t)}$ will never be selected for more than two different couples. The comparison metrics requires first the partitioning of $\mathcal{G}^{(t)}$ into progressively non-dominated Pareto fronts, and a related ranking $r(\mathcal{S})$ of the solutions:

$$\mathcal{G} = \mathcal{F}_1 \cup \mathcal{F}_2 \cdots \quad (19)$$

$$r(\mathcal{S}) = i \text{ if } \mathcal{S} \in \mathcal{F}_i. \quad (20)$$

In addition, each solution has an associated *crowding distance* $c(\mathcal{S})$, measuring how crowded with other solutions is the neighborhood of the given solution. We skip the details of this computation, which follows the conventional niche count metric [30]. By combining $r(\mathcal{S})$ and $c(\mathcal{S})$ we define a comparison operator $\prec$, indicating when a solution $\mathcal{S}_i$ wins a tournament with another solution $\mathcal{S}_j$:

$$\mathcal{S}_i \prec \mathcal{S}_j \text{ if } \begin{cases} r(\mathcal{S}_i) < r(\mathcal{S}_j), \\ r(\mathcal{S}_i) = r(\mathcal{S}_j) \wedge c(\mathcal{S}_i) > c(\mathcal{S}_j). \end{cases} \quad (21)$$

For the construction of the set $\mathcal{M}$ of mating couples, two random perturbations of $[1 \cdots N]$ are generated: $\left[i_1^{(1)} \cdots i_N^{(1)}\right]$ and $\left[i_1^{(2)} \cdots i_N^{(2)}\right]$. Each couple is made by two winning solutions of the tournament:

$$
\begin{aligned}
\mathcal{M} = \Big\langle \\
&\Big\langle \arg\min_{\prec} \left\{ \mathcal{S}_{i_1^{(1)}}, \mathcal{S}_{i_2^{(1)}} \right\}, \arg\min_{\prec} \left\{ \mathcal{S}_{i_3^{(1)}}, \mathcal{S}_{i_4^{(1)}} \right\}, \Big\rangle, \\
&\Big\langle \arg\min_{\prec} \left\{ \mathcal{S}_{i_1^{(2)}}, \mathcal{S}_{i_2^{(2)}} \right\}, \arg\min_{\prec} \left\{ \mathcal{S}_{i_3^{(2)}}, \mathcal{S}_{i_4^{(2)}} \right\}, \Big\rangle, \\
&\dots \Big\rangle.
\end{aligned}
\tag{22}
$$

Equation (22) ensure that the two solutions in a couple are always different, and that the same solution can appear in no more than two different couples. Note that $|\mathcal{M}| = \frac{N}{2}$, and the strategy in Eq. (22) requires that N is a multiple of 4. The operator χ is applied on the couples of $\mathcal{M}$ with the crossover random probability, and on both elements of the couple the operator ω can be applied, with random mutation probability. Let us express with the composite operator ϕ the generation of $\mathcal{M}$ from $\mathcal{G}$, followed by the application of crossover and mutation, and the flattening of the couples into a set of new individual solutions, which will be of size N again.

One complete step of evolution can then be described as:

$$
\mathcal{G}^{(t+1)} \leftarrow \coprod_{\prec}^{N} \left(\mathcal{G}^{(t)} \ \cup \ \phi \left(\mathcal{G}^{(t)} \right) \right) \tag{23}
$$

where $\coprod_{\prec}^{N}$ is the reduction of a set its first N elements, ranked with the comparison operator $\prec$. The size N of the population remains constant during evolution. When $t = t_f$, the final generation programmed for the evolution, a Pareto set $\mathcal{F}_1$ is available, as the non-dominated set of solutions in $\mathcal{G}^{(t_f)}$.

In the presented problem of lighting optimization there are some conditions on the design process to be satisfied, therefore a constraint handling method has to be considered as well. The constrains in question concern positioning the lamps inside the interior environment:

- a lamp must be placed inside the room and in contact with the room surface;
- two lamps can not be placed in the same location;
- a lamp should be mounted on the walls or on the ceiling in accordance with its model of light fixture;
- some areas of the room are not suitable for lamp placement.

The constraint specifications are provided to the system within the 3D model of the environment itself. As stated in the beginning of Sect. 3, the walls and ceiling are structured as a discrete grid of vertices, each representing a feasible position for a lamp. With this approach, the set of constraints can be effortlessly reformulated for different experiments, ensuring absolute flexibility in the design process.

