Parametric Design

Parametric design (PD) uses algorithms and mathematical equations to create and manipulate digital models[1]. In parametric design, the key parameters of a design, such as size, shape, and orientation, are defined and quantified, and their relationships can be defined by mathematical formulas or rules, rather than by static dimensions or measurements. This allows designers to create complex and highly customizable designs that can be easily modified and adapted to meet different needs or specifications.

The origins of parametric design can be traced back to the development of computer-aided design (CAD) technologies in the 1960s and 1970s. As CAD systems became more advanced, designers began to explore new ways of using computers to create and manipulate digital models. In the 1980s and 1990s, researchers and practitioners in fields such as architecture, engineering, and product design began to develop new tools and techniques for parametric modeling, which allowed them to create complex and highly customizable designs that could be easily modified and adapted to meet different needs or specifications.

However, parametric design differs greatly from traditional 3D modeling and, of course, 2D CAD. It's based on set rules and parameters, according to which the design is then generated, managed, and modified. Parametric design has significant benefits that improve the design process and ensure that the ready design is of excellent quality.

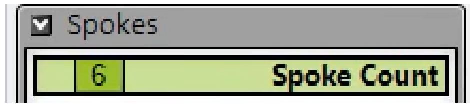
Today, parametric design is an essential part of many design disciplines, and it continues to evolve and expand as new technologies and approaches are developed. The growing use of parametric design reflects a broader trend towards greater customization, flexibility, and efficiency in design processes, as well as a recognition of the potential of digital technologies to transform the way we create and interact with the built environment.

The following sections will take you through the entire engineering design process again - this time in parametric design.

[1] Jabi, W. (2013). *Parametric design for architecture*. Hachette UK.

2.1 PD: Problem Definition

Imagine you are creating a CAD model of a car wheel design with six spokes. How would you model each spoke? You can always create a model of a single spoke, make six copies, and manually rotate each spoke based on your calculations. But what if you could drag a slider called “spoke count” to “6” and see the design being generated right in front of you?



When a designer approaches a problem in Parametric Design (PD), they may still identify the same **objectives** and **constraints**. However, they also create a **parametric schema** (Figure 7) that models the relationships between variables (e.g., the angle between two spokes). This allows them to harness the computational power of modern design software, generate new designs in split seconds simply by entering different values and combinations, and explore a multidimensional design space more systematically and efficiently.

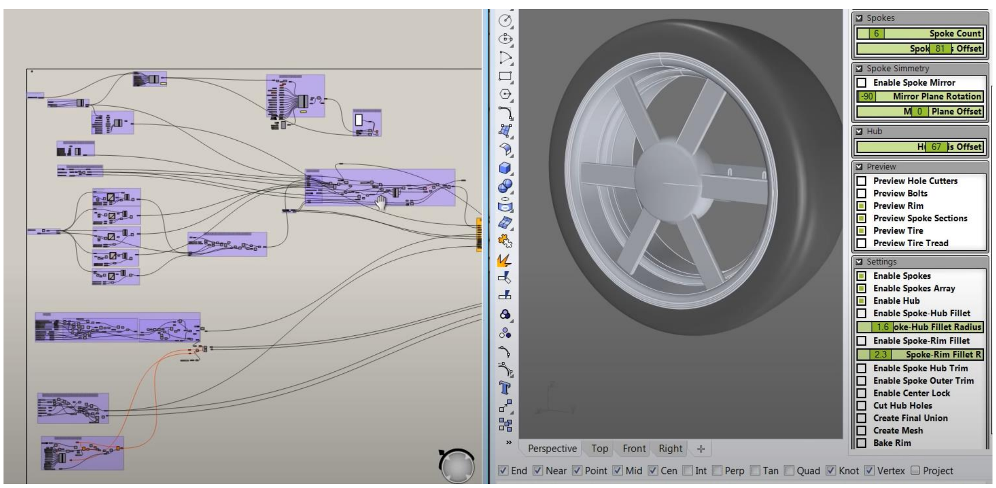


Figure 7. *Parametric design of a car wheel frame in Grasshopper. In addition to identifying the same design variables (e.g., number of spokes, width of spokes), the designer also creates a* ***parametric model / schema*** *that models the relationships between variables. (e.g., the blocks and lines on the left), so they can change the number of spokes in the design simply by dragging a slider (top right).*

To achieve this, the designer must define the design **variables** not just as vague descriptions, but rather as **quantitative and computable variables** with specific ranges. In other words, designers must write equations or use computer software to mathematically define the relationships between all relevant variables. Designers must then specify the acceptable range of values for each variable, by providing an upper and lower limit.

