Proceedings of the International Association for Shell and Spatial Structures (IASS) Symposium 2009, Valencia
Evolution and Trends in Design, Analysis and Construction of Shell and Spatial Structures
28 September – 2 October 2009, Universidad Politecnica de Valencia, Spain
Alberto DOMINGO and Carlos LAZARO (eds.)

Decision Support Tools for Parametric Design

Megan NEILL*, Roel VAN DE STRAAT [a], Jeroen COENDERS [b]

* Delft University of Technology
Faculty of Civil Engineering and Geosciences
Structural Design Lab
Stevinweg 1, 2628 CN
neill.megan@gmail.com

[a,b] Arup, Delft University of Technology
Naritaweg 118, 1043 CA Amsterdam, Stevinweg 1, 2628 CN Delft
[a] r.j.vandestraat@tudelft.nl
[b] jeroen.coenders@arup.com

Abstract

Computer-aided design tools can be of great assistance in the design process. However, commercial software typically caters to the optimisation of one objective and does not address the fact that real world problems tend to be solved by assessing the trade-offs between multitudes of competing objectives. Therefore the question is posed as to how computer-aided design tools can become assimilated into current design practice while aiding the complex decision-making process. A prototypical tool is presented that offers the capability to perform trade-off studies and is integrated within a parametric modelling environment. By defining the objectives, design variables and constraints of interest, a series of Pareto-optimal solutions comprising the trade-off surface is put forth from all possible permutations. The designer is assured that there exists no other solution whereby an objective can be improved upon without simultaneously placing another objective at a disadvantage. Such a tool can be effectively adopted and utilized to make more informed decisions in a relatively short amount of time.

Keywords: multi-objective optimisation, Pareto optimal, trade-off, parametric and associative design systems

1. Introduction

One of the roles of a designer in any discipline is to blend intuition, experience, and heuristics into a design and then communicate the results to others. Time and budget constraints challenge the ability to perform this role, while maintaining high levels of quality. Computer aided design tools could be of assistance under such conditions, most often to perform iterative analysis in order to determine a solution of a single objective optimisation problem. In practice, however, design problems encompass many objectives that must be balanced against one another in order to determine a final design. As the number of objectives increases, finding a solution by grasping the trade-offs between all objectives is likely to become complex and difficult. There is a certain reliance on the ability of the designer to intuit the relationship between the input variables and performance functions. Therefore a prototypical tool is presented that offers the capability to perform trade-off studies by utilizating the pre-established geometric and quantitative relationships within a parametric modelling environment. In the following pages the background of the applied multi-objective decision making methods will be described along with the manner in which the methods are used as the basis for the tool framework. Finally a simple case study is presented to show the utility of using such a tool.

2. Applied methods of multi-objective optimisation

Several key components of a multi-objective design optimisation problem are adopted into the tool framework. Briefly described here, they include the problem definition, a method to determine Pareto optimal solutions, and the expression of preference for a dual objective problem.

2.1 Mathematical definition of multi-objective design problem

A multi-objective optimisation problem is typically concerned with minimizing a number of objective functions subject to equality and/or inequality constraints (Collete and Siarry [1]). Mathematically, a dual objective problem is expressed as the following:

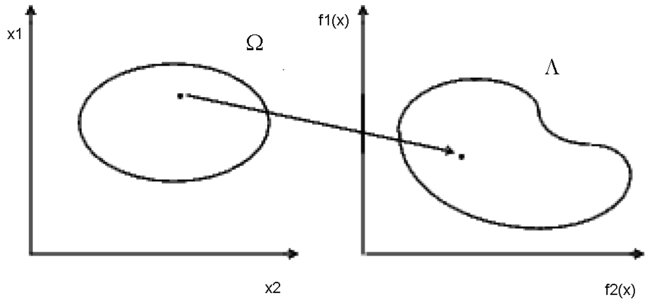
$$\text{Minimize} \quad F(\mathrm{x}) = [f_1(\mathrm{x}), f_2(\mathrm{x})]^T \tag{1a}$$

Subject to:

$$\begin{aligned} g_j(\mathrm{x}) &\le 0, \quad j = 1,2,...m \\ h_l(\mathrm{x}) &= 0, \quad l = 1,2,...e \end{aligned} \tag{1b}$$

Where m is the number of inequality constraints, e is the number of equality constraints, and $\mathbf{x}$ is a vector of design variables. In order to explore the various potential solutions, several plots are of interest.

