Cornell Cup USA presented by Intel
Decision Matrix Guide

Attempting to objectively determine the best design to pursue or even just the best component to purchase can at times can be a significantly challenging. Convincing others or your boss that your end decision is indeed the "best" one can be downright frustrating. Decision matrices offer a formal means for declaring at the beginning of the decision process what are the most important aspects to consider in making a decision. Then once those aspects are established, they can be used to objectively justify which choice is the best and provide a measure of how much better and why.

You may have seen variations of decision matrices before. They are commonly used by many government agencies such as NASA and are a corner stone of many trade-off studies. An example of a decision matrix is also provided in IntelCornellCup_SampleDecisionMatrix.xlsx and in fact, the review criteria for this competition are an example themselves of one kind of decision matrix. This guide provides a step by step explanation on how to create such a decision matrix and the pros and cons of various options in setting up your own matrix.

For introductory purposes, the steps presented here focus around the selection of a component that many people are familiar with, a new laptop. However once understood, these same steps can utilized towards objectively making a variety of decisions. It is not anticipated that you use decision matrices for the majority of your decisions, but they are a great tool when there are potentially many good choices and selecting the best one is very important to your project.

Like many of the guidelines, you are encouraged to read through all the steps first as understanding each step and how they all fit together in the end can help you to perform each of the steps better. You may also find that you want to perform some of the steps in a different order. For example, Step 6 can be done as early as just after Step 1 and is sometimes better to do it that way to help ensure impartialness.

Step 0: Find some examples of potential candidates/options for your desired component. This does not have to be an exhaustive list but a variety of candidates can be helpful. In fact, not all of the candidate components need to completely meet all of your projects needs, i.e. you may think "this would be great, if it only had *X*". The goal of this step really to start to develop an intuition as to what makes a component great and what you notice as potentially lacking from the candidates as well.

Step 0 is Complete When: you have a list of potential candidates for your desired component. You and your teammates do not yet have to agree as to which candidates are better than others.

Step 1: Decide upon the attributes that you would judge a good choice on. This step is one of the most important and should be given some in-depth thought. Attributes can come from the challenge needs, the performance measures, the interface requirements of subsystems, and the overall functionality that the component is helping to achieve. Think about the potential candidates you selected in the last step and why you thought they were good candidates. The goal however is not necessarily to determine every possible attribute nor is the goal to necessarily find the best option overall but rather the attributes that will help you find the best option for your needs. For example, the hard drive may be an important characteristic for many laptop selections, but for this example it is not as important (perhaps an external hard drive has already been selected or is being selected separately)

Step 1 is Complete When: you have a list of attributes that all major players involved with the decision have agreed to. At this stage you do not yet have to agree with what attributes are more important than others.

Step 2: Determine how you will measure each of the attributes. This is similar to how you create performance measures for a challenge's need. In many cases, particularly for components, this can be more straight forward. For the laptop weight, as an example, it may simply be the number of pounds. However for some seemingly simple attributes it may be harder to determine an objective measure. For the laptop example, the graphics attribute could be measured across several different technical criteria but just like selecting attributes, you want to be able to measure the attributes quality in a way that will mean something to your needs. Therefore, in the laptop example a set of criteria is established that the graphics quality of a laptop must meet in order to earn a specified score. (Please see the sample file IntelCornellCup_SampleDecisionMatrix.xlsx to follow along with these steps.)

The key to making a set of good objective criteria is that regardless of the reviewer that is measuring the option using your set of criteria, the score awarded to that option is the same; or at least has as little variance as possible between reviewers. A common way to make criteria objective is to set *scoring levels*; a set of specific measures with thresholds and depending on thresholds are met, the option earns a different score, i.e. an option must do/meet A,B, and C in order to earn a 1, or it must do/meet D,E, and F in order to earn a 2, etc. The key to assigning a score is that all of that score's thresholds must be met in order for that score to be awarded. So to extend our example, if an option met D, E but only met C instead of F, that option would only earn a 1 because it didn't meet all of the requirements to earn a 2.

Typically a 1-3, 1-5, or a 1-7 scoring level system is best to establish. 1-10 is sometimes used as well but 1-5 or even 1-3 is far more common and often better because it can make for more separation between thresholds which in turn can make your scoring more consistent. For example, if 3 reviewers were asked to judge something on a 1-10 scale, they might rate it a 4, a 5, or a 6, but if asked to rate the same thing on a 1-5 chance all three reviewers might give it 3, and if they were asked to rate it on a 1-3 scale most likely all reviewers would give it a 2. The goal sometimes is then to make fewest number of levels necessary in order to make the kind of distinctions that are needed.

