Parametric Study of

> $CO + O \rightarrow CO_2$

`parametric_study.py`

> This script runs the `apothesis` on different `input.kmc` files by taking different y_{CO} and y_O

Script code

```python
import os
import shutil
from decimal import Decimal

stoichiometric_coeff = Decimal(0.9) # yCO = 0.1
stoichiometric_coeff_step = Decimal(0.1) # Increment by 0.1 each time
step = 2

while (stoichiometric_coeff <= Decimal(0.9)):
    file_content = f"""
    #Build the lattice
    lattice: SimpleCubic 100 100 10 A

    #The growing film

    growth: CO2

    #Create steps (alaways in the x-direaction - see above
1st number), number of steps

    #(e.g. if 10 each step will consist of two atoms),
height difference (each step will increase by 20, so 1st
step: 10, 2nd: 30, 3rd, 50 etc)
    #steps: 2 1
    #Total time in seconds
    time: 1000

    #T in Kelvin
    temperature: 1000

    #P in Pascal
    pressure: 101325

    #Random number initialization
    #random: 123434432

    #Simple s0*f*P/(2*pi*MW*Ctot*kb*T) -> Sticking
coefficient [-], f [-], C_tot [sites/m2], MW [kg/mol]

    #A + * -> A*: simple 0.1 2.0e-4 1.0e+19 0.032

    #Constant is a constant value in [ML/s]

    CO + * -> CO*: constant
{Decimal(stoichiometric_coeff).prec()} #0.389

    #Constant is a constant value in [ML/s]

    O2 + 2* -> 2O*: constant {Decimal((Decimal(1.0) -
Decimal(stoichiometric_coeff)) / Decimal(4.0))} all #0.611
    #A reaction
    CO* + O* -> CO2*: constant 1.e+15

    #Example of growth reaction. The 1sr reactant is tranformed to the 1st product, the 2nd reactan to the 2nd product etc.

    #If #Reactants > #Products then only the first Nth reactants are tranformed to the (N-x) reactants.

    #Products that have "*" stay on the lattice (adsorbed)

    #products that do not have "*" are assumed to desorb instantly.

    #CO* + O* -> CO2* : constant 0.25e+5

    #Time to write in log
    write: log 0.1

    #Time to write the lattice heights & species
    write: lattice 10

    #Report the coverage of certain species. The time will follow the write in log file

    report: coverage CO* O*
    """

    with open("input.kmc", "w") as f:
        f.write(file_content) # Write the new input.kmc file
    os.system(f"./apothesis") # Run apothesis on the new input.kmc
    # Store the log file into the output file

    shutil.copyfile("Output.log",
f"parametric_study/output{step}.txt")

    # Increment the stoichiometric coefficient

    stoichiometric_coeff += stoichiometric_coeff_step
    step += 1
```

> This script uses an already present `input.kmc` format to tweak it's constant values to different values using a for-loop.

Output cleaning done

> I used a python script to take the log file and filter out.
> The `outputCleaner.py` is shown below

```
num_files = 9 # Number of output files to read
first_line = 18 # First line of the output file to read

for i in range(num_files):
    # Empty the output file
    op_f = open(f"outputCleaned{i+1}.txt", "w")
    op_f.close()

for i in range(num_files):
    f = open(f"output{i+1}.txt", "r")
    lines = f.readlines()
    count = 1
    for line in lines:
        if count < first_line:
            count += 1
            continue
        op_f = open(f"outputCleaned{i+1}.txt", "a")
        # Write the first, second, last and second last
columns

        op_f.write(f"{line.split()[0]} {line.split()[1]}
{line.split()[-1]} {line.split()[-2]}\n")

        op_f.close()
        count += 1
```

Results obtained

At the end of the run

Time Step	y_{CO}	y_O	Growth Rate	Coverage CO	Coverage O
521.151286357571	0.1	0.225	0.000000	0.563200	0.436700
162.249006918567	0.2	0.2	0.000000	0.723500	0.276400
96.2229559039675	0.3	0.175	0.000000	0.840000	0.159900
48.324643413304	0.4	0.15	0.000000	0.922700	0.077200
35.6641534122616	0.5	0.125	0.000000	0.968800	0.031100
24.5319763553839	0.6	0.1	0.000000	0.988000	0.011900
16.5580035178922	0.7	0.075	0.000000	0.993800	0.006100
12.2896587652807	0.8	0.05	0.000000	0.996200	0.003700

