
Dynamic LCA

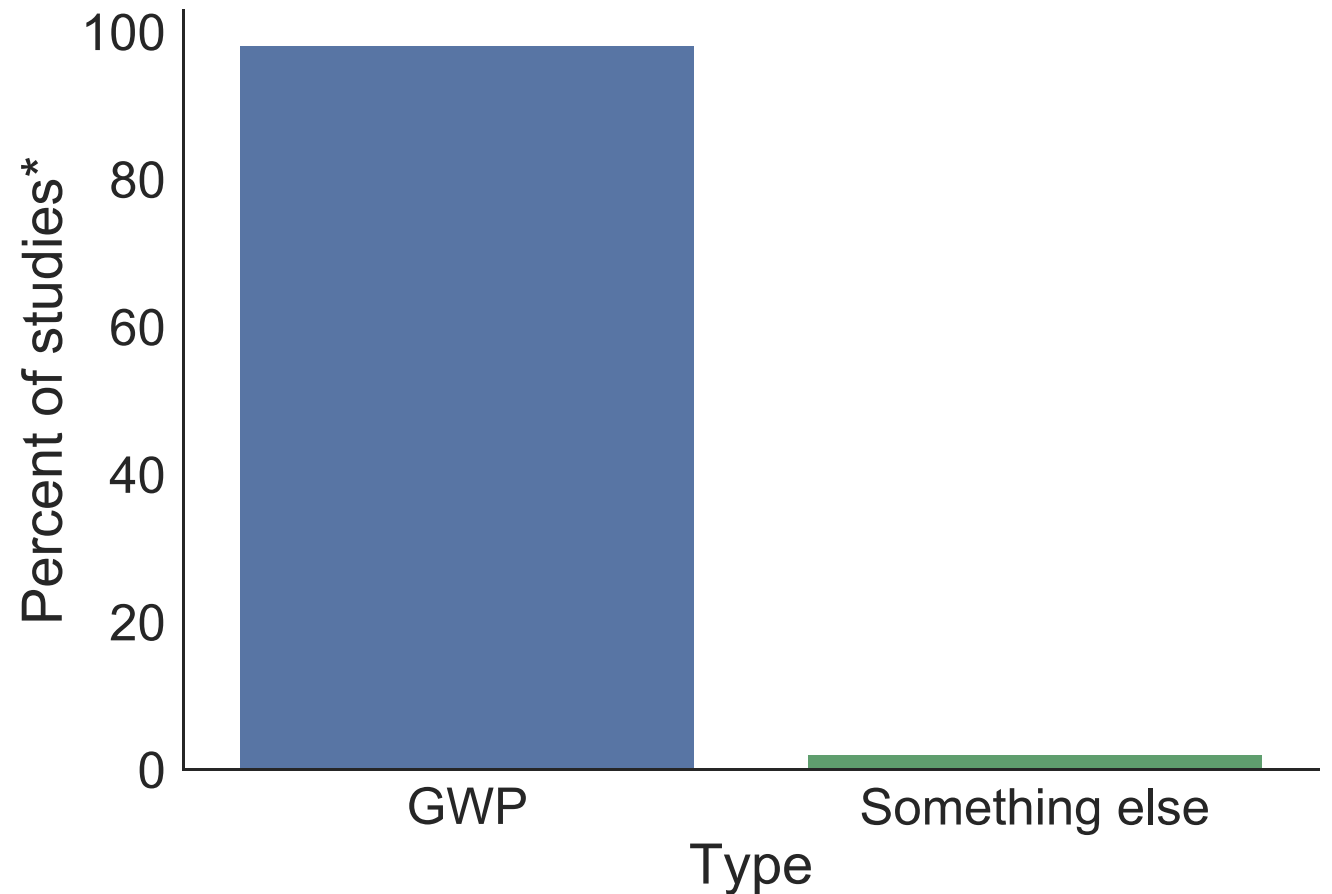
ghgforcing – an open source Python package

Greg Schivley
64th LCA Discussion Forum
March 30, 2017

Most LCA studies use GWP for climate impacts

Why?

