Supplementary information for “Technical note: On comparing greenhouse gas emission metrics” by I. Enting and N. Clisby

The R code that forms the bulk of this document is as used for the calculations presented by Enting and Clisby in *“Technical note: On comparing greenhouse gas emission metrics”* as submitted to *Atmospheric Chemistry and Physics*. This version refers to the plots uploaded for *Atmospheric Chemistry and Physics Discussions*.

The paper [Enting and Clisby, 2020] uses a Laplace transform analysis to compare how various greenhouse gas metrics behave as a function of timescale. The R code defines a set of generic functions to support such an analysis and four functions that generate specific plots which show:

- The Laplace transforms of operators corresponding to various greenhouse gas metrics relating CH_4 to CO_2. This generates fig 1 of [Enting and Clisby, 2020]. These operators correspond to the Laplace transforms of the respective CO_2-equivalents of delta-function CH_4 emissions (when the factor of the instantaneous GWP is included).
- A methane source that increases linearly to a constant, and the CO_2-equivalent sources as defined by various metrics; This generates fig 2 of [Enting and Clisby, 2020];
- The CH_4 concentration from the specified source and the CO_2 concentration from the CO_2-equivalent sources. This generates fig 3 of [Enting and Clisby, 2020];
- A plot of the frequency response of the scaling of periodic source variations. This is not used in [Enting and Clisby, 2020] but is reproduced below.

The analysis is expressed in terms of Laplace transforms of response functions for greenhouse gases CH_4 and CO_2.

The interpretation in terms of derivatives follows from the general relation for Laplace transforms. For a function $A(t)$ with time derivative $\dot{A}(t)$, the relation between $\tilde{A}(p)$ and $\tilde{\dot{A}}(p)$, the transforms of A and $\dot{A}$, is

$$\tilde{\dot{A}}(p) = p\tilde{A} - A(t = 0) \tag{1}$$

Although the Laplace transform formalism has not often been used in studies of global change, it provides a convenient way of representing system responses. Convolutions of forcing with response functions transform into products of transforms, making it simple to exploit the commutative and associative properties of such products rather than the more complicated derivation (often involving change of variables) required for deriving the corresponding commutative and associative properties of convolution relations.

If we regard a metric transformation, denoted $\tilde{\phi}_A(p)$, as defining

$$\tilde{S}_{\mathrm{CO2:eq}}(p) = \tilde{\phi}_A(p)\tilde{S}_{\mathrm{CH4}}(p) \tag{2}$$

then we can rewrite FEI approximation as

$$a_{\mathrm{CH4}}\tilde{R}_{\mathrm{CH4}}(p)\tilde{S}_{\mathrm{CH4}} \approx a_{\mathrm{CO2}}\tilde{R}_{\mathrm{CO2}}(p)\tilde{S}_{CO2:eq} = a_{\mathrm{CO2}}[\tilde{R}_{\mathrm{CO2}}(p)\tilde{\phi}_A(p)]\tilde{S}_{\mathrm{CH4}}(p) \tag{3}$$

Thus a metric that acts on $S_{\mathrm{CH4}}(t)$ can be evaluated in terms of the approximation involving response functions:

$$a_{\mathrm{CO2}}\tilde{R}_{\mathrm{CO2}}(p)\tilde{\phi}_A(p) \approx a_{\mathrm{CH4}}\tilde{R}_{\mathrm{CH4}}(p) \tag{4}$$

```
# defines ’#" as comment character
# COMPARISON of EMISSION EQUIVALENCE METRICS
#  I. Enting and N. Clisby
# file equivcomx.r  --- run as source("equivcomx.R")
#-- last change 24/9/2020
#
# This code defines a set of functions for defining
# greenhouse gas emission equivalence metrics
# particularly for CH4 referenced to CO2.
# It has functions for comparisons as Laplace
# transforms and for comparing with specific time series.
# It has several pre-packaged cases used to generate
# plots for Technical note by Enting and Clisby 2020
# Some of the functions for Laplace transforms
# are generalised to do frequency response by
# putting p = 2*pi*frequency*sqrt(-1).
```

The comparisons noted above can be expressed in terms of frequencies, f with $p = 2\pi if$. Taking amplitudes and ignoring phase shifts, equation (4) becomes

$$a_{\mathrm{CO2}}|\tilde{R}_{\mathrm{CO2}}(p = 2\pi if)||\tilde{\phi}_A(p = 2\pi if)| \approx a_{\mathrm{CH4}}|\tilde{R}_{\mathrm{CH4}}(p = 2\pi if)| \tag{5}$$