Time Step	y_{CO}	y_O	Growth Rate	Coverage CO	Coverage O
9.96456773722922	0.9	0.025	0.000000	0.999500	0.000200

Graphs obtained over time of the run

Script used to obtain plots

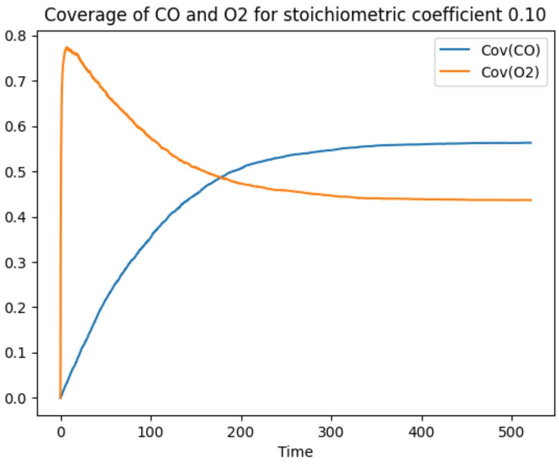
```python
import pandas as pd
from decimal import Decimal

num_files = 9 # Number of output files to read

for i in range(num_files):
    df = pd.read_csv(f"outputCleaned{i+1}.txt", sep=" ", header=None)

    df.columns = ["Time", "GrowthRate", "Cov(CO)", "Cov(O2)"]

    # Plot the coverage of CO and O2

    res1 = df.plot(x="Time", y=["Cov(CO)", "Cov(O2)"], title=f"Coverage of CO and O2 for stoichiometric coefficient {round(Decimal(0.1 + i * 0.1), 2)}").get_figure()

    res1.savefig(f"coverage{i+1}.png")

    # Plot the growth rate

    res2 = df.plot(x="Time", y="GrowthRate", title=f"Growth rate for stoichiometric coefficient {round(Decimal(0.1 + i * 0.1), 2)}")
    res2.get_figure().savefig(f"growthRate{i+1}.png")
```

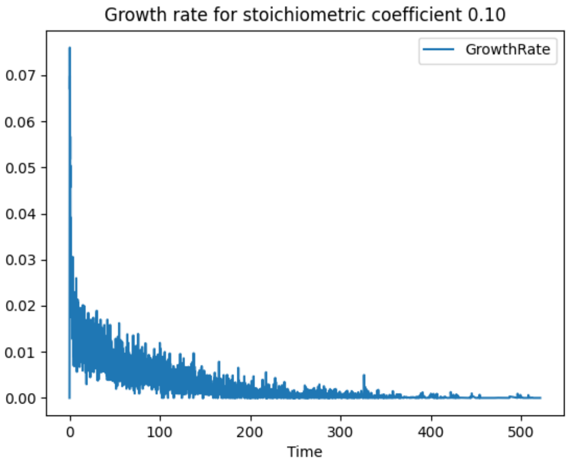
Inference from plots

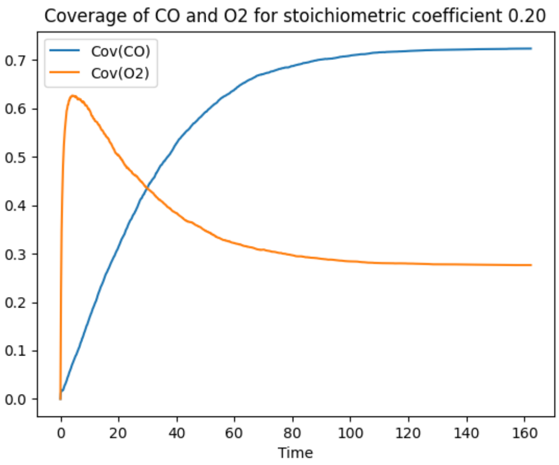
- The growth rate quickly grows to a maximum and then slowly goes to zero over time.
- The coverage of `O` increases with increasing stoichiometric coefficient.
- The magnitude of growth rate is low initially, increases to a kind of a maximum and then decreases again, perhaps closer values of Y_O and Y_{CO} give a higher growth rate or there exists an optimal value of the growth rate.
- Coverage of `CO` increases initially to a maximum and then slowly drops down to 0, the magnitude of this maximum decreases with it's stoichiometric coefficient.