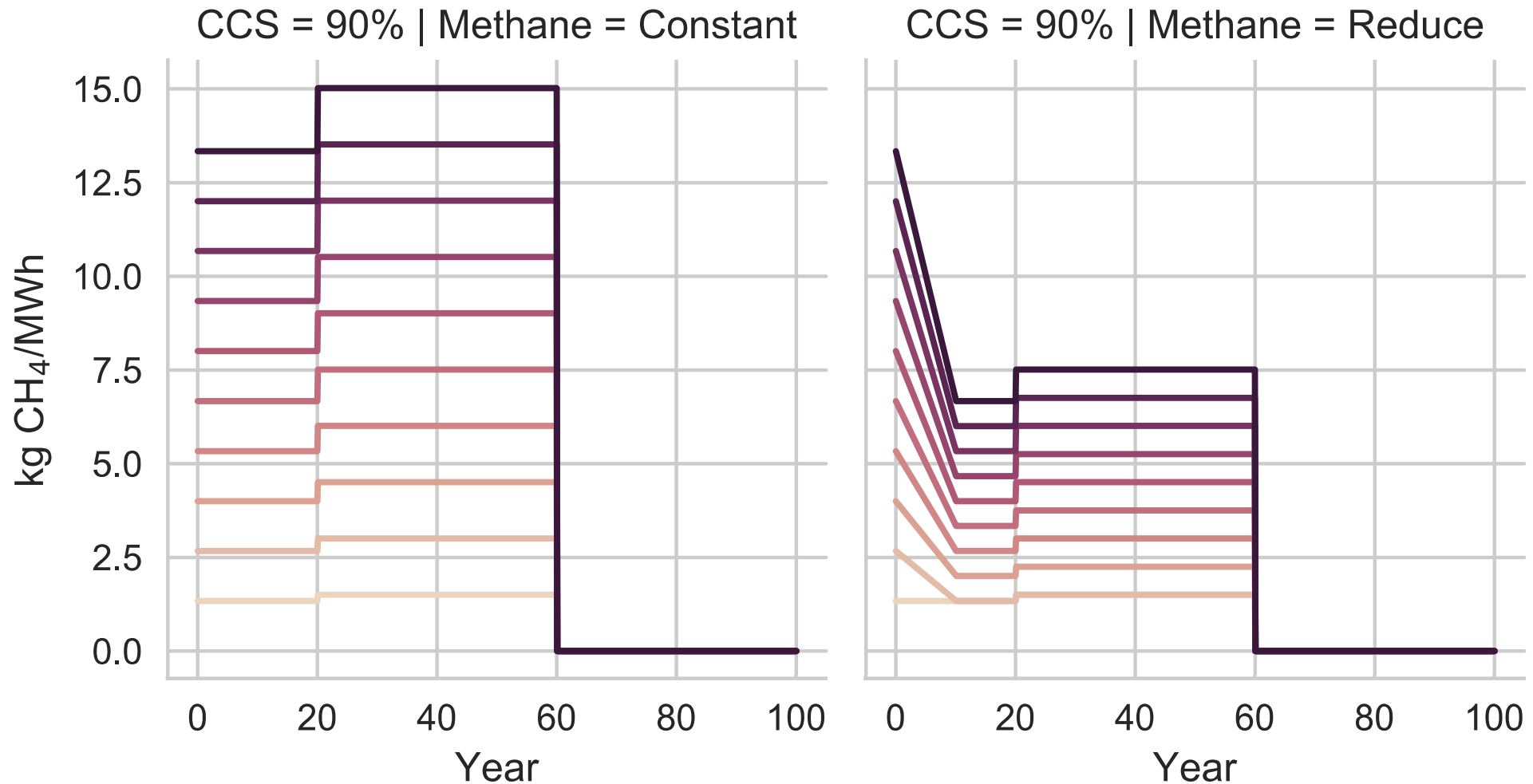
1. Multiplication is easy
2. It's good enough for everyone else
3. All emissions are constant or in a single pulse**
4. Don't make my life more difficult