Since the satisfaction of the above constraints is mandatory for the problem, they can be referred as *hard constraints*. To handle them, we adopted a strategy based on preserving feasibility of solutions, where crossover and mutation operations are designed to always produce feasible offspring from feasible individuals. By construction, the mutation operator is only able to produce solutions that satisfy all the constraints. Conversely, if the crossover generates an infeasible solution, the last one is discarded and the operation is repeated with a different choice of crossover points.

3.3 Partitioning the Final Pareto Set

As in most multi-objective optimization problems, our lighting design system typically generates too many solutions in the final Pareto set, and selecting a single one that best reflects the preferences of the architect can be a daunting task. A considerable amount of research effort has been devoted to alleviate this inconvenience in the general multi-objective case, with several proposed methods that reduce the Pareto optimal set to a set of solutions that is attractive to the decision maker. A large part of the proposed methods assumes that the preferences of the decision maker are well known in advance, and can be expressed in mathematical terms and incorporated in the optimization algorithm [8, 9]. The situation of the architectural lighting design is different. Although the objectives defined in our optimization problem capture important requirements of the design process, there are aesthetic and stylistic components of the design process that elude mathematical formulations.

The great advantage of a tool like the one here proposed is for the architect to drastically restrict the search space of solutions, and to concentrate his or her creativity on a small number of simulated solutions. It is difficult to prescribe in advance any preferred part of the Pareto front, in principle the entire front can offer attractive solutions to the lighting designers, the choice is up to their expertise and aesthetic disposition. For this reason we focused on methods commonly classified as *a posteriori* [31], where the selection of a small subset of solutions is made on the entire final approximate Pareto front, computed without the incorporation of preferences from the decision maker.

First, we partitioned the set of solutions into a predefined number of clusters N_c, using the subtractive clustering algorithm [32, 33]. Let us define $\mathcal{O}$ the set of vectors in the fitness space of the final solutions $\mathcal{F}$:

$$\mathcal{O} = \{\mathbf{f}\,(\mathcal{S})\,|\mathcal{S} \in \mathcal{F}\}. \tag{24}$$

The vectors are normalized with all dimensions in range $[0 \cdots 1]$, we call $\bar{\mathcal{O}}$ the set of normalized vectors. For each solution a "potential" function ψ is introduced, that captures the neighborhood size of the solutions:

$$\psi^{(0)}(o_i) = \sum_{o \in \bar{\mathcal{O}}} \mathrm{e}^{\frac{4}{r_I{}^2}\|o_i - o\|}, \quad o_i \in \bar{\mathcal{O}}. \tag{25}$$

The superscript (0) is meant because Eq. (25) provides the initial values of the potentials, which are updated recursively, each time identifying as a cluster center the solution with the largest potential:

$$c_k = \arg\max_{o \in \bar{\mathcal{O}}} \left\{ \psi^{(k)}(o) \right\}, \tag{26}$$

$$\psi^{(k+1)}(o_i) = \psi^{(k)}(o_i) - \mathrm{e}^{\frac{4}{r_O{}^2}\beta\|o_i - c_k\|}\ \psi^{(k)}(c_k), \quad o_i \in \bar{\mathcal{O}}. \tag{27}$$

Equation (26) computes the center of the k-th cluster, the recursive loop is terminated when $k = N_c$, the predefined number of clusters. The parameters r_I in Eq. (25) and r_O in (27) act effectively as radii, influencing, respectively, the range of neighborhood of a solution and the closeness of distinct cluster. Their values are computed as a function of the number of desired clusters N_c:

$$r_I = \frac{2}{N_c}, \tag{28}$$

$$r_O = \frac{2.5}{N_c}. \tag{29}$$

All solutions $\mathcal{S}$ in $\mathcal{F}$ are partitioned in the clusters according to the distance of the vectors in fitness space to the cluster centers. Calling $\bar{\mathcal{S}}^{(k)}$ the solution in $\mathcal{F}$ that is center of cluster k, corresponding to the normalized vector c_k, the partitioning is done as following:

$$\begin{aligned} \mathcal{Q} = \Bigg\langle & \\ & \left\{ \mathcal{S} \ :\ \arg \max_{k \in [1..N_c]} \left\{ \left\| \mathbf{f}(\mathcal{S}) - \mathbf{f}(\bar{\mathcal{S}}^{(k)}) \right\| \right\} = 1 \right\}, \\ & \ldots, \\ & \left\{ \mathcal{S} \ :\ \arg \max_{k \in [1..N_c]} \left\{ \left\| \mathbf{f}(\mathcal{S}) - \mathbf{f}(\bar{\mathcal{S}}^{(k)}) \right\| \right\} = N_c \right\} \Bigg\rangle. \end{aligned} \tag{30}$$