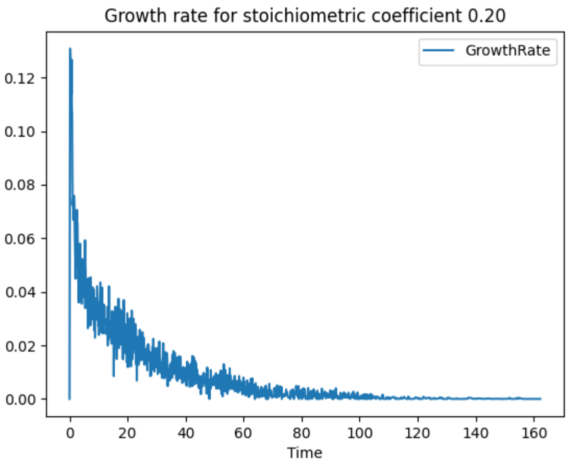
Plots obtained

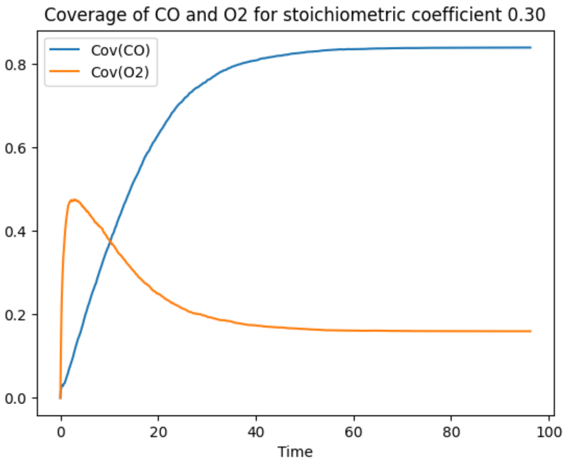
> Incase the plots aren't visible, kindly turn the page over.