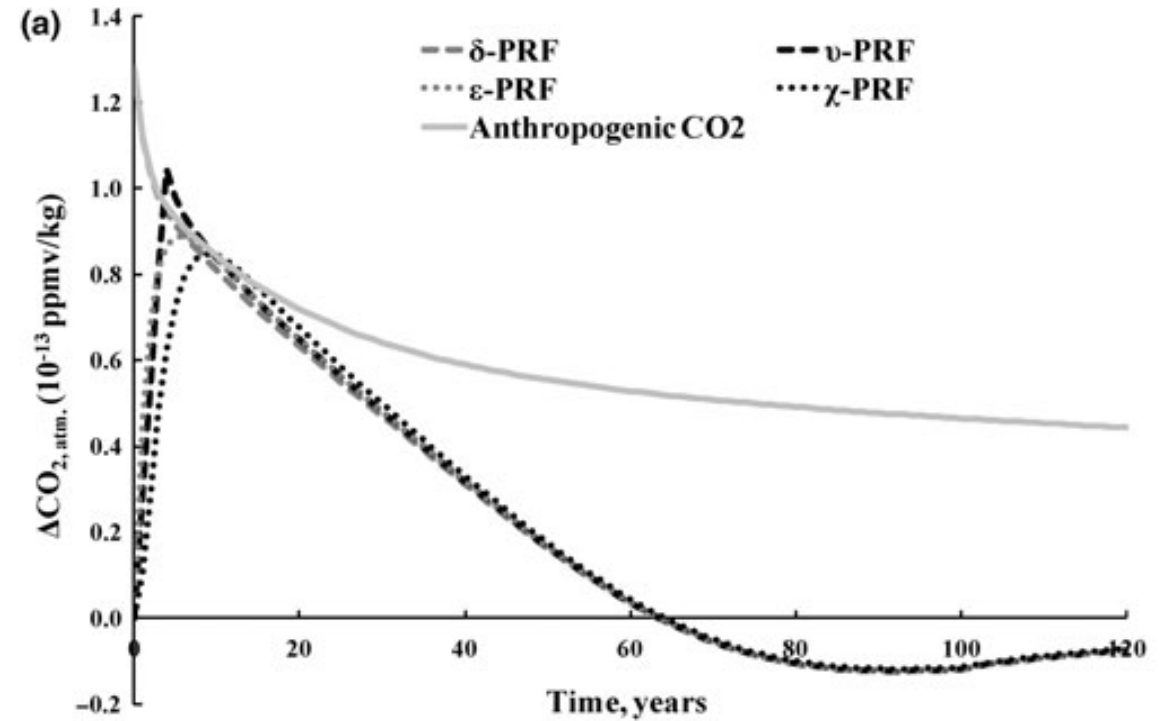
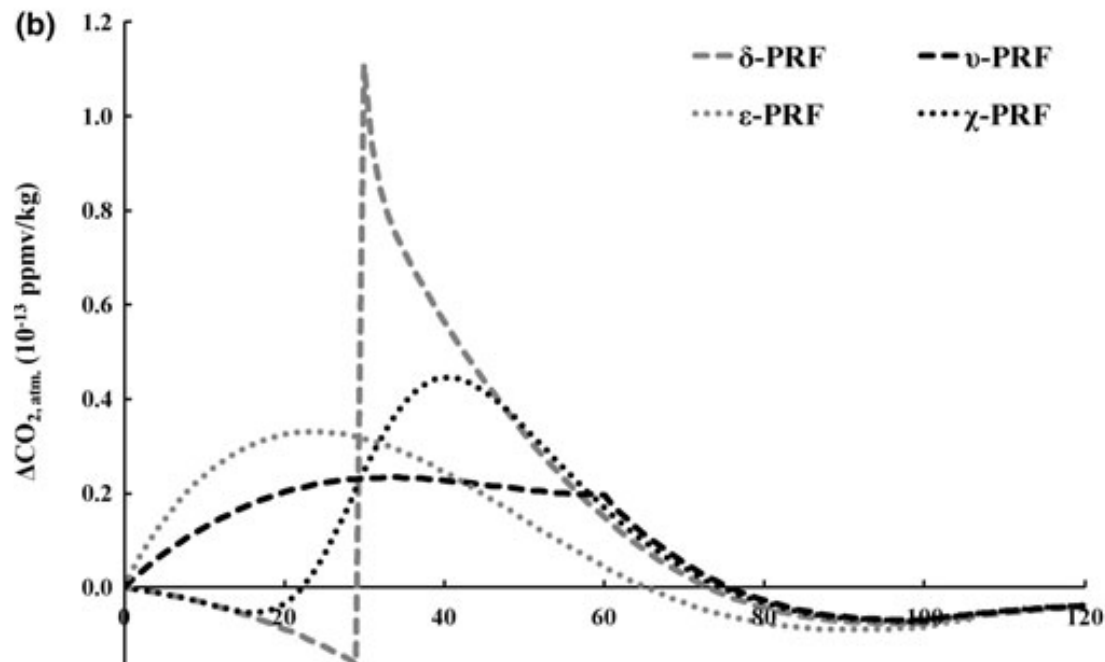
*Not real data

**Not true

Emissions happen over time, might be variable



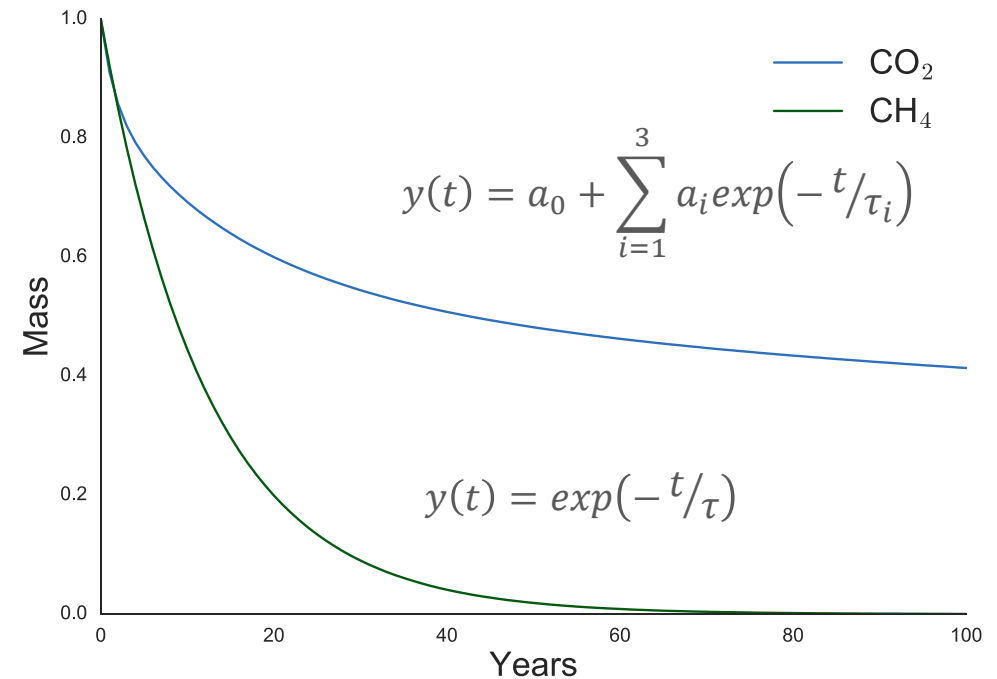
Biogenic uptake and emissions at different times