For each cluster, a central representative is picked, so that the final set of solutions presented to the designer is very small.

4 Results

As discussed in Sect. 1, a satisfactory lighting quality is dependent on the visual tasks that are to be performed in the interior space, and on specific requirements of visual interest within the space. The translation of these user's requirements in the model is basically by means of the placement of samplers in the areas most critical from the lighting point of view, and by imposing the target illumination level.

All genetic parameters of the model have been tuned in a preliminary phase on simpler and smaller rooms, and these settings did not required further tweaking in the case studies eventually considered. The two case environments chosen for evaluating empirically our lighting optimization algorithm, are complex architectural interiors, with irregular and non-convex planimetries. The first case study is a reproduction of a coffee shop, the dimensions of the environment are $14 \times 10 \times 2.8$ m. The architecture of this room is characterized by a narrow dining area leading to a wider space with a lounge room and a bar counter. A total of 13 samplers have been used to evaluate illumination levels, placed in key areas where light should create visual interest. The genetic algorithm has been run with a population of 200 individuals, the final Pareto front is shown in Fig. 3, where it is possible to appreciate how the solutions smoothly span a large front of the two fitness. In the upper plot the complete population of final solution is shown, together with the Pareto front. The lower plot contains the partitioning of the final front in three clusters, and the solutions highlighted in red are the best representatives of each cluster. The solutions in cluster 1 and 2 are the solutions most qualified for, respectively, a lighting plan that privileges optimal cost and energy, or one that gives more importance to the quality of illumination.

The plots in the left column of Fig. 4 shows the structure of the Steiner tree wiring the lighting solutions, in (a) the solution with optimal light quality, in (c) that with the lower cost. It can be seen that the overall wiring is shorter in (c) than in (a). The plots in the right column of Fig. 4 are the isophotes computed in the room at a surface 1.5 m above the floor, for the solution with best quality in (b), and that with lowest costs in (d). The target level is well approximated in the solution (b), even close to the internal walls. In the solution (d) the overall level is slightly below the target, especially in the center of the shop, the uniformity is still acceptable, except the lower side of the horizontal internal wall, an area difficult to light properly with few lamps.

The Fig. 5 shows photorealistic renderings of the interior space from two different points of view, (a) and (b) refer to the solution with optimal light quality, while

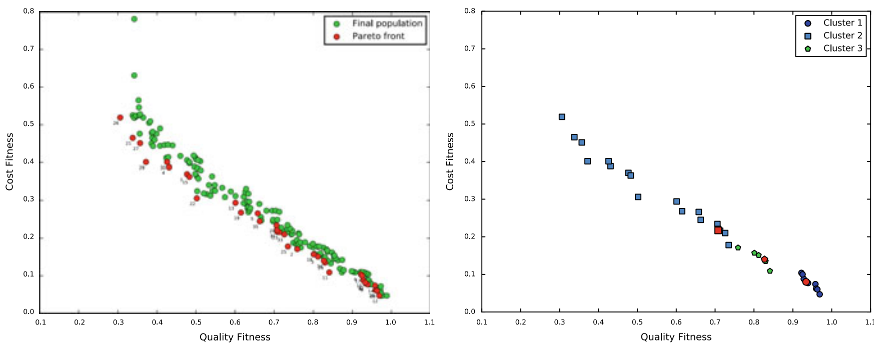

Fig. 3 Final Pareto front of the optimization of the coffee shop case. In the upper plot there is the complete populations at the end of the optimization, and the Pareto front. In the lower plot the solutions are grouped in three clusters, and for each cluster the best representative is marked in red