For an example, let's return to the car wheel design problem. If the weight is an important variable, its mathematical relationship to other variables (e.g., material(s) used, number of spokes, width of the spokes) should be expressed as an equation. Similar equations defining cost and durability might also be necessary. These and all other relevant equations comprise the parametric schema for that design problem.

The software will gladly solve equations for you, but it struggles with reading comprehension. The next practice example will give you an opportunity to define a design **variable** yourself.

Practice example: *Parametric* solar farm design

We can also define the design of solar farms with a few variables:

- **Tilt angle**: How the solar panel is oriented in the vertical direction. An appropriate tilt angle orients a solar panel towards the Sun.
  - 0º = flat (parallel to the ground)
  - 90º = vertical
  - In the photo on the right, the front panels have a lower tilt angle, and the back panels have a higher tilt angle

- **Row width**: Each row of solar panels may have multiple sub-rows. A wider row casts a longer shadow.



RW = 1 RW = 2 RW = 4

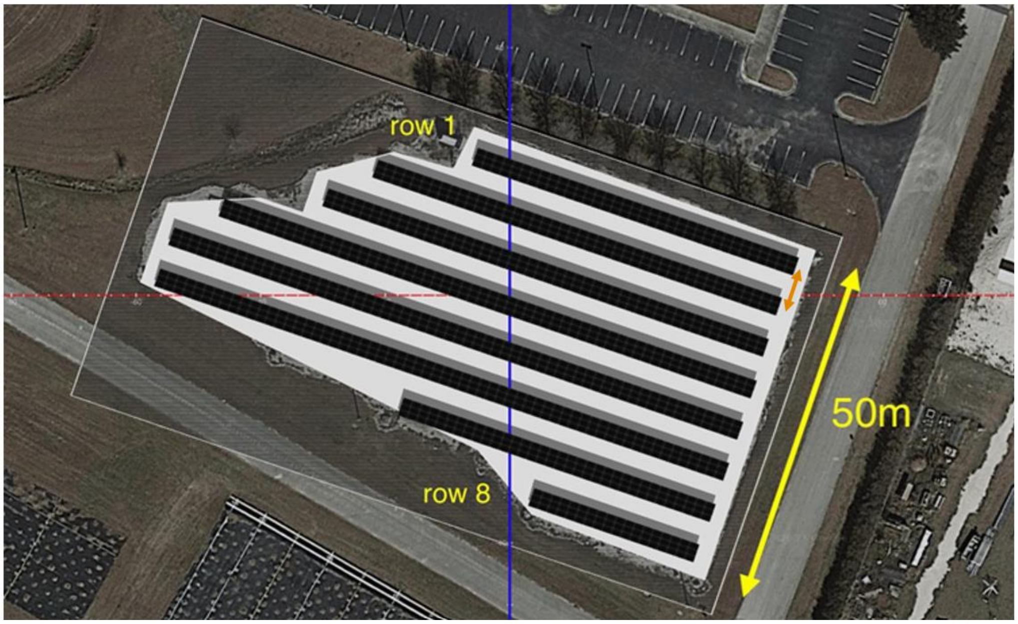
1. Because most solar farm designs follow a similar pattern, we can define another variable called "**inter-row spacing**". Look at the two examples below:

- Can you provide a definition for "inter-row spacing"?


Definition of inter-row spacing (IRS)	

2. Imagine you have sketched the solar farm design shown below:

- The width of the field is approximately 50m.
- There are 8 rows of solar panels, equally spaced throughout the field.

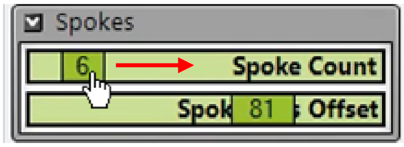


Now you need to calculate the parameters of this design:

- Can you calculate the approximate inter-row spacing of this design? (Assume the IRS is measured center-to-center, not edge-to-edge, see the orange arrow in the above figure.)
- Now that you know the IRS, can you describe a way to calculate the position of each solar panel row using its row number (1-8) and IRS? Feel free to respond with text, pseudo code, or any other format.