In these terms, the metric operation is acting like an equaliser in an audio system, transforming the frequency response for CO_2 to match the frequency response for CH_4.

```
#
comment.char <- '#'
rm(list = ls()) # Clear pre-existing object definitions
#
# Set up default parameters for a plots
#
# This could be tidied up with better
# understanding of how parameters interact
maix <- 0.3 # if no right axes
# maix <- 1.8 # for postscript with right axis
 maix<- 0.8  # for screen with right axes
#
set_box0 <- function(mair = 0.3) {
      par(lwd = 2);   par(lty = 1)    #  solid line, width of 2
      par(type = "l") #  default, use lines
      par(col = "black") #  default colour black
      par(las = 1) # tick labels horizontal
      par(tck = 0.01) # ticks inside
      par(mai = c(0.8, 0.8, 0.2, mair))
          #setting margins of 1" to B & L, 0.2T, 0.3R.
      par(mgp = c(2,0.5,0)) # Spacing for axis labels
   }
#
#
xlab1 = "Years"; ylab1 = "Y_name"       # Default axis labels
#
set_box0()
#
Npts <- 500  # for range of calculated values. Plot usually less.
xlim1 = c(-10,Npts); ylim1 = c(-1,6)    # Default Plot range

twopi  <- pi*2.0
GWPfac <- 3.7/0.14 # Molar from IPCC AR4 WG1 table 2.14
GWPfac <- GWPfac*44.01/16.01  # (per kg, Used to scale second axis)
# GWPfac is the H -> 0 limit of GWP

#  Routines for response functions ................
```

In these calculations, the response used for CO_2 is the multi-model mean from [Joos et al., 2013, Table 5] and the response of CH_4 described by a 12.4 year perturbation lifetime. In each case, these represent the response to small perturbations about current conditions .

```
#     versions with z are for complex cases, but may not be
#     needed, if coercions make orginal version work.
#     Mod gives amplitude of things defined as complex
#
#  LT of single exponential response
# Should work with complex nums
ET <- function(p,amp,rate){ amp/(p+rate) }
#
#   Using CH4 perturbation lifetime
Rch4  <- function(p){ET(p,1,1/12.4)}
zRch4 <- function(p){ET(p,1+0i,1/12.4+0i)}

pRco2 <- function(p) {0.2173 + p*ET(p, 0.2240,1/394.4) +
  p*ET(p,0.2842,1/36.54)+p*ET(p,0.2763,1/4.304)
}
zpRco2 <- function(p) {0.2173+0i + p*ET(p, 0.2240+0i,1/394.4+0i) +
  p*ET(p,0.2842+0i,1/36.54+0i)+p*ET(p,0.2763+0i,1/4.304+0i)
}
```

Much of the discussion of revised metrics has concentrated on a distinction between short-lived and long-lived constituents. However CO_2 has a component that does not go to zero, except on geological time scales [e.g. Sundquist, 1985]. This long-term behaviour means that $\tilde{R}_{CO2}$ has pole at $p = 0$ and any gas with a lifetime τ will act as short-lived relative to CO_2 once the timescales of interest exceed τ. The function Rn2o is included to allow exploration of this aspect. The generalisation to other gases is obvious.

```
# Nitrous oxide response
Rn2o <- function(p){ET(p,1,1/120.0)}

FEI    <- function(p){p*Rch4(p)/pRco2(p) }
ModFEI <- function(f){z <- 0 + twopi* f*(0+1i);
        Mod(z*zRch4(z)/zpRco2(z)) }

# GWP normalised by GWP(H=0)
nGWP100 <- FEI(0.01)* 1.033 # Calclated off-line


FEIn2o    <- function(p){p*Rn2o(p)/pRco2(p) }

GWP <- function(p){nGWP100+p*0.0001} #R rejects null dependence
# This is p-dependence of GWP100 metric, not GWP(p=1/H)

Deriv    <- function(p){p* nGWP100/0.01}
ModDeriv <- function(f){Mod(Deriv(twopi*f*(0+1i)))}

# Note that Oxford group use GWP* in several ways
Gstarold    <- function(p){ 0.75* Deriv(p) + 0.25*GWP(p)}
ModGstarold <- function(f){z <- twopi*f*(0+1i)
         Mod(0.75*Deriv(z)+ 0.25*GWP(z))}
```