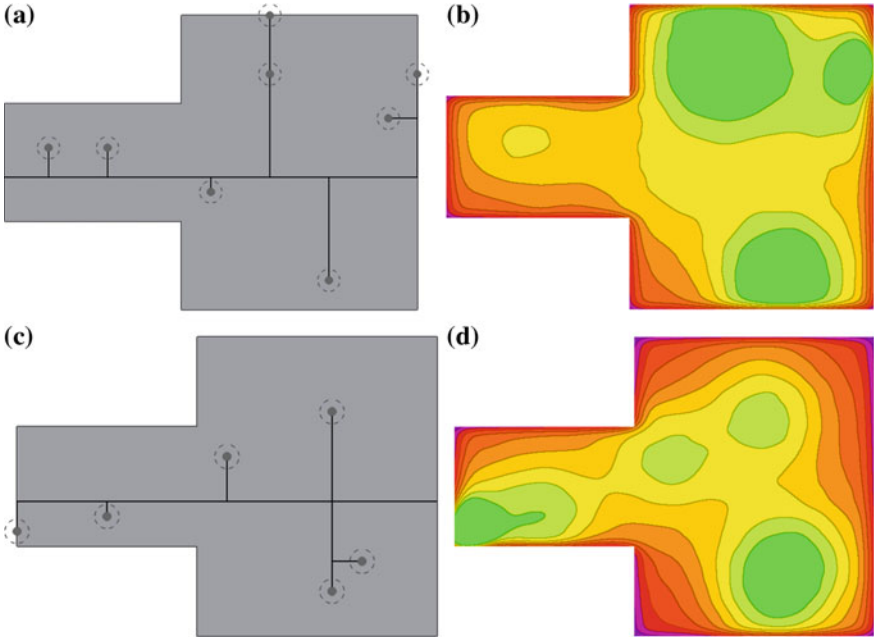

Fig. 4 Results of the optimization for the coffee shop case. Plots on the left are schemes of the Steiner tree for the electrical wiring of the light fixture. Plots on the right are isophotes at 1.5 m level, colors are coded with green the perfect matching of the desired target of illumination, and hues towards yellow, orange, red, and purple, are progressive displacements from the target level. Plots **a** and **b** refer to the solution with best quality, plots **c** and **d** to that with lower costs. In all plots the internal walls and pillars are shown

(c) and (d) refer to the solution with minimum cost. It can be seen that even this solution, which saves 67% of the energy consumption of the previous solution, has an acceptable level of lighting with fair uniformity.

The second case study is a reproduction of a hall in a shopping mall, with dimensions of $12 \times 11 \times 4.0$ m, composed of a central area connected to secondary small shop. The main space contains a column with display stands and an area serving as lounge room, while the secondary area for the small shop has a lower ceiling level and contains several product racks and a counter with the cash register.

A total of 14 samples have been used, with a genetic population of 200 individuals. As in the previous case study, there is a wide and smooth coverage of the Pareto front in Fig. 6, although some solutions of this case study reached poor level of quality fitness, compared to the previous case. This result can be explained by the brighter shading of walls and floors in the mall environment (pale yellow and white) reflecting more light than the deep red and beige color tones of the coffee shop, which requires more intense light sources in order to reach the same perceived illumination level. Moreover, the construction of the suboptimal Steiner tree is less straightforward than in the coffee shop case, because it is not possible to set up a trunk exploiting symmetry