Making the highest score something that is hard to earn can be a good way to help make sure you separate out the good from the great. At the same time, however, also make sure that your highest & lowest ratings have meaning to your needs; for example if any laptop that has a weight under 2 lbs will completely meet all your needs, there is no use in rewarding a 1 lb laptop over a 2 lb laptop because it will make no difference or at least practically no difference to your overall project needs.

In some cases, the criteria may even require small test(s) be completed. In the laptop example, the graphics criteria requires that 5 test programs be run. Although the disadvantage is that it usually takes longer to run a test versus looking up a property like the weight, the advantage is that the criteria is specifically matched to your needs.

Step 2 is Complete When: a means for measuring each of the attributes is established. In some cases this may require a scoring system. All major players involved with the decision have agreed to the measuring methods.

Step 3: Establish min and max values for the attribute measurements where relevant. In some cases you may know already that any component that has a certain value above or below a given threshold will never be able to meet your needs. The most straightforward example of this might be cost. Any component exceeding a given cost would be above your available budget and no matter how good it is otherwise you would not be able to use that component.

Step 3 is Complete When: min and max attribute measurement values have been established. Please note that not all or even none of your attributes may require min or max values.

Step 4: Determine all of your potential options and the attribute values for all of your potential options. In some cases you may not be able to find all of the exact values, or it may require you to purchase some of the options to find all of the values you need. In these cases, it is often best to provide your "best guess" and perhaps lean towards the lower scoring side of your guesses. Later on if the option you guessed about fairs well in the other attribute areas, you can decide whether the real values might influence your decision significantly and whether it is worthwhile to investigate these options further.

In assigning attributes' values via a scoring system like the one created for the graphics category, the rule to follow is that all of a single score's criteria must be met in order for that score to be assigned. For example, consider a potential option that was tested and it was found that the "maximum resolution is obtained at the highest frame rate for 3 out of 5 sample programs" which meets the criteria for a score of a 4, however for the remaining 2 sample programs they could only be run at the "2nd largest resolution at the 2nd highest frame rate" which meets only the criteria score of a 3. In this case, the attribute is assigned a value of a 3 since not all of the criteria of the 4 were met.

It is also highly recommended that you do not try to average the half-met criteria scores and give a 3.5 instead. Not only is this more complicated and harder to ensure that multiple reviewers will be consistent across their scoring, if you want to make these kinds of distinctions you are better off splitting the attribute into 2 or more so that you can potentially weight each attribute differently (as will be shown in Step 6).

Step 4 is Complete When: when you have a matrix of all of the options you are considering and all attribute values for those options.

Step 5: Normalize the attribute values. It can be hard to compare attribute values directly such as pounds of weight, to hours of battery life, to gigs of hard drive space. Furthermore if some attributes are best when they have a high value and some that are best when they have a lower value, like potentially cost or weight, it can be even harder to make direct comparisons. Therefore it is convenient to be able to normalize the attribute values to a 0 to 1 scale.

If you have a scoring system for all attributes, such as the 1-5 scoring system mentioned in step 2, it can be quite easy to do this. For other cases however, the sample decision matrix provides 5 different method examples for normalizing.

Example 1: Battery Life: normalization occurs by taking the largest attribute value and then dividing all of the options' attribute values by this largest value. This gives the option with the largest attribute value a normalized value of 1 and all others will be less than or equal to that normalized value. The first step in doing this first is make sure *all of the measurements have the same units*. In this case, hours were chosen as the common unit. Option A also had the largest attribute value of 15 hrs so all options' battery life values were divided by 15.

Example 2: Cost: in this situation the lower the score, the better and the maximum score is a positive number and minimum score is a zero. In this case, normalization can again occur by taking the largest attribute value and then dividing all of the options' attribute values by this largest value. Then as an additional last step all values are subtracted from 1. This gives the option with the largest attribute value a normalized value of 0 and all others will be greater than or equal to that normalized value.

Example 3: Weight: determine a baseline value and then normalize the attribute values according to their distance from the baseline. In this example, a baseline of 3 lbs was chosen and being below the baseline is good and being above it is not good. The first step after ensuring *all of the measurements have the same units* is to determine the difference of each option's attribute value from the baseline. Next determine the range of values from the baseline; an easy way to do this is to take the absolute value of the maximum difference from the baseline, in this case 2 lbs and then double it because we will be considering values both above and below the baseline. Then similar to what was done in earlier examples, divide all of the differences by the maximum range, i.e. twice the absolute maximum difference. This will place all values between -0.5 to 0.5, so in order to get all values on a 0 to 1 scale, simply add 0.5 to all values. Finally, in this case because lower values are better than higher ones, subtract all values from 1 as was done in example 2.