The first illustrates the *design space*, Ω, defined as a plot of the objective function values in a space defined by the design variables. Applicable constraints may also be plotted and the region bounded by these constraints form the *feasible design space*. Conversely the *criterion space*, Λ, is a plot whereby each axis is attributed one of two objective functions. The mapping of points from the feasible design space to the criterion space creates the *feasible criterion space*.



(a) Design space (b) Criterion space

Figure 1 Mapping from a design space with two variables to a criterion space of two objective functions (Mathworks [2])

2.2 Definition of Pareto optimality

It is rarely the case where a vector of design variables **x** minimizes both objective functions simultaneously. Instead, the plots of criterion space are used to determine a set of vectors whereby no other combination of design variables improves upon the values of one objective function without having a determinental affect on the other. By this definition the solution associated with this instance of design variables is considered to be *Pareto optimal*. In other words, these solutions are considered to be *nondominated*, implying that a *dominated* solution can be improved upon in all objective functions and is therefore considered of no interest.

Looking at the criterion space of Figure 2, the set of nondominated solutions lies on the darkened curve C-D. When moving from point A to point B, the value of f_1 decreases from f_{1A} to f_{1B}, a desired move in case of a minimization problem. However the value of f_2 increases from f_{2A} to $f_{2B,}$ an undesired move. Because an improvement in one objective, f_1, requires degradation in the other, f_2, points A and B are considered nondominated in relation to each other. Ultimately, multi-objective optimisation methods are concerned with the generation and selection of nondominated points.

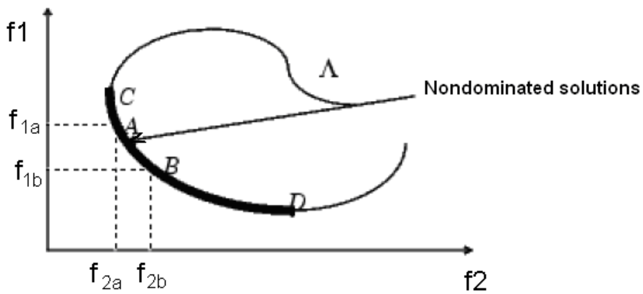


Figure 2 Set of nondominated solutions (Mathworks [2])

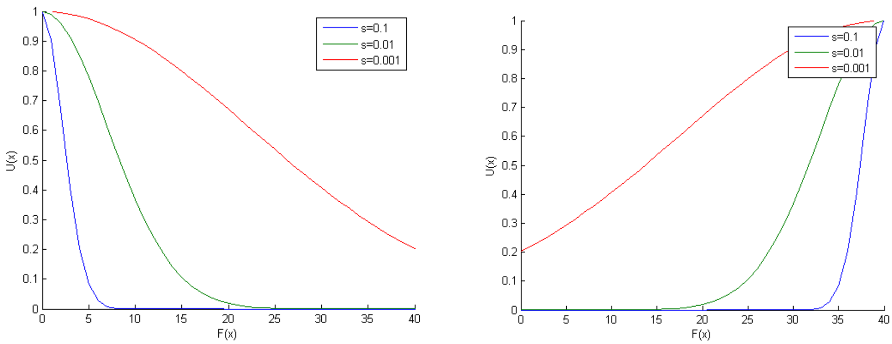
2.3 Expressing preference using utility functions

In multi-objective design problems it is possible to express preference for particular values of objective functions. In doing so, the designer can more quickly arrive at desired solutions. Various approaches of generating and selecting non-dominated points are available, of which the linear weighting method, the distance function method, physical programming method are common (Messac [3]). The preference in the case study tool [see Chapter 4] will be expressed according to the *physical programming method*. This method is based on the use of a utility function as expressed in Equation 2:

$$U(\mathrm{x}) = e^{-S[f(\mathrm{x})-f(\mathrm{x}_o)]^2} \qquad (2)$$

whereby U takes on a value between 0 and 1, $f(\mathbf{x}_o)$ is the most preferred objective function value, $f(\mathbf{x})$ represents the objective function value unique to the instance of design vector $\mathbf{x}$, and S represents the rate of the attenuation to zero of the slope of U(x).

For example, a set of solutions with objective function values between 0 and 40 are evaluated. The effect of modifying the preferred objective function value to favor higher numbers (i.e. $f(x_o)$ = 40) or lower values (i.e. $f(x_o)$ = 0), is illustrated in Figure 3. In addition, the varied slope of the function demonstrates the effect of modifying the value of S between 0.001, 0.01, and 0.1.