2.2 PD: Exploration

As a designer, having 2-3 designs is great, but having 20-30 designs is even better. For example, what if you want to compare designs with not just 6 spokes, but also 7, 8, … 20 spokes? Do you repeat the same design process 14 more times…



or would you rather drag a "spoke count" slider to generate each design automatically?

In this stage, designers use parametric modeling tools (Figure 8a) to generate and evaluate multiple design options or alternatives (Figure 8b). This may involve creating digital prototypes, testing different combinations of design parameters, and exploring different scenarios or use cases.

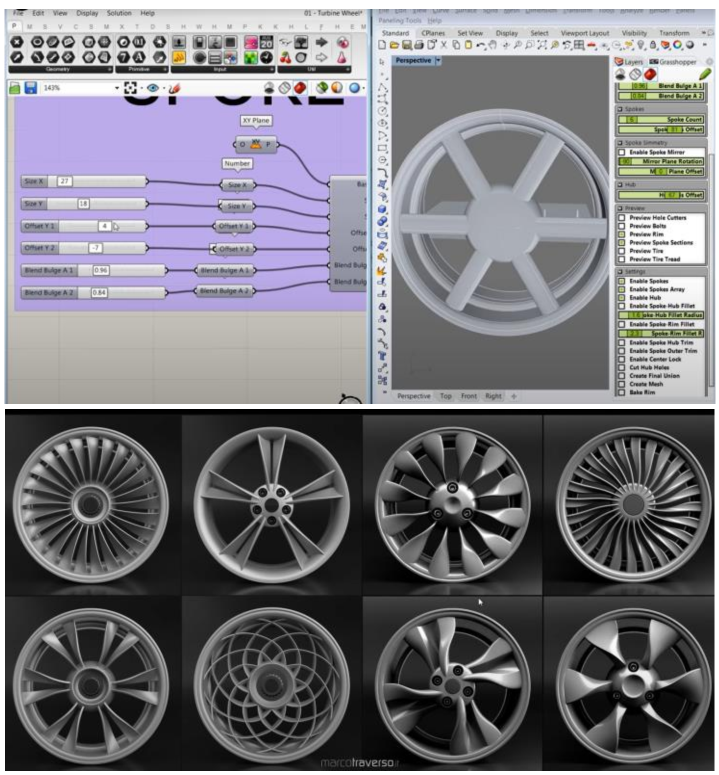


Figure 8. *(a) Example of a wheel frame design in Grasshopper to show the Exploration in PD. These methods begin from a basic model and allow the designer to explore different design possibilities by changing the parameters using sliders and then visually representing the changes to the model. (b) A selection of parametrically designed car wheels.*

Practice example: *Parametric* solar farm design

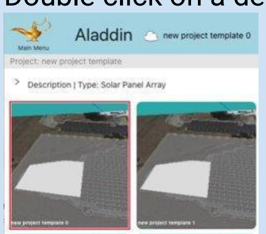
Organize Your Designs with *Projects*

You can create, view, and compare multiple designs within a single *Project.* A project template has been provided for you, which contains multiple identical copies of the Pickle solar farm design template. To create your PD designs, simply open one of the design templates in the project; you don't have to start from scratch.

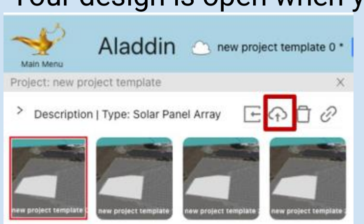
- **Open** this *Aladdin* PD project template. Make sure you are signed in to an account.