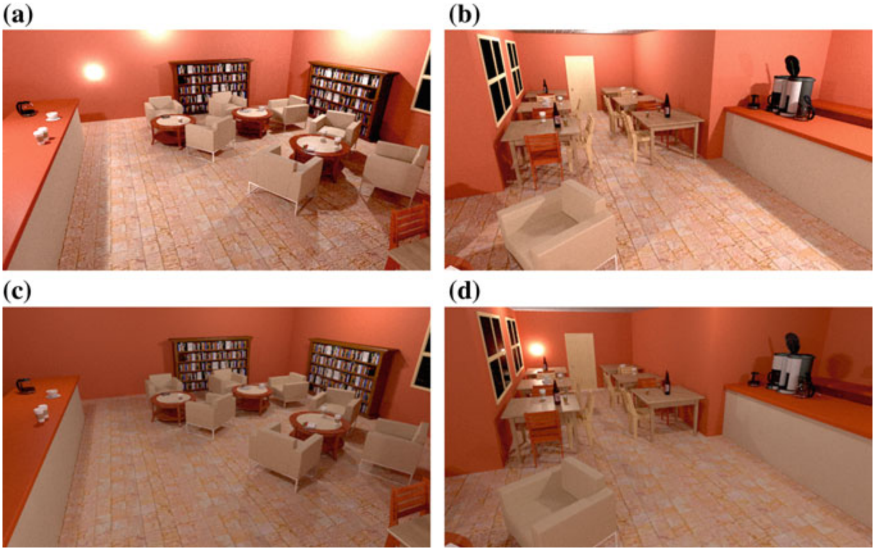

Fig. 5 Rendering of two views inside the coffee shop, lighted with two different solutions, the representative of the cluster with best quality in (**a**) and (**b**), and the representative of lower costs in (**c**) and (**d**)

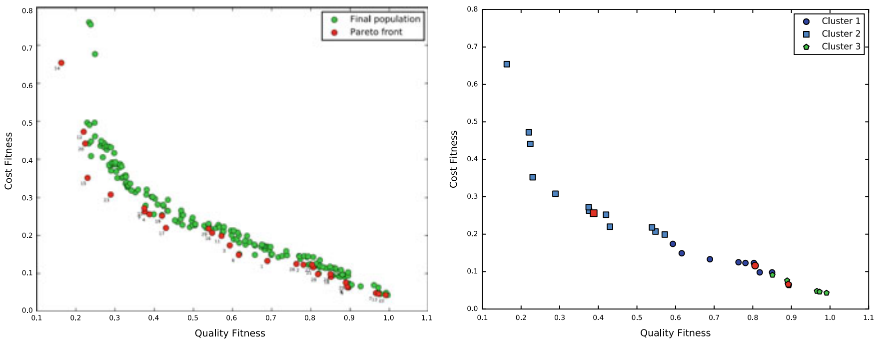

Fig. 6 Final Pareto front of the optimization of the shopping mall case. In the upper plot there is the complete populations at the end of the optimization, and the Pareto front. In the lower plot the solutions are grouped in three clusters, and for each cluster the best representative is marked in red

along the horizontal dimension, as visible in the left columns of Fig. 7. Nonetheless, the visual results are rather satisfying, as shown in the photorealistic renderings of two of the best representative solutions in Fig. 8, the first one preferring light quality and uniformity, the second one considering optimal level of energy saving and costs. The solution illustrated in (c) and (d), even though is clearly darker than the other, its energy saving is as high as 66% with respect to the light configuration in (a) and (b).

