EML2322L – Design and Manufacturing Laboratory

Tips for Selecting Good Decision Matrix Objectives

Importance

Concept selection using decision matrices is often the most intensive part of the design process, but also the most important. For this course, the effort invested in this phase of the project will return dividends in the form of truly selecting the best ideas among all of your team's concepts. In industry, being able to create proper evaluation matrices to aid in decision making and concept selection is an invaluable skill that will set you apart from your peers.

Good Objectives

Correct objective justification requires a clear definition of the objective being evaluated, a clear explanation of why this objective is important for the project's success, an explanation of how the objective is assessed, and a justified weighting factor for the objective. If using qualitative objectives, be sure to consider how the objective(s) will be tested. Common objectives that should appear in all matrices in EML2322L are: manufacturing time, cost, and modularity (or size).

Bad Objectives

Unfortunately, decision matrix objectives are often poorly defined or constructed. Common problems include: testing criteria or methods that do not correlate well with the objective being tested, objectives that do not differentiate between individual designs in a meaningful way, and objectives that have little relevance to the actual performance of the subsystem.

1. An example of a poor objective for a ball or bucket manipulator is speed. While it makes perfect sense to assess speed for a mobile platform subsystem, doing so for a manipulator rarely provides meaningful insight. A much better assessment would be the time required to manipulate the object, as this allows direct and quick comparison of the concepts under consideration.
2. An example of a poor objective for a ball launching mechanism is shooting speed when evaluated on a linear scale. Many shooting mechanism concepts consist of wheels attached to high speed motors that fire balls along a track towards the target, similar to baseball pitching machines. The intention of considering shooting speed is to ensure the balls will have enough energy to reach the target, which is an important objective to evaluate for a ball launcher. However, regardless of the selected motor/wheel combination, several concepts may have the ability to fire a ball more than fast enough to hit the target. But, since the objective is assessed simply on the speed at which the ball is launched, the highest speed motor would be awarded more points in that category, even though it does not offer ANY added benefit over other motor/wheel combinations of similar design. Instead, groups should use a more results-based testing procedure *(e.g. a test where they launch 5 balls with each design concept and count how many fall)*. This way, the objective differentiates between other types of designs where ball launching energy is a notable issue (like a gravity-fed mechanism) without differentiating between designs that have comparable performance.

Dependent Objectives

Another common issue is objectives that are too similar to each other; that is, two objectives that evaluate the same characteristic or feature of the design. For an effective decision matrix, a design's performance in any given objective should be independent (or as independent as possible) of its performance in other categories.

3. A common example of dependent objectives can be found in the manner in which speed and controllability are evaluated for a mobile platform. Groups typically use their motor/wheel combination to evaluate speed, and some form of obstacle course to evaluate controllability. However, oftentimes the obstacle course is composed in a way consistent to *"drive forward 5 meters, around a few buckets, and back, where the fastest time wins"*. The performance of a design in this controllability assessment will be extremely correlated with (i.e. dependent on) its performance in the speed assessment due to the testing method. A much better way to evaluate controllability would be to design an experiment (e.g. an obstacle course) to measure controllability where the mobile platform never has the opportunity to reach higher speeds that would bias the controllability objective being assessed.

Objectives Erroneously Evaluated

Sometimes objectives are just erroneously evaluated, which is why it's important to decide on objectives as a group and to proof each other's work very thoroughly.

4. An example of an erroneously evaluated objective is when groups define modularity the same way as the mobile platform matrix example (e.g. the number of fasteners removed to disassemble a subsystem for storage each week in the project box), but then evaluate it as the total number of fasteners in the subsystem.

Objective Weighting Problems

In many instances, objective weighting problems stem from the performances of the evaluated designs in their objectives. For example, mobile platform designs commonly perform similarly on objectives such as manufacturing time. Therefore, the weighting assigned to this objective should not be very high, since the objective does not provide profitable information for the selection of the design. It does not make sense for something like this to be weighted too heavily, since doing so makes it more difficult for other objectives to determine the superior design. The weighting factors also need adequate justification in the objective definitions, which includes a clear explanation as to why each objective is more or less important than others.

Also, if objectives given for one subsystem are measurable for others, they should usually be included in those matrices as well. For example, if groups specify weight as an objective for a ball hopper because they want to reduce the overall weight of their robot, then weight should be included for all other subsystems as well. If the goal of the objective is to reduce the overall weight of the design, why should it only be important for one system? In the previous example, if the group defines the need for weight to be an objective because it benefits the individual performance of that subsystem somehow *(not because you are trying to decrease the overall weight of the robot)*, then it does not need to be included in other system's matrices, even though it is still a measurable quantity for those components.