  Select Main menu> Project> Save project as 

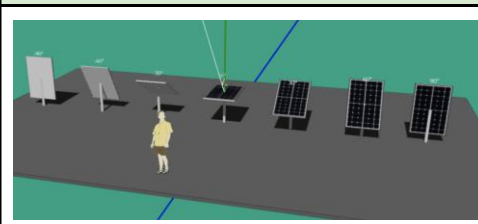
- Double click on a design in a project to open the design

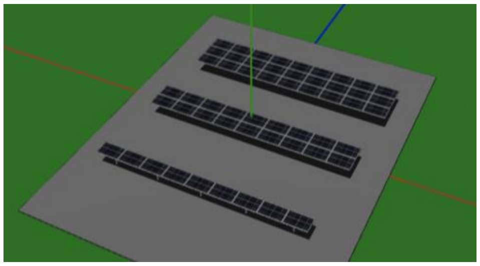
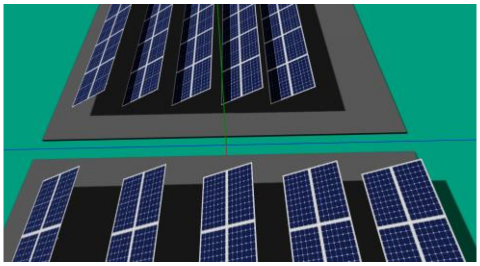


- Your design is open when you see a cloud put on the project panel



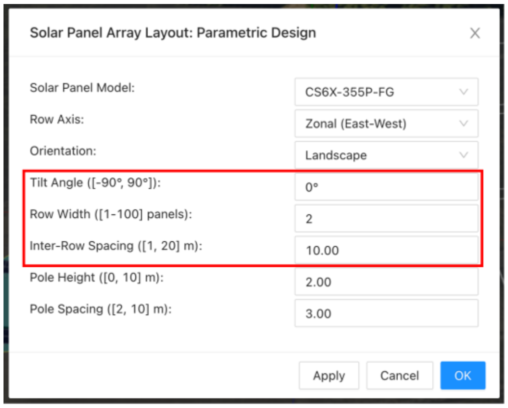
As a refresher, here are the three solar farm design variables you can change:

Property	Explanation	Example
Tilt angle	How the solar panel is oriented in the vertical direction. An appropriate tilt angle orients a solar panel towards the Sun.	

Solar panel row width (RW)	Each row of solar panels may have multiple sub-rows. A wider row casts a longer shadow.	
Inter-row spacing (IRS)	The distance between every two rows of solar panels (center-to-center). If two rows of solar panels are too close together, some panels may be shaded and produce less output.	

In traditional design, the designer manually adds solar panels and change their position and size to create the desired design. In parametric design, **you can simply enter the values** for each design variable, and a **layout wizard will generate the corresponding design for you**.

Steps:



A. **Open** your own copy of the PD project and **double click** on the first design.

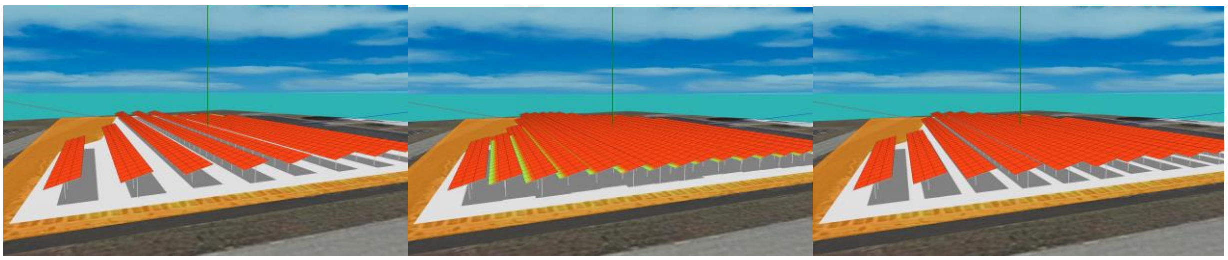
B. To open the layout wizard, **right click** on the gray area below the solar farm and **select** *"Layout > Solar Panel Array Layout"*.

C. **Change** the three design variables to the desired values and **click** "OK".

D. The layout wizard will ask you if you want to replace existing solar panels. **Click "Yes".**

E. **Update** the selected design  to save your first PD design.

One of the powers of parametric design is to explore the design space much more efficiently. For example, **what is the best IRS, if we decide to use RW=3?** After a trying a few different values, we may discover that:



- IRS=6.4 (left): There is too much space between the solar panels.
- IRS=3.2 (middle): The inter-row shading decreases the output of each solar panel.
- IRS=4.8 (right): This layout is more balanced.