(a) Less is better, $f(x_o) = 0$ (b) More is better, $f(x_o) = 40$

Figure 3 Comparison of utility functions with varying values of S and $f(x_0)$

3. Design Tool Framework

As demonstrated in Figure 4, the goal of the proposed design tool is to improve upon the traditional method whereby external programs outside of the CAD model perform optimisation routines. The improvement is accomplished by fully integrating optimisation methods into pre-existing industry software. With this approach the learning curve associated with using a new program and the lag time between updating both models is bypassed. An added benefit of integrating the optimisation framework within the CAD model is the ability to represent results visually rather than numerically.

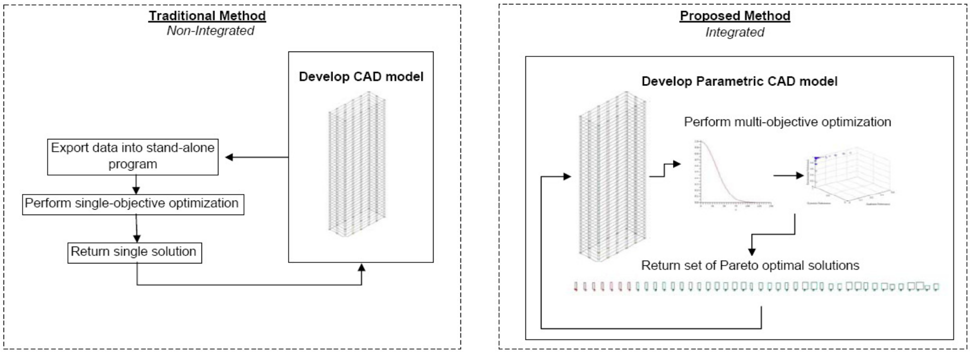
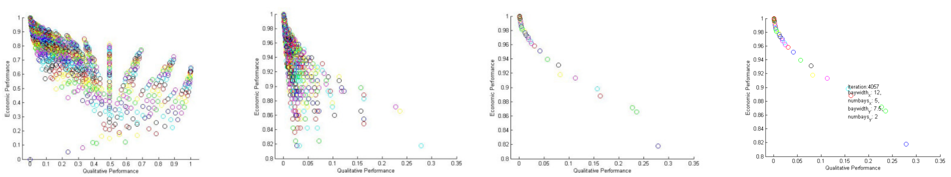


Figure 4 Comparison of non-integrated method vs. proposed integrated method

The design tool workflow is outlined in Figure 6. As represented by the vertical dashed line, a key component to the workflow is the human-computer feedback loop. The human component consists of transcribing the methods of multi-objective design, as described in Chapter 2, into the parametric model through features uploaded via a dynamic link library. The features include the vector of design variables **x**, objective functions, constraints, and the preference functions. With these components captured into the design model history, the computer component solves an algorithm is used to determine the set of Pareto optimal solutions. The iterative looping occurs when the designer chooses to modify the components of the design problem as a reaction to the solutions presented. It is the intent that this iterative process becomes an interactive experience with which to explore the design space and direct the search of Pareto optimal solutions. Plots of the design space, criterion space, and set of Pareto optimal solutions can be plotted and viewed as shown in Figure 5. In order to explore the Pareto optimal points, each plotted point in Figure 5d can be selected to reveal the instance of design variables to that solution.



(a) Criterion space (b) Feasible criterion space (c) Pareto optimal solutions (d) point and click display of vector **x**