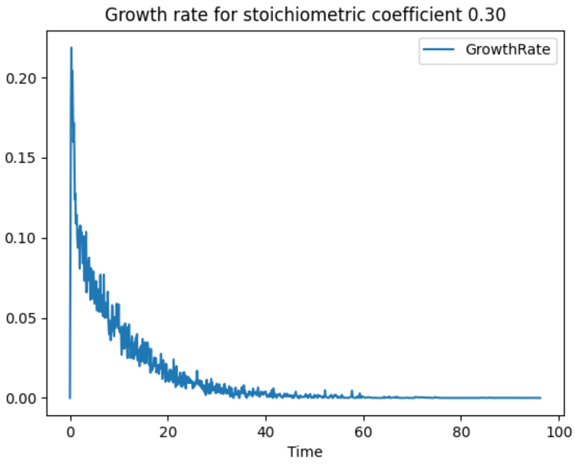

Coverage of CO and O2 for stoichiometric coefficient 0.10

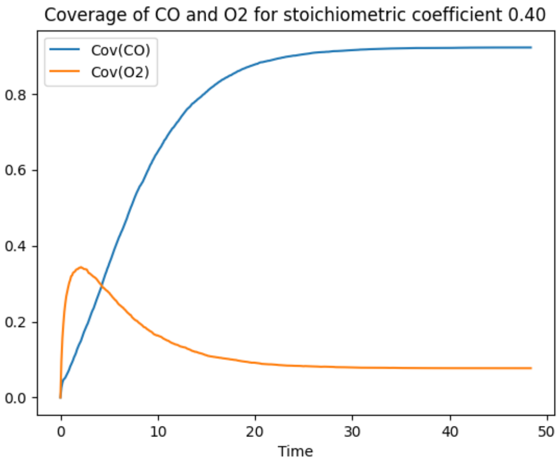

Growth rate for stoichiometric coefficient 0.10

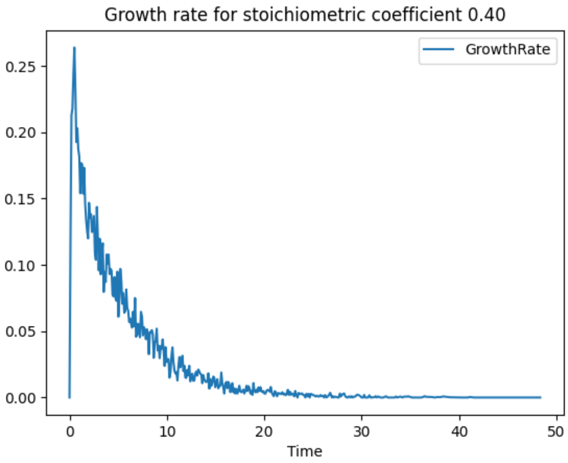

Coverage of CO and O2 for stoichiometric coefficient 0.20

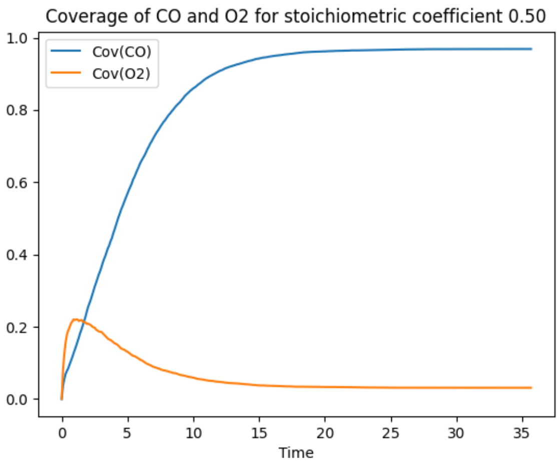

Growth rate for stoichiometric coefficient 0.20

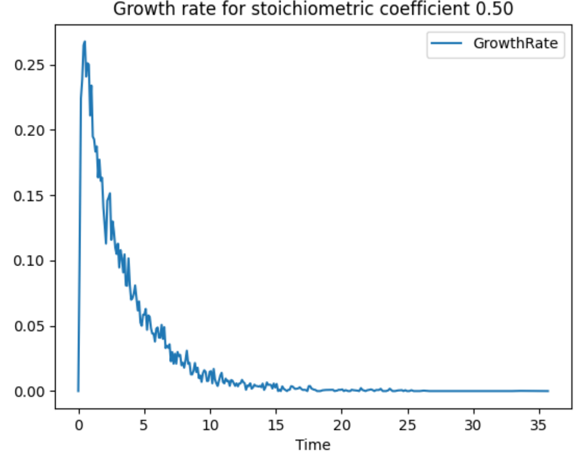

Coverage of CO and O2 for stoichiometric coefficient 0.30

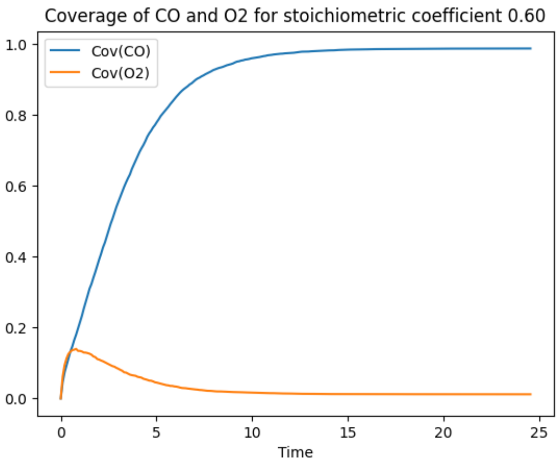