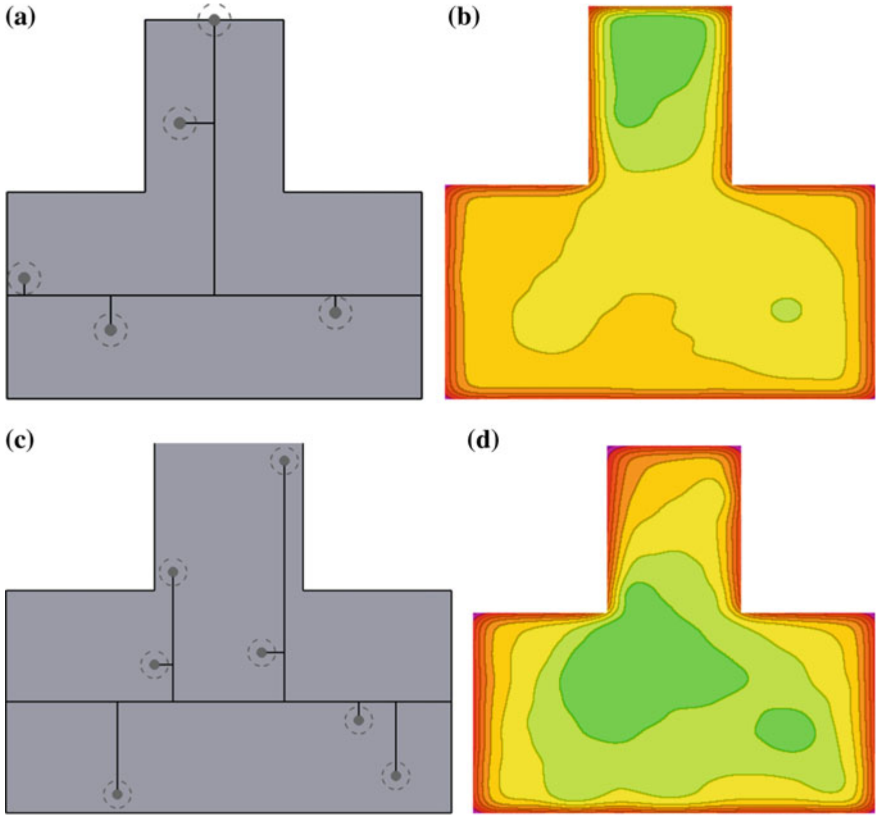

Fig. 7 Results of the optimization for the shopping mall case. Plots on the left are schemes of the Steiner tree for the electrical wiring of the light fixture. Plots on the right are isophotes at 1.5 m level, colors are coded with green the perfect matching of the desired target of illumination, and hues towards yellow, orange, red, and purple, are progressive displacements from the target level. Plots **a** and **b** refer to the solution with best quality, plots **c** and **d** to that with lower costs. In all plots the internal walls and pillars are shown

This case study demonstrates how the presented algorithm can be a suitable tool to effectively design light configuration for a frequently changing environment, a shopping mall, with minimum effort from the user.

5 Conclusions

In this paper we described a system for inverse design of interior lighting based on the integration between the 3D computer graphic software Blender, a NSGA-II based multi-objective genetic algorithm, and a post-selection of best solutions based on

Fig. 8 Rendering of two views inside the shopping mall, lighted with two different solutions, the representative of the cluster with best quality in (**a**) and (**b**), and the representative of lower costs in (**c**) and (**d**)

cluster analysis. The system takes as input an arbitrary interior environment, including realistic furniture and materials, with the description of the lighting requirements in terms of desired average illumination, and placement of samplers in the key locations of the interior space. We grouped two conceptually different sets of objective functions: on one side those contributing to the pleasantness of the lighting, and on the other side those contributing to the expenses in the realization and the functioning of the lighting system. In the implementation here described, we chosen specific objectives, common in the lighting design process: for the first group the compliance with the target illumination level, the uniformity of light distribution in the interior space; for the second group the overall length for electrical wiring, and the consumption of electric power. The generality of our approach allows for easy addition of other requirements, like, for example, a desired distribution of color spectra, of glaring avoidance. The cases presented as results demonstrate the effectiveness of the system in helping the process of interior lighting design.

References

1. Gordon, G.: Interior Lighting for Designers. Wiley, New York (2014)
2. Livingston, J.: Designing With Light: The Art, Science, and Practice of Architectural Lighting Design. Wiley, New York (2015)

