

FRONTIER OPTIMIZATION

Snyder Wilson Consulting

Abstract

Many different approaches to engineering design have been proposed over the years. A modern approach, used by many companies requiring extensive engineering work, is the technique of optimization. For design problems with multiple criteria, the concept of Pareto efficiency can be used to define optimal designs. A design is said to be Pareto efficient if there is no other design with a more desirable value for all criteria. That is, the set of all Pareto efficient designs, known as the Pareto frontier, represent the available trade-offs between co-optimal designs. Visualizing the entire Pareto frontier allows the designer to bring their judgment to bear on the problem of picking the best possible design using competing criteria. A type of visualization, the Frontier graph, allows designers to visualize the Pareto Frontier for problems involving many designs and criteria.

Background

A common approach to engineering design uses an iterative design process. After formulating the problem, the engineer will create a design using educated guesswork informed by his experience. The engineer evaluates this design through calculations, usually performed by software, which determine if the design meets the requirements. If the design is lacking in some way, the design is modified to ameliorate the deficiencies. The new design is re-evaluated and the process is repeated until an acceptable design is produced.

This process is natural, requiring no specialized training or tools to be employed. As such, it is commonly used throughout industry. There are some drawbacks however.

It is very labor intensive. During each iteration the engineer must compare the current design's performance to the desired specifications, tweak the design to compensate for any shortcomings, and rerun all calculations and simulations. As the number of iterations increases, the labor required increases proportionally. In response to these limitations, there is growing interest in optimization.

Optimization is a technique which attempts to find not just an acceptable design but the best possible design. Designs judged by a single criterion can be evaluated using single-objective optimization. Consider, for example, a design where cost is the only criterion under review. Say a function can be created which represents the cost of the design for a given set of design parameters. Minimizing this function would yield the optimal design. For multi-objective designs, Pareto optimization can be used.

Pareto Optimization

Many designers view their work as an additive process, creating components and adding them to the blank slate of endless potential that they start with. An alternative perspective is to view design as winnowing down the set of all possible designs. Anyone designing a product for the real world must create within certain limits, such as the laws of physics. These constraints form the design space within which the engineer can work. Every decision the designer makes, every parameter fixed, shrinks the design space. Once a structural engineer decides to use steel beams in a building, for instance, the design space for that structure shrinks from one that included every element in nature to one restricted to a handful of alloys. The design is complete, then, when the design space has shrunk to include only one possible design.

Writers, poets, and philosophers have long discussed the design space for the written word by using a theoretical library, known as the Library of Babel, containing all possible books. The first book in this library is entirely blank. The next book starts with the letter “a” followed by empty pages. The following book begins with “aa,” then “aaa,” and so on until there is a book entirely filled with the letter “a.” The next book starts with the letter “b” followed by the letter “a” filling the rest of the book, then “bb” and so on. In this way a collection of books with every possible character combination can be created. Such a library would contain not just every book that has been written but all books that can be written. It would contain every bestselling novel, every poem, and every play that has ever been committed to paper. Somewhere in the library there is a book containing a complete running record of your inner monologue from birth until the present.

This library, however, is unmanageably vast. Let’s say the library is filled with books that are all exactly 500 pages long each with 40 lines containing 50 characters. Each character is from a set of 100 possibilities including the upper and lower case English alphabet, numbers, blank spaces and punctuation with room left over for some non-English letters. Books longer than 500 pages can be created in volumes from books in the library. Constraining the library in this way, by using the values Dennet used in *Darwin’s Dangerous Idea*, makes the math easy. Since each book consists of one million characters, and each character has 100 options, the library contains $100^{1,000,000}$ books. For reference, as Dennet notes, the observable universe only contains 100^{60} particles. It is then impractical to find any particular book you need, or indeed to find any useful book at all. Since the books contain all possible character combinations, the vast majority of books will contain complete gibberish. The Library,

then, could not be created or stored by us, but is useful in framing how we think about design problems.