Calculation of radiative forcing

$$\text{RF}(t) = \text{RE} \int_0^t g(t')y(t - t')dt'$$

- Every emission decays according to a response function
- Mass of a species in the atmosphere is calculated using a convolution of the emission and response functions (above)



Introducing ghgforcing

- Open-source Python library
 - github.com/gschivley/ghgforcing
 - `pip install ghgforcing`
- Calculates RF, CRF, temperature for CO₂ and CH₄

Policy Analysis Review

Identifying/C
Reduction S

Greg Schivley^{*†}, W

Enhancing life cycle impact assessment from climate science: Review of recent findings and recommendations for application to LCA

Annie Levasseur^{a,*}, Otávio Cavalett^b, Jan S. Fuglestvedt^c, Thomas Gasser^{d,e}, Daniel J.A. Johansson^f, Susanne V. Jørgensen^g, Marco Raugei^h, Andy Reisingerⁱ, Greg Schivley^j, Anders Strømman^k, Katsumasa Tanaka^l, Francesco Cherubini^k

ntial

it Metrics
atural Gas

ima, Derrick Carlson,

ghgforcing basic parameters

- Array of `emissions`
- Array of `time` (same length as `emissions`)
- kind of calculation (`RF`, `CRF`, or `temp`)

- Returns: results on annual basis over length of `time` array

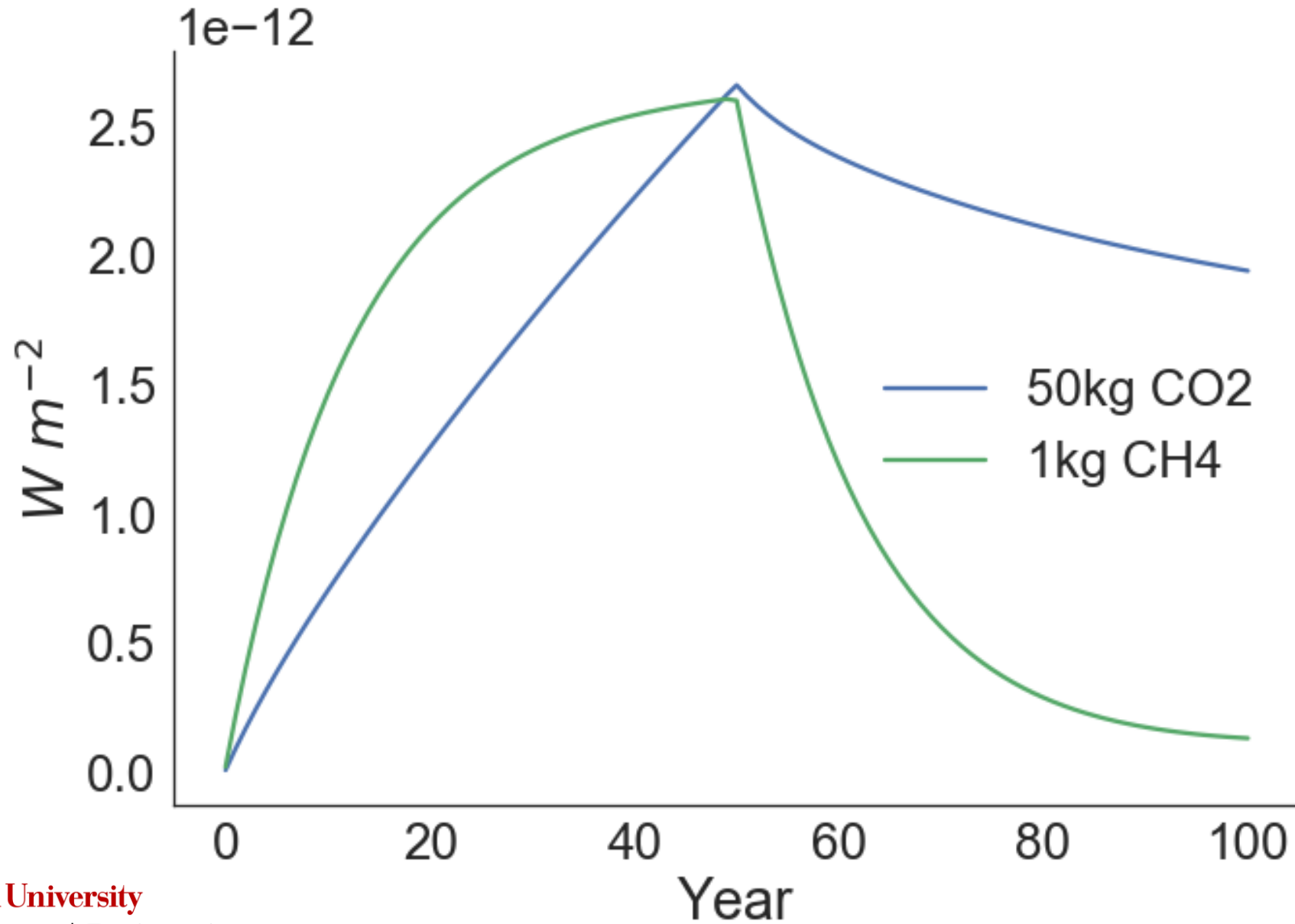
A simple example

1 kg generic emission every year for 50 years

```
end_time = 100
tstep = 0.1
time = np.linspace(start=0, stop=end_time, num=int(end_time/tstep))
emission = np.ones_like(time)
emission[500:] = 0
```

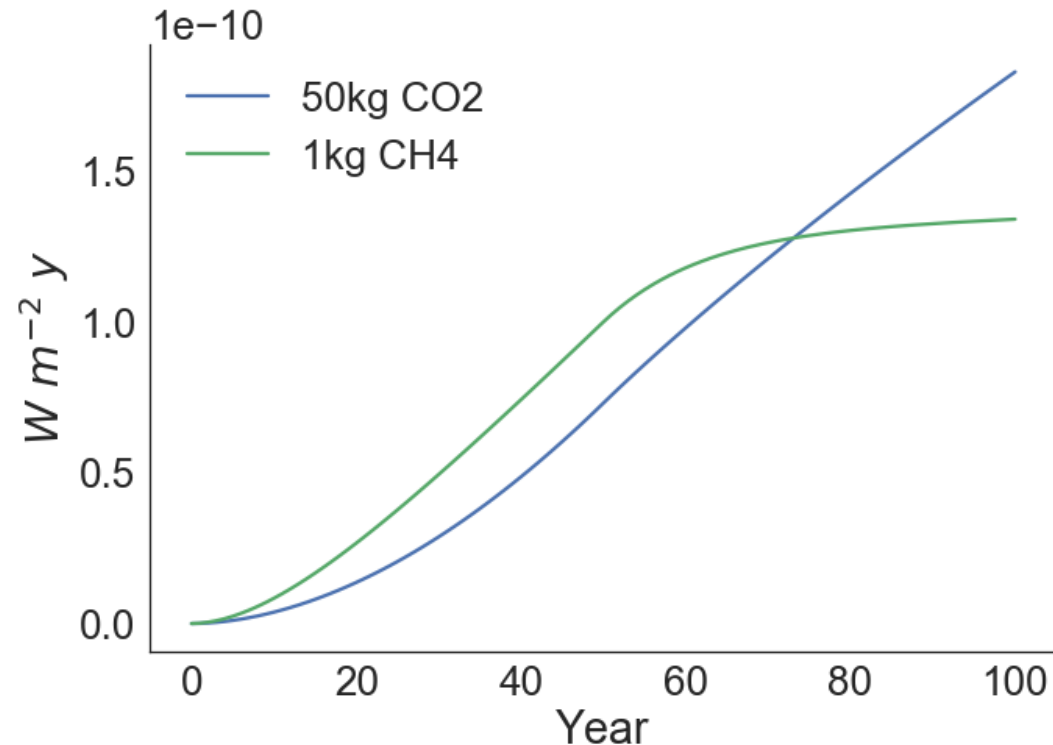
Make CO₂ 50 kg/yr

```
co2_rf = CO2(emission * 50, time, tstep=0.1, kind='RF')
ch4_rf = CH4(emission, time, tstep=0.1, kind='RF')
plt.plot(co2_rf, label='50kg CO2')
plt.plot(ch4_rf, label='1kg CH4')
```