Practice example: *Parametric* solar farm design

3. Use parametric design to create the example above (tilt = 15º, RW = 3, IRS = 4.8m) in your PD project.
4. Now that you have your first PD design, create two more solar farm designs by yourself and **update** them in your PD design project.
    - One design with thinner rows (RW=2)
    - One design with wider rows (RW=5)

    Find the best IRS for these two designs by trying as many IRS values as you like. [Additional tips: Keep the same tilt angle (we recommend 15º for this task), so you can compare the effect of different IRSs.]

    Use the shadow and heatmap visualization to estimate the “sweet spot.”

What is the best IRS for RW=2?	
What is the best IRS for RW=5?	

2.3 PD: Evaluation

At first sight, the **Evaluation** stage in PD looks similar: Designers evaluate the performance, feasibility, and suitability of the design options that were generated in the exploration stage. However, with the ability to create dozens (if not hundreds) of designs quickly in PD, designers can now evaluate and compare more designs, increasing their chance of finding the optimal design solution or important information that can be used in future iterations.

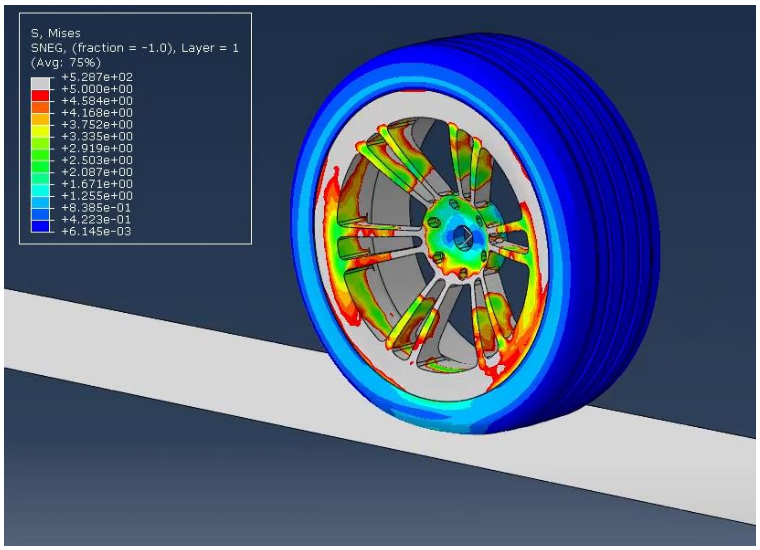


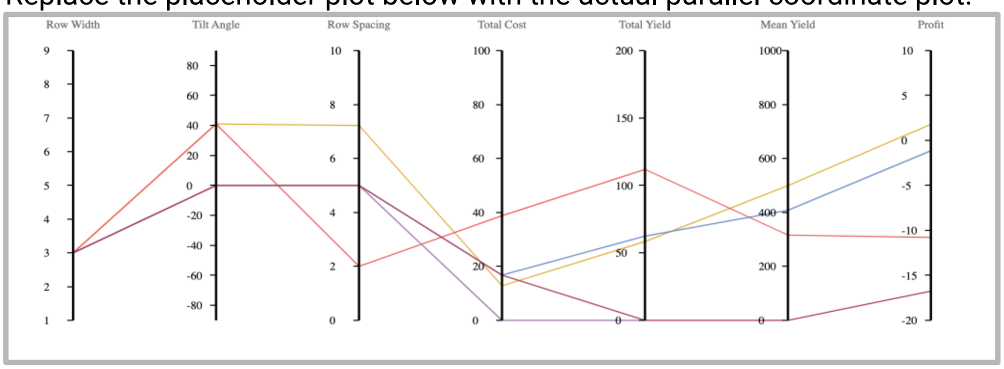
Figure 9. *Evaluation of the structural integrity of a car wheel design in CAE software. Even though designs can now be generated from a set of design parameters, designers still need to evaluate each design afterwards.*

Practice example: *Parametric* solar farm design

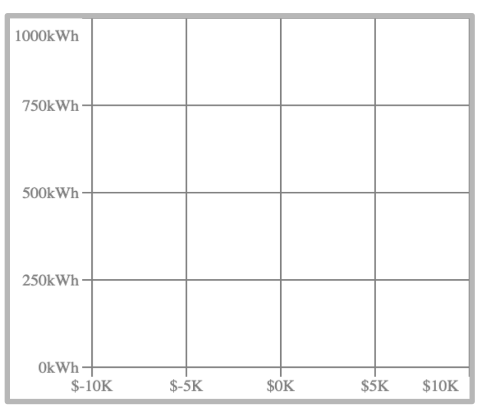
5. Now that your PD designs have been saved to your project, do they actually perform as well as you expect? Follow the instructions below to evaluate your designs.