Growth rate for stoichiometric coefficient 0.30

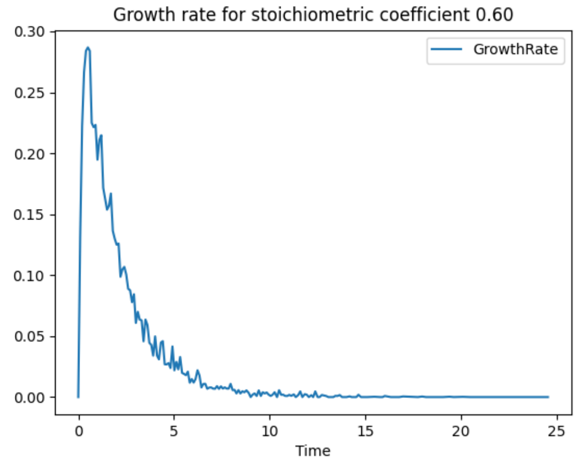

Coverage of CO and O2 for stoichiometric coefficient 0.40

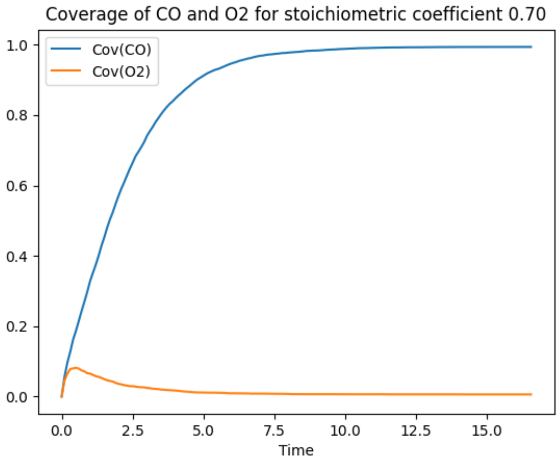

Growth rate for stoichiometric coefficient 0.40

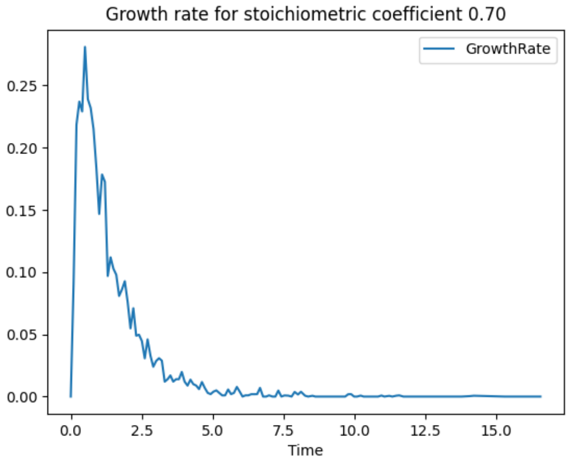

Coverage of CO and O2 for stoichiometric coefficient 0.50
Cov(CO)
Cov(O2)
0.0
0.2
0.4
0.6
0.8
1.0
0
5
10
15
20
25
30
35
Time

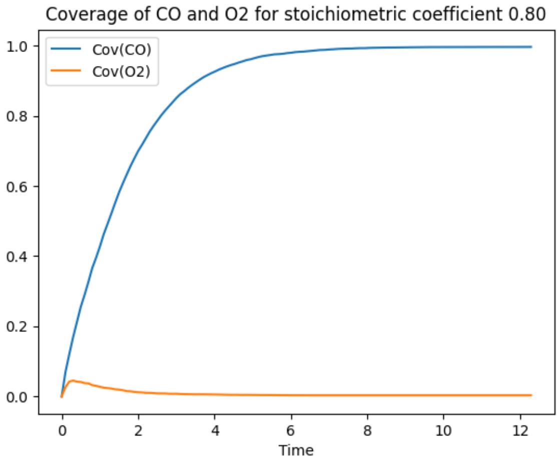

Growth rate for stoichiometric coefficient 0.50
GrowthRate
0.00
0.05
0.10
0.15
0.20
0.25
0
5
10
15
20
25
30
35
Time