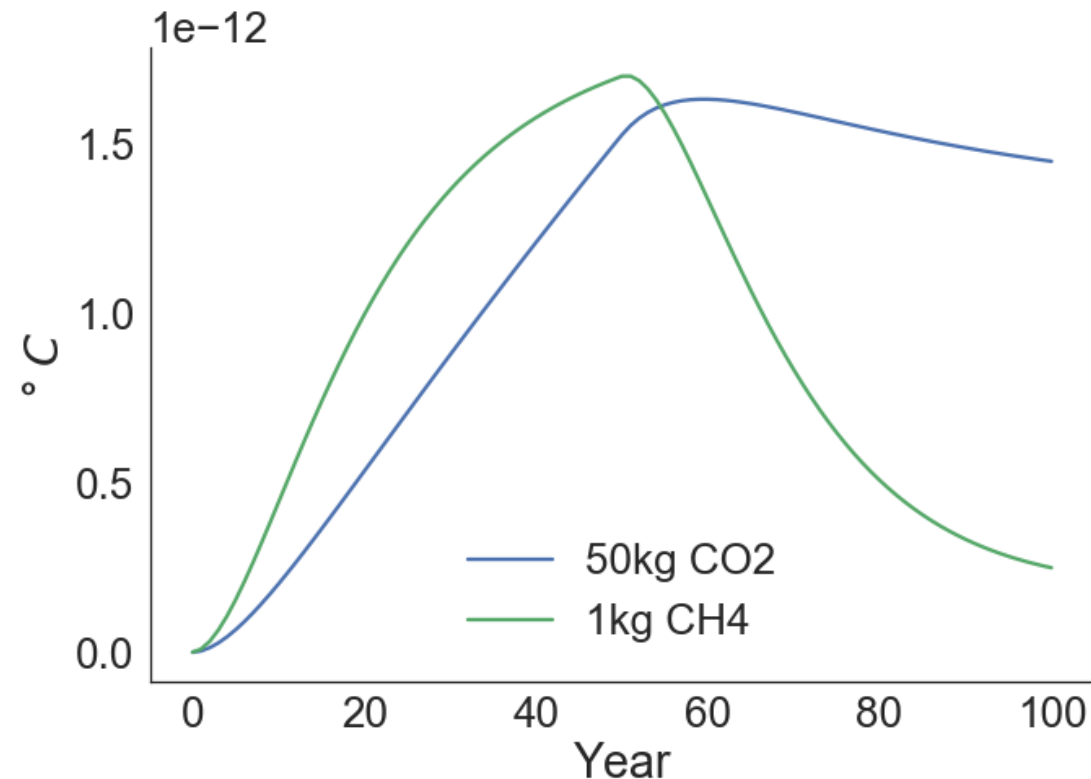
Calculate CRF

```
co2_crf = CO2(emission * 50, time, timestep=0.1, kind='CRF')
ch4_crf = CH4(emission, time, timestep=0.1, kind='CRF')
plt.plot(co2_crf, label='50kg CO2')
plt.plot(ch4_crf, label='1kg CH4')
```



Or calculate temperature

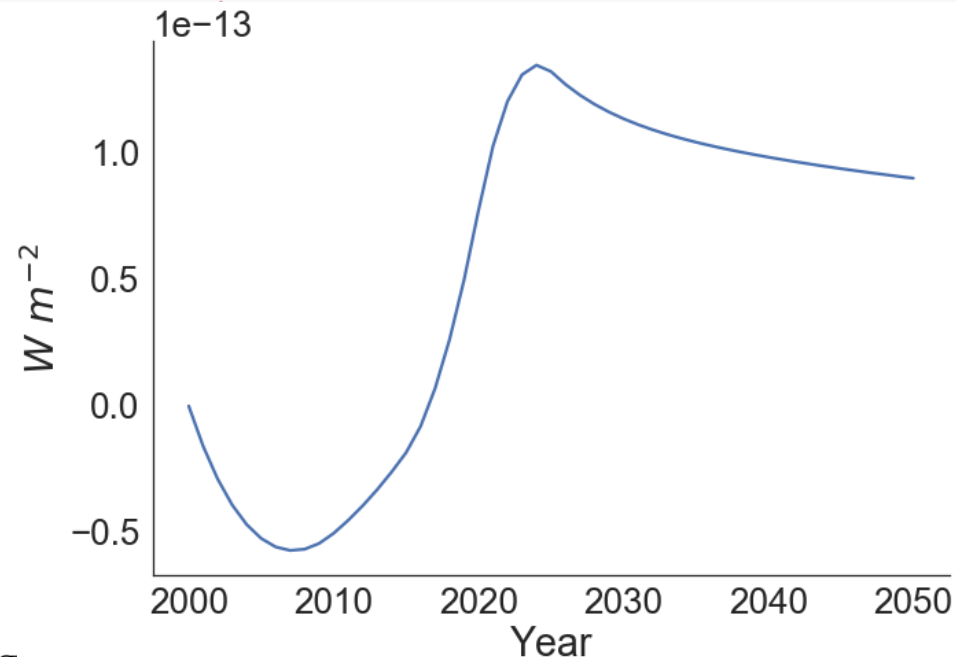
```
co2_temp = CO2(emission * 50, time, timestep=0.1, kind='temp')
ch4_temp = CH4(emission, time, timestep=0.1, kind='temp')
plt.plot(co2_temp, label='50kg CO2')
plt.plot(ch4_temp, label='1kg CH4')
```