Figure 5 Plot of criterion space for a two objective design problem

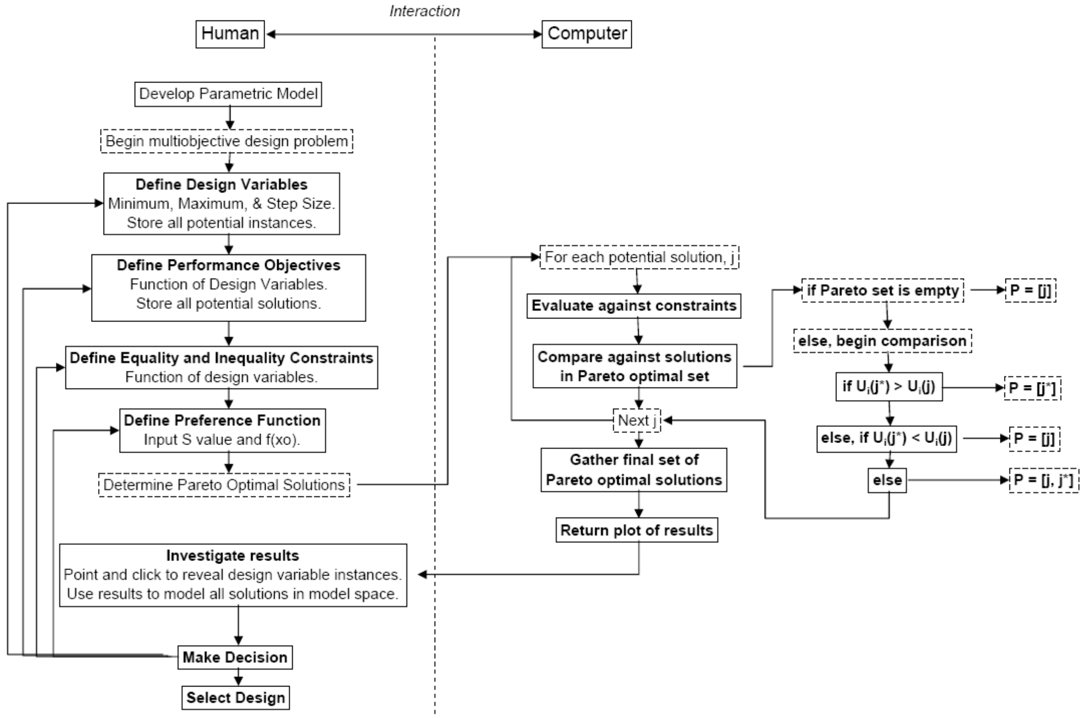
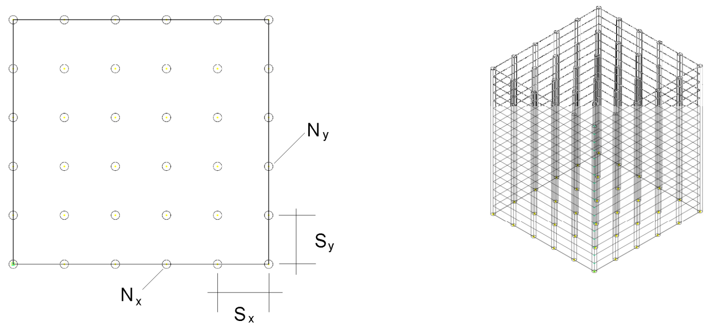


Figure 6 Tool workflow chart

4. Application to building model case study

A parametric model of an orthogonal, multi-story building with glass facade is created as a demonstrational case (Fig. 7b). The design variable feature is composed of both the number of columns, N_i, and column bay widths, S_i, in each respective direction (Fig. 7a). These values have a set minimum, maximum, and step size value. From the permutation of possible variable combinations there are 4,096 different building models to evaluate.

The constraint feature applies a maximum building height, minimum floor area, and minimum floor-to-floor clearance height. Solutions that are generated which fail to meet the constraint criteria are discarded. Three performance function features and corresponding preference features express the desirability to maximize the depth of exposure to natural light, minimize the structural costs, and minimize the cooling load due to heat gain through the facade.



(a) Plan view showing design variables (b) Isometric view of parametric model

Figure 7 Views of parametric model used for demonstration problem

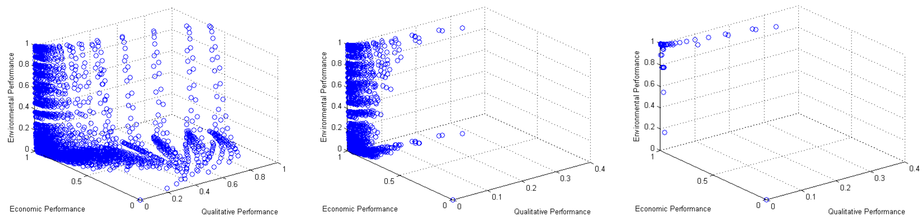
Specific data is researched externally and included in the definition of the performance functions. This information includes building orientation, environmental data, thermal properties of glass facades, and unit material costs. The results show that from the 4,096 potential solutions, 42 Pareto-optimal solutions exist. The data attributed to three sample solutions are shown in Table 1 in order to illustrate the idea of nondomination and how Pareto optimal points are determined.