3. Wunderlich, C.H.: Light and economy: an essay about the economy of prehistoric and ancient lamps. In: Chranovski, L. (ed.) Lychnological News, pp. 251–264. LychnoServices, Hauterive (Suisse) (2003)
4. Kahraman, C. (ed.): Computational Intelligence Systems in Industrial Engineering With Recent Theory and Applications. Atlantis Press, Paris (2012)
5. Commercial buildings energy consumption survey: Technical report, U.S. Energy Information Administration (2012)
6. Bertoldi, P., Hirl, B., Labanca, N.: Energy efficiency status report. Technical report, European Commission—Institute for Energy and Transport (2012)
7. Sansoni, P., Farini, A., Mercatelli, L. (eds.): Sustainable Indoor Lighting. Springer, Berlin (2015)
8. Jaimes, A.L., Coello, C.A.C.: Interactive approaches applied to multiobjective evolutionary algorithms. In: Doumpos, M., Grigoroudis, E. (eds.) Multicriteria Decision Aid and Artificial Intelligence: Theory and Applications, pp. 191–207. Wiley, New York (2013)
9. Bechikh, S., Kessentini, M., Said, L.B., Ghédira, K.: Preference incorporation in evolutionary multiobjective optimization: a survey of the state-of-the-art. Adv. Comput. **98**, 141–207 (2015)
10. Plebe, A., Cutello, V., Pavone, M.: Evolving illumination design following genetic strategies. In Sabourin, C., Merelo, J.J., Warwick, K., Madani, K., O'Reilly, U.M. (eds.) 9th International Joint Conference on Computational Intelligence, pp. 222–233. Scitepress (2017)
11. Larson, G.W., Shakespeare, R.: Rendering with Radiance: The Art and Science of Lighting Visualization. Morgan Kaufmann, San Francisco, CA (1997)
12. Baltes, H. (ed.): Inverse Source Problems in Optics. Princeton University Press, Princeton, NJ (1978)
13. Kawai, J., Painter, J.S., Cohen, M.F.: Radioptimization: goal based rendering. In: Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques, pp. 147–154 (1993)
14. Schoeneman, C., Dorsey, J., Smits, B., Arvo, J., Greenberg, D.: Painting with light. In: Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques, pp. 143–146 (1993)
15. Patow, G., Pueyo, X.: A survey of inverse rendering problems. Comput. Graph. Forum **22**, 663–687 (2003)
16. Papalambros, P.Y., Wilde, D.J.: Principles of Optimal Design. Cambridge University Press, Cambridge, UK (1988)
17. Andersen, M., Gagne, J.M., Kleindienst, S.: Interactive expert support for early stage full-year daylighting design: a user's perspective on Lightsolve. Autom. Constr. **35**, 338–352 (2013)
18. Gagne, J., Andersen, M.: A generative facade design method based on daylighting performance goals. J. Build. Perform. Simul. **5**, 141–154 (2012)
19. Futrell, B., Ozelkan, E.C., Brentrup, D.: Optimizing complex building design for annual daylighting performance and evaluation of optimization algorithms. Energy Build. **92**, 234–245 (2014)
20. Moylan, K., Ross, B.J.: Interior illumination design using genetic programming. In Johnson, C., Carballal, A., ao Correia, J. (eds.) Proceedings IV Conference on Evolutionary and Biologically Inspired Music, Sound, Art and Design, pp. 148–160 (2015)
21. Daenzer, S., Montgomery, K., Dillmann, R., Unterhinninghofen, R.: Real-time smoke and bleeding simulation in virtual surgery. In: Westwood, J.D., Haluck, R.S., Hoffman, H.M., Mogel, G.T., Phillips, R., Robb, R.A., Vosburgh, K.G. (eds.) Medicine Meets Virtual Reality, pp. 94–99. IOS Press, Amsterdam (2007)
22. Plebe, A., Grasso, G.: Particle physics and polyedra proximity calculation for hazard simulations in large-scale industrial plants. In: American Institute of Physics Conference Proceedings, pp. 090003–1–090003–4 (2016)
23. Janikow, C.Z., Michalewicz, Z.: An experimental comparison of binary and floating point representations in genetic algorithms. In: Proceedings of the 4th International Conference on Genetic Algorithms, pp. 31–36 (1991)

24. Deb, K., Agrawal, S., Pratap, A., Meyarivan, T.: A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In: International Conference on Parallel Problem Solving From Nature, pp. 849–858 (2000)
25. Fortin, F.A., De Rainville, F.M., Gardner, M.A., Parizeau, M., Gagné, C.: DEAP: evolutionary algorithms made easy. J. Mach. Learn. Res. **13**, 2171–2175 (2012)
26. Hwang, F.K., Richards, D.S., Winter, P.: The Steiner Tree Problem. North Holland, Amsterdam (1992)
27. Hanan, M.: On Steiner's problem with rectilinear distance. SIAM J. Appl. Math. **14**, 255–265 (1966)
28. Kahng, A.B., Robins, G.: On Optimal Interconnections for VLSI. Springer, Berlin (1994)
29. Chen, H., Qiao, C., Zhou, F., Cheng, C.K.: Refined single trunk tree: a rectilinear Steiner tree generator for interconnect prediction. In: Proceedings of the International Workshop on System-Level Interconnect Prediction, pp. 85–89 (2002)
30. Deb, K.: Multi-objective Optimization Using Evolutionary Algorithms. Wiley, New York (2001)
31. Zio, E., Bazzo, R.: Multiobjective optimization of the inspection intervals of a nuclear safety system: a clustering-based framework for reducing the pareto front. Ann. Nucl. Energy **37**, 798–812 (2010)
32. Chiu, S.L.: Fuzzy model identification based on cluster estimation. J. Intell. Fuzzy Syst. **2**, 267–278 (1994)
33. Zio, E., Bazzo, R.: A comparison of methods for selecting preferred solutions in multiobjective decision making (4), 23–43