> **Organize Your Designs with *Projects***
>
> - Double click the first design in your project to open it.
> - Analyze the yearly energy output ("Main Menu > Analysis > Solar Panels > Analyze Yearly Yield").
> - Update the design  in the project. Make sure the design is both open and selected. You should now see the updated objectives (such as total yield, profit, and mean yield) in the parallel coordinate plot below.
> - Repeat these steps for all designs in your project.

a. Replace the placeholder plot below with the actual parallel coordinate plot.



b. Generate a scatter plot with "Profit" as the X-axis and "Mean Yield" as the Y-axis, and replace the placeholder plot below with the actual scatter plot. [Note: you can change the scatter plot and parallel coordinate plot axes ranges by double clicking on the variable name in the parallel coordinate plot and changing the minimum and maximum values.]



6. Which design has the most yearly profit (and is therefore non-dominated)?

7. Which design has the most yearly average output (and is therefore also non-dominated)?

8. Is the remaining design non-dominated (i.e., better in at least one objective than any other design)? Or is it dominated by any design?

2.4 PD: Iteration

Imagine you have evaluated your car wheel design and want to make a couple changes. Maybe you want to add 2 spokes, reduce their width by 5%, and switch to a different material. Would you start from scratch, or would you rather change some numbers and drag some sliders?

In this stage, designers refine and improve the selected designs by adjusting design parameters (Figure 9) through a series of iterative steps. This may involve incorporating feedback and insights from testing and evaluation into the design, making adjustments or modifications to address any issues or concerns, and then repeating the process until a satisfactory solution is reached. The goal is to create a final design that meets all of the project requirements and that has been optimized for performance, efficiency, and user experience.