The design space confronting an engineer differs from that facing a writer so a slightly different kind of library will be needed. The analogue to each character in a book would be a value for a parameter in the design. The book then becomes a one dimensional array with each element holding the value for a parameter. The parameters could be discrete, continuous, or binary. For example the design of a helical coil spring includes the parameters defining the material, wire diameter, coil diameter, and surface treatment. The material choice is a discrete parameter consisting of the available choices of material. The wire diameter and coil diameter are at least theoretically continuous. In reality, springs can only be ordered in discrete sizes defined by the tolerance of manufacturers or by the choices available in the marketplace. The common surface treatment known as peening, which is used to remove residual stresses, is a binary parameter. That is, the spring either did or did not undergo the process. In the Library of Babel, every location in a book has the choice of the same 100 characters. The books in our library, which are analogous to a single complete design, do not have the same set of possible choices for each location. Locations representing continuous parameters do not offer the same choices as locations representing discrete or binary parameters. Even two continuous parameters may have two different ranges. For example, a parameter describing wire size might have a range on the order of a fraction of an inch while the coil size is on the order of several inches. If the design of a helical coil spring were restricted to a circular cross section, with ground ends, set and peened, a book describing a specific design would only need three characters: one to describe

the wire size, one for the coil diameter, and a third for the material used. We could then create the Library of Helical Coil Springs with the Given Constraints. This Library would encompass the entire design space for our problem.

There are four problems with using a complete design space. The first two problems are to create and store the design space. Computers can perform enormous quantities of calculations and store vast amounts of information, but even they have limits. The problem must contain a small enough number of parameters limited to a small enough range that computers can generate and store the entire design space. Determining which parameters to include and what ranges to use can have a big impact on the resulting options facing the designer. Picking ranges that are too small can eliminate potential valuable designs. Using ranges that are too large can increase calculation time and inundate the designer with too many choices to make sense of.

The third problem is navigation. In order to keep track of the designs in the design space, each design will be assigned a unique number, or index. However, we do not want to be overwhelmed by a huge number of worthless designs. We need a filter to separate the good designs from the bad ones. Fortunately, engineering problems have objective criteria that can be used to eliminate poor designs. One set of filters could use pre-determined limits such as a minimum safety factor or maximum stress to eliminate designs that would fail to meet the requirements. Another type of filter could use the criteria, such as cost, that the designer will use to evaluate the designs. The filter could take each design and compare it to all the other designs based on these criteria. If there is another design that beats the test design for all the criteria, the test design can be

eliminated. Once all the designs have been analyzed in this manner, the remaining designs are all Pareto Optimal and are collectively known as the Pareto Frontier.

Frontier graph

Once a Pareto Frontier has been defined, it must be evaluated in some way. Our approach visualizes the Frontier so that knowledge about the Frontier can be used to select the desired design. A frontier graph allows for the visualization of many criteria for many designs in a single graph. In addition, each criterion is easily compared to every other criterion. First, if the design space consists of continuous parameters, the Pareto Frontier must be discretized. A set of discrete designs that is representative of the Frontier must be created. These discrete designs are then plotted on a two dimensional line graph with each design listed on the X axis and multiple Y axes each representing a different criterion as shown below for the design of a spring.