Support for irregular timeframes

```
years = np.array([2000, 2010, 2015, 2020, 2025, 2050])  
emissions = np.array([-10, 2, 5, 20, 0, 0])
```

```
co2_rf = CO2(emissions, years,  
             interpolation='linear')  
plt.plot(range(2000, 2051), co2_rf)
```



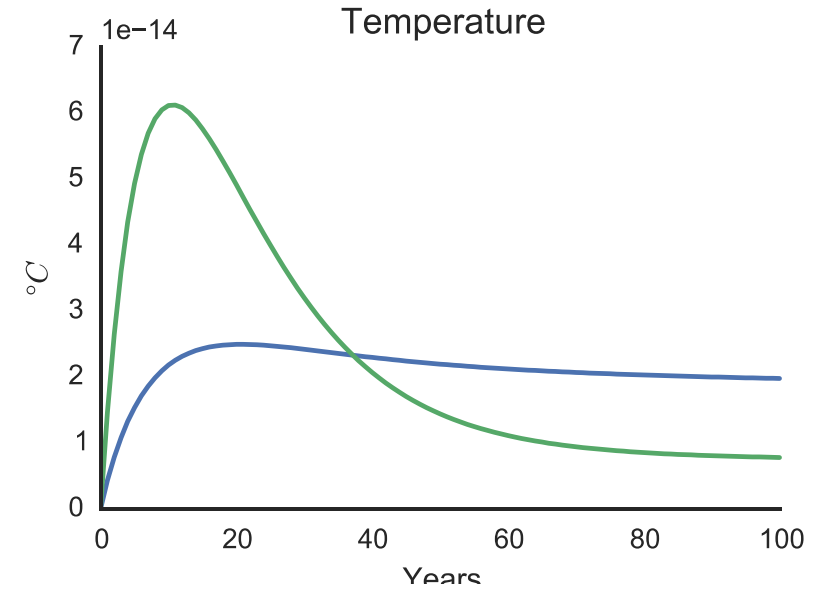
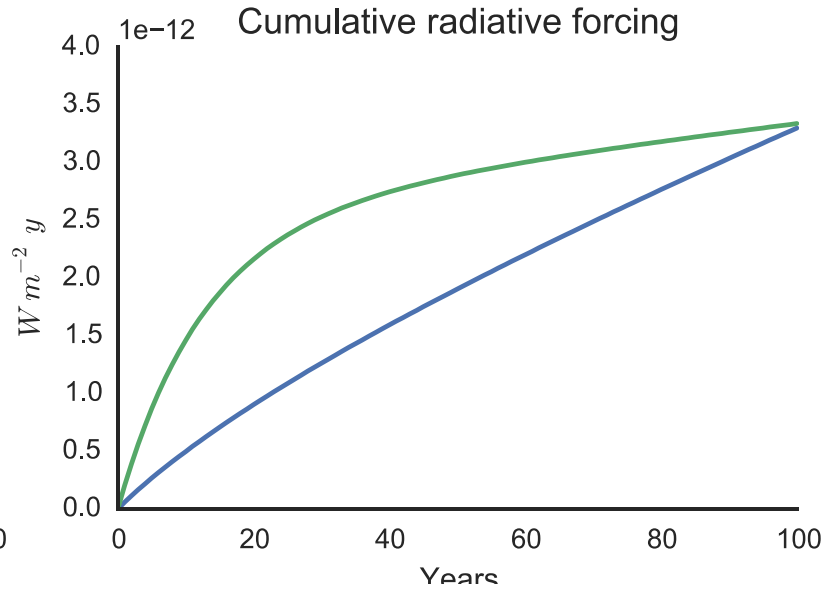
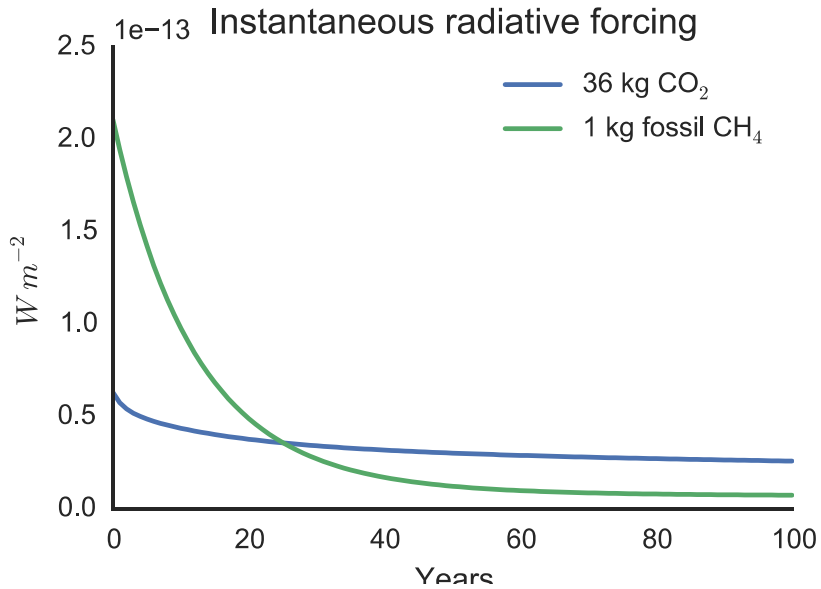
Other calculation features

