As talked about last meeting I can not directly compare my solution with existing technologies due to differences in what my algorithm does, and therefore I will split up the research questions of my thesis into 2 parts:

1. I will compare my kano-matrix updating part of the algorithm against fully regenerating the kano matrix as they did in their first paper.

***What is the difference in time cost between an incremental update of the reachability matrix compared to newly generating the reachability matrix for every event and how does this difference scale with pod/policy numbers***

Hypothesis: The new generation will outperform the updating approach for pod creation and deletion events but will be slower for the network policy creation and deletion events since this both involves changing the layout of the matrix and revisiting the corresponding networkpolicies. This difference will be more apparent when upscaling the amount of pods and policies

***What is the difference in space cost between an incremental update of the reachability matrix compared to newly generating the reachability matrix for every event and how does this difference scale with pod/policy numbers***

 Hypothesis: The difference between the two methods will be very small to the point it that it will be negligible

A table with colorful text

Description automatically generated

Green = Variables; Blue = Constant; Pink = Not applicable

Extra variable: Type of event

* 1. Generate given amount of random pods and policies
  2. Generate start kano matrix as baseline
  3. For each event type:
* Generate the event (e.g. random new Policy)
* Look at time for kano regeneration upon event
* Undo event
* Look at time for incremental update upon event

  Same strategy for space cost

Please not that I will only focus on the first kano paper and as proposed mention the second paper and its pros, cons and effects for my thesis in the discussion

1. I will measure the overhead that my full solution brings in terms of space and time consumption.

***What is the relationship between pod/policy numbers and the time cost of conflict detection?***

***What is the relationship between pod/policy numbers and the space cost of conflict detection?***

Setup: like previous experiment, but with the full solution (so including conflict detection)