The simulationDriver for cantera is being upgraded to the YAML way because it will allow surface reactions to be modified on the fly.

To convert to the YAML way, we have two things to convert:

1. The input files
2. The functions.

The functions are actually easier to convert. However, we need to convert the input files to the YAML way first because the old cti way does not create the right reaction object type (in cantera python objects) for on-the-fly modification.

\*\*\*

First, let’s consider how the CTI file gets made so that we can switch it to the YAML way. Actually, we can just, for now, use the CTI to YAML converter at the very end of full cti file creation because we already know that will convert to a suitable YAML file.

The current flow basically occurs inside a single function, as seen inside example\_runfile.py, it is the function call: canteraSimulate.create\_cti\_and\_SimulatePFRorTPRwithCantera

That function takes a CTI top info file (like ceO2\_cti\_top\_info.cti ) which has the species names and phases, and it also takes an array of *reactions and parameters*.

It passes the information to another function, create\_cti\_and\_cantera\_phases, which creates the cti\_file using canteraKineticsParametersParser.create\_full\_cti and then goes on to create the phases using cantera directly.

Those phases are then passed back to canteraSimulate.create\_cti\_and\_SimulatePFRorTPRwithCantera which then also calls the simulation function.

So to interrupt that flow and introduce YAML while allowing backwards compatibility, there are a few obvious ways to do it.

1. Create a function create\_yaml\_and\_cantera\_phases, make that function then call a helper function called canteraKineticsParametersParser.create\_full\_yaml
   1. Make that function then