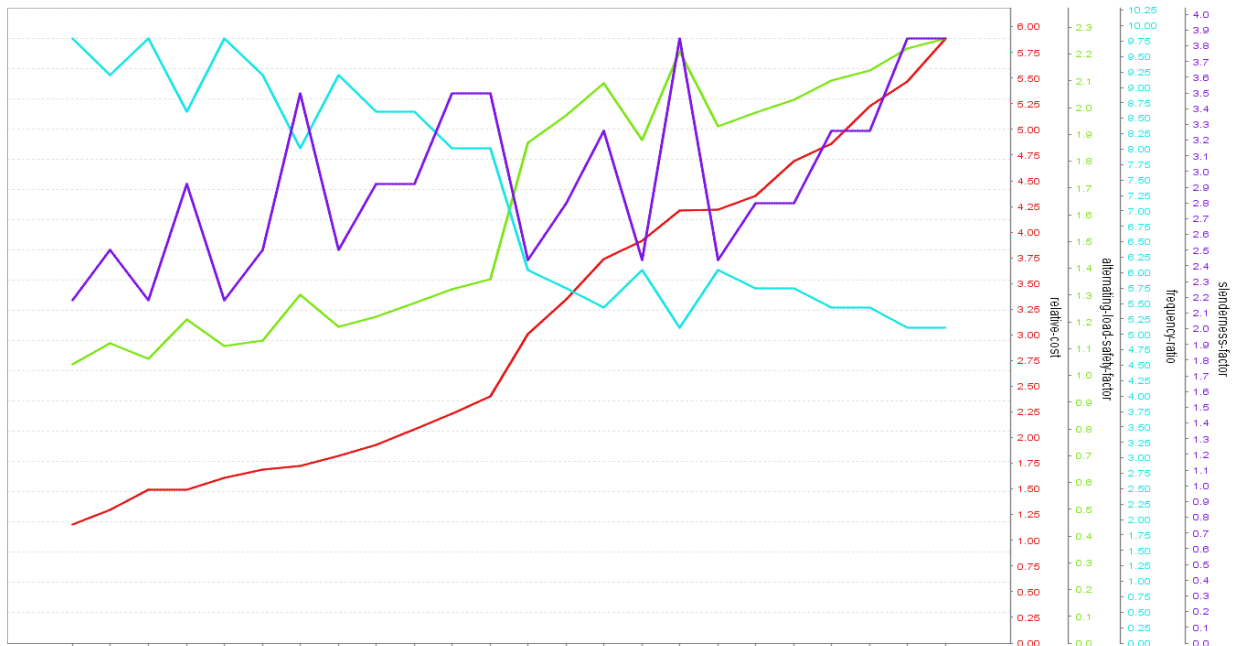


Figure 1: Helical Coil Spring Pareto Frontier Sorted by Cost

This allows us to visualize the entire design space at once so we can see the tradeoffs between the different parameters. We can then make an informed choice about the balance we want to strike between the parameters. To make the graph more readable, the scales for each of the requirements can be set up so that a higher Y value represents a better value for that criterion. This graph allows the comparison of many different designs across many conflicting criteria.

