`initiate.py`

64 sloc

START

```
main()

    VARIABLES

START_VALUE &&
END_VALUE &&
STEP <-- manual edit
```

for `quantity` = CoR, the values of coefficient of restitution are modified in ss.par, which is opened and written to in `get_dynamical_time.py`, so the ability to do that is in that script.

make_directories(START_VALUE, END_VALUE, STEP)
Create simulation directories named by the value of the quantity, each directory has its own simulation files.

os.makedirs(path) FOR `path` in list of
`values <-- [START_VALUE, START_VALUE + STEP, ..., END_VALUE]`

copy_files(START_VALUE, END_VALUE, STEP)
Copy parameter files from a sample simulation directory (named default) to all the simulation directories.

```
files <-- [list of static parameter files for each simulation || rubber pile
files IF quantity = CoR]

FOR value in values <-- [START_VALUE,...,END_VALUE]

    FOR file in files DO shutil.copy(default/file, val/file)
```

END   YES   `quantity = CoR`?

NO

change_values(START_VALUE, END_VALUE, STEP)
Open rpg.par in each simulation directory and edit the value of varying quantity to the matching value of the directory.

```
FOR value in values <-- [START_VALUE,...,END_VALUE]
    open({value}/rpg.par)
    data <-- readlines()
    data[line_number][value] <-- val
    writelines(data)
```