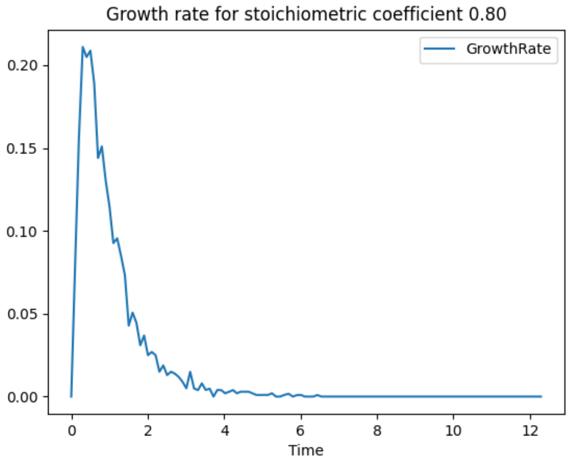

Coverage of CO and O2 for stoichiometric coefficient 0.60
Cov(CO)
Cov(O2)
0.0
0.2
0.4
0.6
0.8
1.0
0
5
10
15
20
25
Time

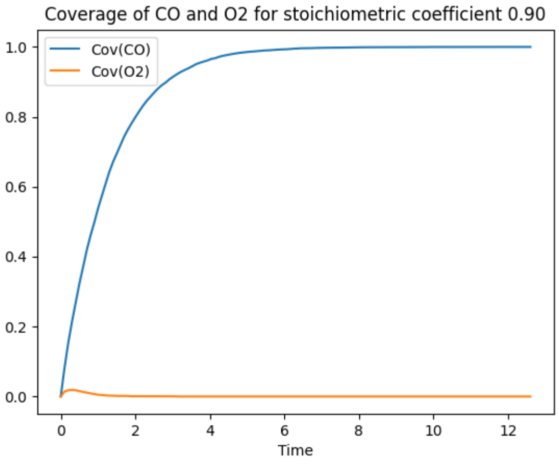

Growth rate for stoichiometric coefficient 0.60
GrowthRate
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0
5
10
15
20
25
Time

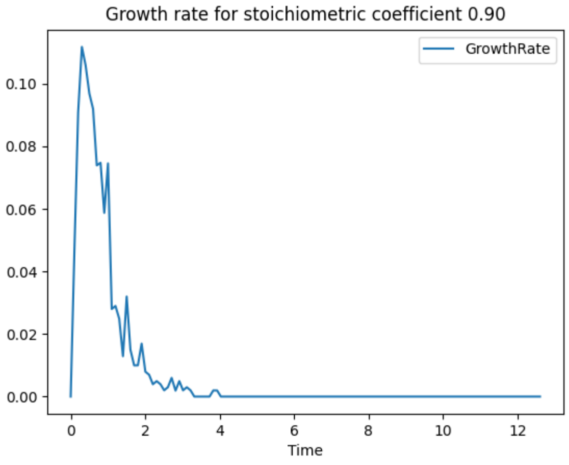

Coverage of CO and O2 for stoichiometric coefficient 0.70
Cov(CO)
Cov(O2)
0.0
0.2
0.4
0.6
0.8
1.0
0.0
2.5
5.0
7.5
10.0
12.5
15.0
Time


Growth rate for stoichiometric coefficient 0.70
GrowthRate
0.00
0.05
0.10
0.15
0.20
0.25
0.0
2.5
5.0
7.5
10.0
12.5
15.0
Time


Coverage of CO and O2 for stoichiometric coefficient 0.80
Cov(CO)
Cov(O2)
0.0
0.2
0.4
0.6
0.8
1.0
0
2
4
6
8
10
12
Time


Growth rate for stoichiometric coefficient 0.80
GrowthRate
0.00
0.05
0.10
0.15
0.20
0
2
4
6
8
10
12
Time


Coverage of CO and O2 for stoichiometric coefficient 0.90
Cov(CO)
Cov(O2)
0.0
0.2
0.4
0.6
0.8
1.0
0
2
4
6
8
10
12
Time


Growth rate for stoichiometric coefficient 0.90
GrowthRate
0.00
0.02
0.04
0.06
0.08
0.10
0
2
4
6
8
10
12
Time