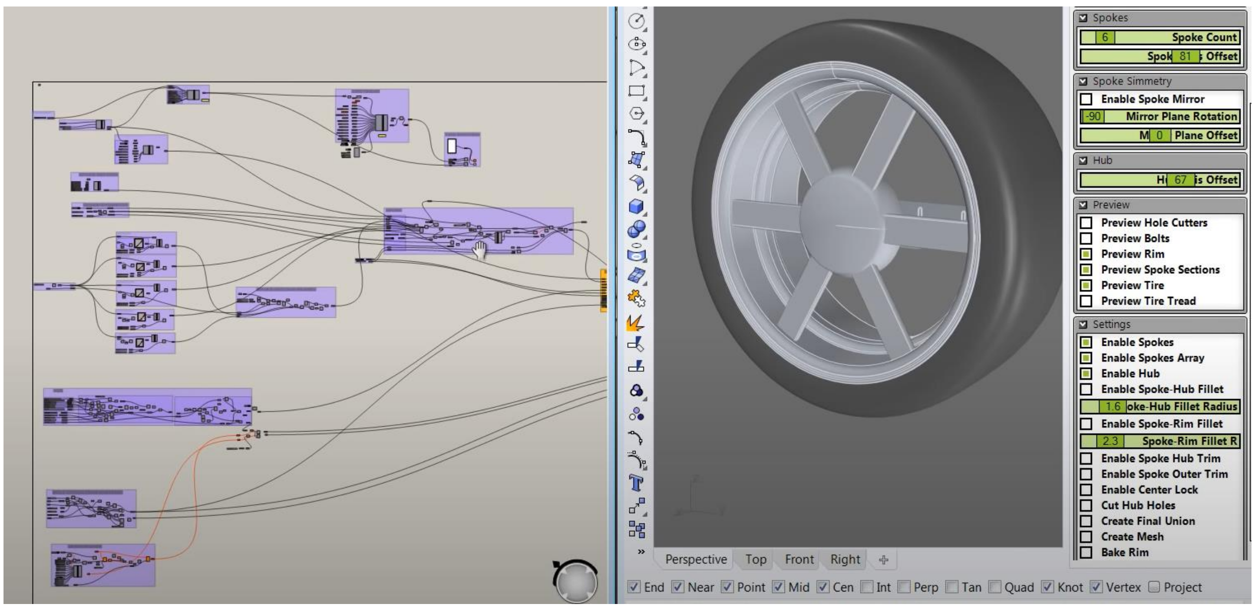


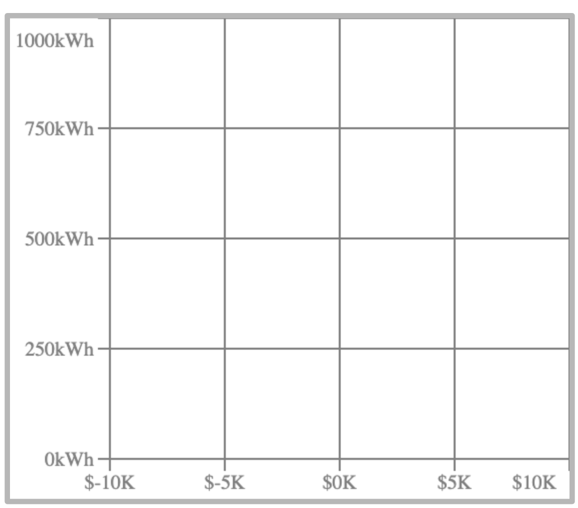
Figure 10. *Example of a wheel design in Grasshopper to show the Iteration in PD. As mentioned previously, iteration can happen within design steps or between steps (e.g., Problem Definition, Exploration, and Evaluation). For example, the problem might be redefined based on the evaluation results. In this case, compared to TD, which is more focused on directly adjusting the geometries of designs, iteration in PD might involve modifying the parametric models (parameters, relationship of parameters) and changing the parameters.*

Practice example: *Parametric* solar farm design

9. Now that you have evaluated your first design, you may decide that an IRS of 4.8m is still too much spacing. Try to add one more row of solar panels to design 1 and maintain an equal IRS without using PD (i.e., make all the changes manually). Spend no more than 5 minutes on your attempt and insert a screenshot of your final progress below. (The goal is for you to experience the difference between TD and PD.)

10. You have a few options with your other two designs:
    - Converge: Choose what you think is the most promising RW so far (2,3, or 5) and optimize the tilt and IRS.
    - Diverge: Try other RWs first, before choosing what you think is the most promising RW and optimizing the tilt and IRS.

    Iterate a few more times using parametric design, **generate** at least three (3) more designs, and **curate** (duplicate) them to your GD project.
    Once you finish all your designs, replace the placeholder plot below with a scatter plot of Profit (X-axis) vs. Mean Yield (Y-axis).



1. Which of these designs are now non-dominated?

2. Which design achieves the best balance between yearly profit and yearly average output? Can you briefly explain why in 1-2 sentences? (The answer can be subjective.)

Open-ended problem: Parametric Design

A southwestern University has announced that they plan to develop a solarized parking lot on campus. They have tasked you with proposing designs for this solarized parking lot using Parametric Design in Aladdin. Click here to open the Aladdin design template. The goal is to maximize profit and yearly average output.

Submit as many designs as you can. To leave spacing for utility vehicles, the University has requested a minimum of 7m Inter-Row Spacing, and a minimum of 3.5m Pole Height.

Use the design space graph below as a guide. Place your designs in the design space by inserting links or screenshots.

Yearly Average Output (kWh)				
>=500: Exceeds expectations	*[insert link here]*	*[insert link here]*	*[insert link here]*	*[insert link here]* *Aim at here!
>=475: Meets expectations	*[insert link here]*	*[insert link here]*	*[insert link here]*	*[insert link here]*
>450: Approaching expectations	*[insert link here]*	*[insert link here]*	*[insert link here]*	*[insert link here]*
<450: Not meeting expectations	*[insert link here]*	*[insert link here]*	*[insert link here]*	*[insert link here]*
Profit ($)	**<2,900:** Not meeting expectations	**>=2,900:** Approaching expectations	**>=3,100:** Meets expectations	**>=3,300:** Exceeds expectations

Give a brief description of your design process and discuss the best design(s).