Table 1 Comparison of sample selections of potential solutions

No.	Iteration	$U_1(x)$	$U_2(x)$	$U_3(x)$
1	125	0.278	0.817	0.996
2	1340	0.235	0.865	0.987
3	1600	0.229	0.864	0.981

When comparing solution #1 and #2, it is evident that solution #1 dominates solution #2 in performance objective 1 and 3, and is dominated by solution #2 in performance objective 2. In this case these solutions are considered nondominated in relation to one another and lie on the Pareto optimal front. However, when comparing solution #2 and #3, it is evident that solution #2 dominates solution #3 in all objectives. Solution #3 is therefore considered of no value to the designer and discarded.

The plots of the criterion space, feasible criterion space, and the set of Pareto optimal solutions are illustrated in Figure 8. All data associated with the Pareto optimal solutions are captured within the design model and can be used as input for other features.



(a) Criterion space (b) Feasible criterion space (c) Pareto optimal solutions

Figure 8 Plot of criterion space and Pareto optimal solutions given three performance objectives

For example, as shown in Figure 9, the instances of design variables are used to quickly model the floor plan of each solution. Additional information, such as the iteration number and the specific utility function value, can easily be included in the graphical output as text.

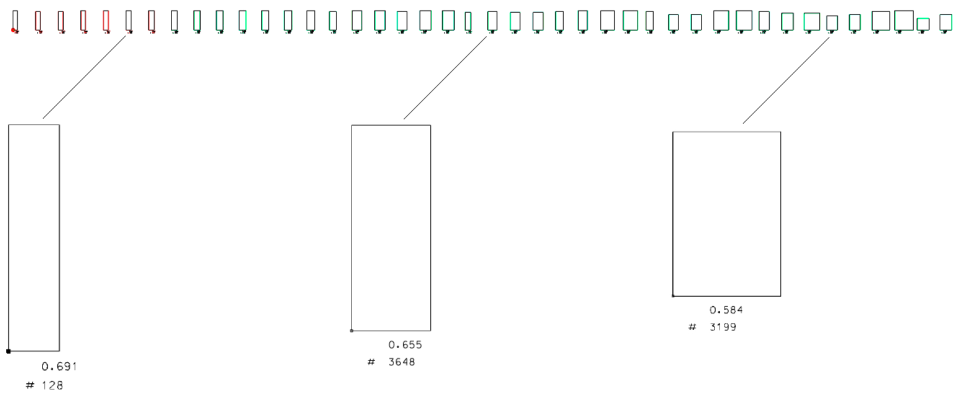


Figure 9 Representation of all Pareto optimal solutions in model space

5. Discussion

The potential benefits of working in the proposed method are four fold. First, rather than using a stand-alone program, the multi-objective design problem is captured within a parametric model and is updated in real-time as changes to the model are made. Secondly, the data relating to the Pareto optimal solutions can be used as inputs for other modeling features, significantly decreasing the time between acquiring, interpreting, and communicating numerical results. Thirdly, there is a certain insight and control the designer achieves by participating in the iterative human-computer loop. For example, the designer can begin searching the solution spaces without specifying any preference. As certain values of the performance functions are identified as desirable, the designer can narrow the window of preference and effectively steer the design toward particular area of the design space. As the design evolves, the range and step size of the design variables can be refined to explore subregions of the design space. Finally, as is custom with parametric models, "black box" issues do not arise due to the reliance on the designer for the definition of performance rather than an external analysis program. Although this limits the depth of analysis, such quick assessment proves especially useful during conceptual phases of design.

6. Conclusion

The idea of integrating multi-objective design methods into a parametric associative modeling environment expands upon the use of stand-alone single-objective optimisation programs. Such a tool is an aide to the complex decision making process by providing a framework in which to define the components of a design problem. In doing so, the focus is placed less on exploration through heuristic methods and more on exploration through the holistic perspective gained from viewing the criterion space. With such an understanding the designer can anticipate the effect on the project as design changes arise. The extraction of Pareto optimal solutions streamlines the evaluation period by ensuring that only solutions worth the designer's time are selected from the entire design space. The application of such tools into parametric associative environments encourages the designer to develop both the model and design problem simultaneously. In this way, the benefits of multi-objective methods are more readily adopted by being integrated into the already established design process. These advantages together provide an integrated aide to the complex decision making process, ultimately leading to better design.

References

[1] Collette, Yann, & Siarry, Patrick. 2003. *Multiobjective optimisation: principles and case studies*. Berlin: Springer.

[2] http://www.mathworks.com, July 2008. © *1984-2009- The MathWorks, Inc.*

[3] Messac, A. 1996. Physical programming: Effective optimisation for computational design. *Aiaa Journal,* **34**, 149-158.