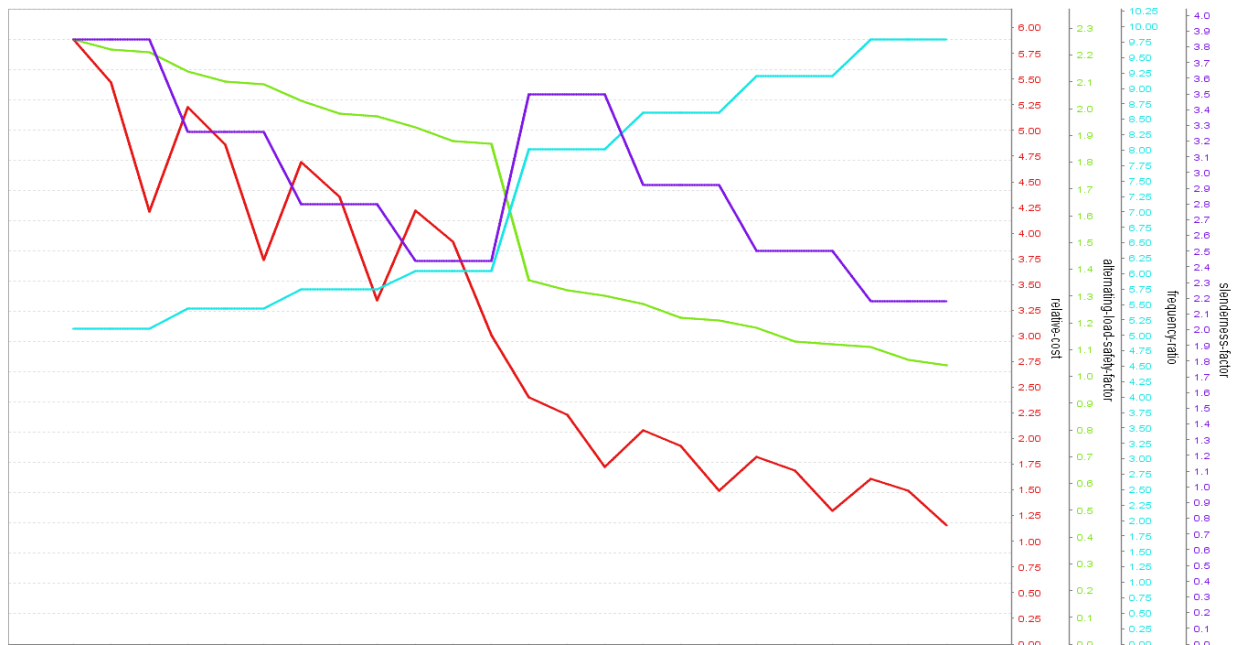


Figure 2: Helical Coil Spring Pareto Frontier Sorted by Safety Factor

Note the large jump in cost and safety factor near the middle of the graph. This jump exists because only two wire diameters remained after optimization. The less expensive designs use 0.207" wire and the more expensive designs use 0.257" wire. Examining only one design at a time it likely would not be clear that wire diameter is such a critical parameter. The combination of Pareto Optimization and visualization using the Frontier graph enables insights not available otherwise.

Linked Designs

One particularly problematic complication in implementing optimization in engineering design is that in many cases there is a requirement to use common components between different designs. For instance, air conditioners are often designed as part of a product family which has some common dimensions and components between products. Adding links between two designs greatly increases the complexity of optimizing them. If two air conditioners are designed with 10 choices for each of 10 parameters, there are 10^{10} or ten billion possible designs each. If there is no relationship between the designs, they can be examined in isolation. To finalize the design of the two products, twenty billion designs will have to be examined (ten billion to select the first product and ten billion to select the second). However, if a constraint is added that one of the parameters, say the compressor, must be the same for both designs, the designs cannot be independently evaluated. For each of the 10 choices for the shared parameter, there are now 18 parameters (9 from each linked design) that have 10 choices. Thus there are now $10 \cdot 10^{18}$ or ten quintillion unique design combinations. In many situations, there are links between many more than two designs. This situation also arises when separately optimizing components of a larger machine or building. For instance, the fan, compressor, and coil of an air conditioner may each be separately optimized as well as their combination into an assembly.

Snyder Wilson Consulting has software capable of finding optimal design combinations in design spaces with very complex networks of linked designs. The proprietary software was able to produce optimal combinations for a data set consisting of a trillion trillion, or 10^{24} possible combinations.

Visualizing in Parallel

While it is possible to use the structure of the problem to find globally optimal design combinations, large design spaces may create optimal sets too large to visualize. In these situations, a representative subset must be selected for visualization. This subset can be used to give the overall shape of the design space. Sections of the design space with preferable trade-offs can then be selected for more detailed review. This can be used in an iterative process to narrow the design space to the area with the best possible design. In this way the ideal design can be found without ever having to view the entire design space at once.

Conclusion

Frontier optimization is a powerful new design approach that can help engineers make better decisions and select better designs. Frontier optimization allows for an alternative to the cost function approach where relative weights are assigned to different parameters without the visibility that the frontier chart enables. The chart is so intuitive and powerful that even those without detailed understanding of the design may be able to contribute to design decisions. This approach allows engineers to focus less on plug and chug calculations and instead balance all the competing criteria to reach the best possible design.