- Methane with or without:
 - Climate-carbon feedbacks
 - Decomposition to CO₂

cc-feedbacks	CH ₄ → CO ₂ (fossil CH ₄)	Equivalent 100-yr GWP
×	×	28
×	✓	30
✓	×	34
✓	✓	36

- Uncertainty
 - Radiative efficiencies
 - Methane indirect effects
 - Methane lifetime
 - Fraction methane to CO₂
 - Climate-carbon feedbacks
 - CO₂ response function
- Return ± 1-sigma, or full monte carlo results

Sample uncertainty

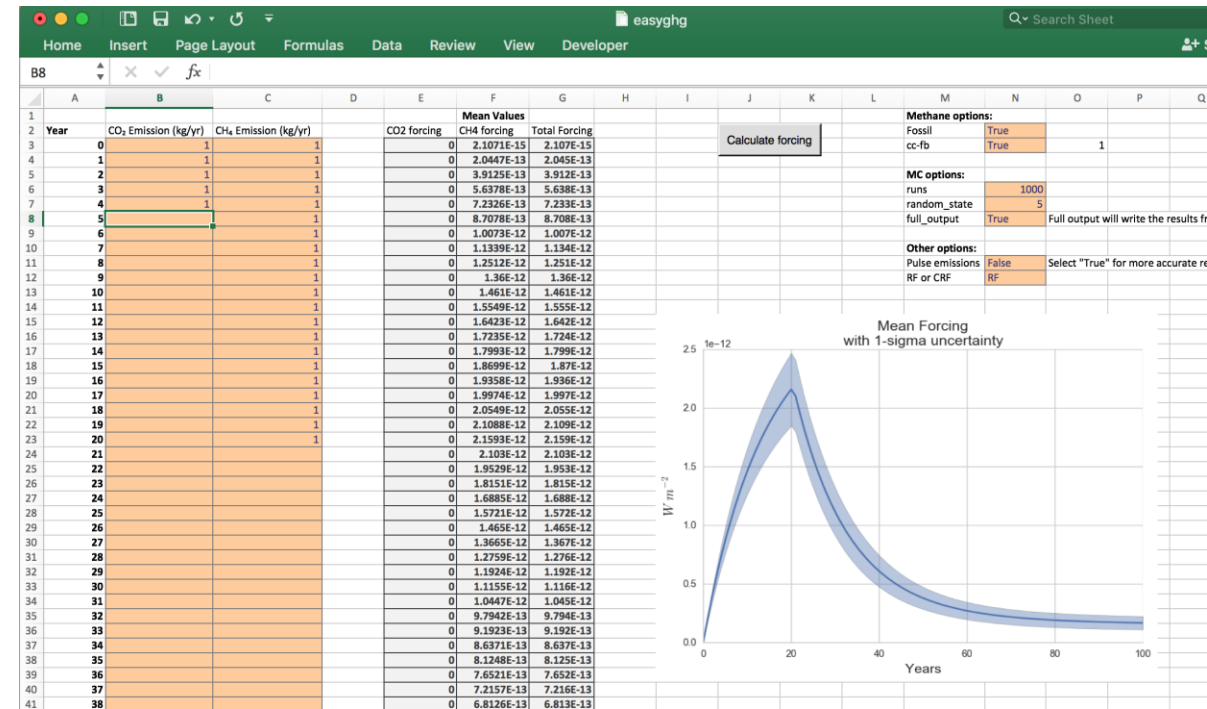


Caution with pulse emissions!

- ghgforcing was designed for continuous emissions
 - 1kg at every timestep over a year is interpreted as 1kg for the year
 - Default time-step is 0.01 years
 - A pulse in the first step will be $1/tstep$ too small!
- Change $tstep$ or multiply a pulse emission by $1/tstep$

easyghg if you don't want to use Python

- Basic excel interface
- Uses ghgforcing
- github.com/gschivley/easyghg
 - Needs to be fixed after recent updates to xlwings
 - Let me know if this is something you'd like to use



Thanks

- Contact
 - Email: gs1@cmu
 - Twitter: [@gschivley](https://twitter.com/gschivley)
- Contribute to development or log an issue
 - github.com/gschivley/ghgforcing
 - github.com/gschivley/easyghg