Example 4: CPU: develop a scoring system as was done for the attribute measurements in step 2. In this case a 0 to 1 scoring system equivalent to a 1-5 scoring system was created. Similar to how values were assigned in step 4, the scores are assigned to the options. Actually a similar process can be applied to easily translate any scoring system of step 2 into a 0-1 normalized scale. It is useful sometimes to make the scoring a part of the normalization process so that the actual attribute value(s) can be recorded as a part of the matrix.

Example 5: Memory: normalize to the sum of all values. This provides a more relative comparison as to how a single options compares to the options overall. Other similar methods are to take the mean or median value and use that as the baseline value for example 3. In this case however, all attribute values across a row of the decision matrix are summed and then all attribute values across that row are divided by that sum. This technique works best when there are fewer options to consider, i.e. the points are not spread too thin and potentially diluted compared to other attributes' values. It is not as commonly used but it is shown just as another example.

Each of these example normalization methods can potentially influence the way the data will be interpreted and it is important to ask yourself if you are accidentally favoring the data. For example, if applied the method used in example 1 and the highest value is considerably higher than the rest, say at 1000, and the rest of the options have values between 1-10, are you really being fair in your normalization if all your system behaves very well for anything with a value above a 7? In this case perhaps establishing a scoring system may be the best approach.

Step 5 is Complete When: all attribute values have been normalized to the same range. All the examples here are on a 0-1 scale but it could have just as easily been done to a 0 to 25 or a -1000 to 1000 scale.

Step 6: Assign a weight, a.k.a. an importance, to each attribute. These weights represent how important each attribute is relative to each other. Typically weights are first assigned on some kind of scale, 1-5 or 1-10 are very common ones. The weights will only be assigned once for the entire decision matrix so the 1-10 weight scale does not have the same issues that were noted in creating the scoring systems of step 2 or step 5.

More than one attribute may receive the same weight but the most important thing is that the weights are agreed upon by all players involved with the decision making process. The weights should be decided upon without taking into account the attribute values or normalized values. Otherwise, you may run the risk of potentially inserting your personal influence so that your favorite will "win".

This would defeat the purpose of the decision matrix and for this reason many times weights are decided upon right after step 1. They are presented here as step 6 as they have not been absolutely needed until now and they will need to be used in step 7.

As a final part of setting the weights, the weights themselves are often normalized, where all of the weights are summed down the column and then each weight is divided by that sum. This will make every weight into a percentage and the sum of these normalized weights will be to 1.

Step 6 is Complete When: All of the attributes have been assigned weights along the same weight scale (i.e. 1-5, 1-10, etc.) and all players who have a say with the decision making agree on the weights.

Step 7: Calculate the weighted scores for all of the options. For each option, multiply their normalized attribute values by each attribute's respective weight. Then sum all of the resulting values for that option to produce that option's weighted score.

Step 7 is Complete When: all options' weighted scores are calculated.

Step 8: Interpret the weighted scores to select your chosen option(s). The simplest interpretation of the end results is that the option that has the highest score is the winning option that should be selected. However, the question of how much it won by is often worth investigating. A common way of analyzing this question is to take a certain percentage of the top score, say perhaps 10% to 20%, and then any other option that is within that percentage of the top score should also be given at least a second look. If there are no other options within that percentage you can feel more confident about your solution. Otherwise, if there are a couple or more options that are close to being the top you may want to closely examine some of the original attribute values, particularly those that may have been estimated.

Overall at this point, it is a good idea to see whether any estimates made early on could have influenced the end results or decision. Close results may also provide good supporting evidence that more resources be used to determine more accurate attribute value results.

Another way to answer this question of what other options, if any, you should investigate further is to do so graphically. Plotting the options versus their scores in order from highest scoring option to lowest, can typically produce a sloping off curve. The point in the curve where there is visibly a sudden change in the score can be a good cut-off point where those above the cut-off point should be investigated further.

Any option with weighted scores that are better than or equal to the weighted scores of another option for every attribute is said to dominate that lower scoring option. Furthermore, since the dominating option is equal to or better than the lower scoring option in every way, the lower scoring option can be eliminated entirely from consideration. However if you have already done the work to evaluate the lesser option, it can be worthwhile to leave it in the decision matrix so you can least show objectively why that option was not chosen.

Step 8 is Complete When: A final option has been selected, or at least a results supported plan for further investigation has been established.

It is not uncommon that different projects you may work on in the future may require many of the same components but just have slightly different priorities. For this reason, it can be very helpful to hold on to decision matrices as they can often be re-used or at least partially reused where perhaps only a few scoring systems or the weights ratings needs to be swapped out for new ones to better represent the new project's priorities. This is can be a significant time saver in the long run and if not too many scoring systems or weights need to be changed it can even potentially help provide some consistency between project decisions.