The Laplace transform of a time offset by τ is $\exp(-p\tau)$. The 'Gdiff' metric is from Cain et al. [2019]. There appears to be scope for combining the $\exp(-p\tau)$ behaviour of truncations with the convolution approach of the 'reduced model'. The utiiity of such metrics is likely to depend on considerations of practical utility (and thus outside the scope of the present analysis) as much as on how closely FEI is mimiced.

```
#  From Cain et al 2019.
```

```
Gdiff     <- function(p){nGWP100*(4- 3.75 * exp(-p*20)) }
 ModGdiff <- function(f){ Mod(Gdiff(twopi*f*(0+1i)))}

RM    <- function(p){p/(p+0.035) } # Reduced model
ModRM <- function(f){z <- twopi*f*(0+1i)
       Mod(z/(z+0.035+0i)) }

#......................... routines for integration etc
#
# One step of heun predictor-corrector
heunstep <- function(x,s, amp, rate, j){
  xp <- x[j-1] + amp*s[j-1]-rate*x[j-1]
  xx <- x[j-1] + 0.5*(amp*(s[j-1]+s[j])-rate*(x[j-1]+xp))
xx
}
#
heun <- function(x,s, amp, rate){ # integrate dx/dt = amp*s - rate*x
   x[1] <- 0.0
   for(j in 2:Npts) { x[j] <- heunstep(x,s,amp,rate,j)}
  x # return value (remember copy of x used within function)
}     # i.e. call by value, not call by reference
#
xlags <- function(x,xlag,rate){
  xlag[1] <- x[1]
  for(j in 2:Npts) {
   xlag[j] <- x[j] + xlag[j-1]*(1.0 - rate)
  }
xlag
}
#
```

The convolution with a response function that is expressed as a sum of exponentials can be re-expressed in terms of a sum of contributions from linear DEs.

```
intco2 <- function(s){
 # From Joos 2013, multimodel mean
 x1 <- x2 <- x3 <- x0  <- Tref
 x1 <- heun(x1,s,0.2240,1/394.4); x2 <- heun(x2,s,0.2842,1/36.54)
 x3 <- heun(x3,s,0.2763,1/4.304); x0 <- heun(x0,s,0.2173,0.0)
 xx <- x1+x2+x3+x0
}
#
# .......................................
#
Tref <- c(1:Npts)
Tref[1:Npts] <- 0
Xval <- Tref
Xval <- seq(1:Npts)
DSmeth     <- Tref # Derivative of Methane source , to be set to test function.
Smeth      <- Tref # Methane source , derived from test function.
Smethbak   <- Tref # Backdated Methane source , derived fromtest function.
#Smethbak25 <- Tref # Backdated Methane source , derived rerived test function.
Mmeth    <- Tref # Methane concentration , gives forcing
Smethlag <- Tref # For use in RM metric
SeqGWP   <- Tref # CO2-equivalent source under GWP
```

```
SeqGstar <- Tref # CO2-equivalent source inder Gstar
SeqRM    <- Tref # CO2 equivalent source under reduced model
SeqDiff  <- Tref # CO2 equivalent source under reduced model
MeqGWP   <- Tref # CO2-equivalent mass under GWP
MeqGstar <- Tref # CO2-equivalent mass inder Gstar
MeqRM    <- Tref # CO2 equivalent mass under Reduced model
MeqDeriv <- Tref # CO2-equivalent mass under Derivative
MeqDiff  <- Tref # CO2-equivalent mass under Difference
#
#  Set up Tests ................................

# Simple test source (rate of change)
DSmeth <- DSmeth*0.0
for (j in 51:65) DSmeth[j] <- 0.15

#.......................................................
Smeth <- heun(Smeth,DSmeth,1.0,0.0)
Smethbak[21:Npts] <- Smeth[1:(Npts-20)]
# Smethbak25[26:Npts] <- Smeth[1:(Npts-25)]
SeqGWP   <- nGWP100*Smeth # code prior to 14/9/20 used FEI(0.01)
SeqDeriv <-  DSmeth*nGWP100/0.01 # check amplitude
SeqGstar <- 0.75*SeqDeriv + 0.25 *SeqGWP
SeqDiff  <- nGWP100*(4.0*Smeth - 3.75* Smethbak) #  From Lynch 2020 CHECK
Smethlag <- xlags(Smeth,Smethlag, 0.035)
SeqRM    <- Smeth - 0.035*Smethlag
Fac <- 1.0
Mmeth     <- heun(Mmeth, Smeth,1.0, 1.0/12.4) * Fac
MeqGWP    <- intco2(SeqGWP) *Fac
MeqDeriv  <- intco2(SeqDeriv) *Fac
MeqGstar  <- intco2(SeqGstar) *Fac
MeqDiff   <- intco2(SeqDiff) *Fac
MeqRM     <- intco2(SeqRM) *Fac
names4  <- c("FEI", "Diff","RM", "Deriv", "GWP")
names4a <- c("CH4", "Diff","RM", "Deriv", "GWP")
cols4   <- c("black","black","black","black","black")
#cols   <- c("black","black","black","black","green","red")
types4  <- c(1,2,3,4,5)


#

SourceRun <- function(){
fname <<- "eqsrc.eps"
ylab1 <<- "Mt CO2-eq/year"
xlab1 <- "Time (years)"
CO2min <- -0.5 ; CO2max <- 4; ylim1 <- c(CO2min,CO2max)
xlim1 <- c(0,300)
yrange2 <- c(CO2min/GWPfac,CO2max/GWPfac)
mairPS <<- 1.8; mairX11 <<- 0.8 # Right margins
set_box0(mairCurr)
plot(c(0),type="n",xlab=xlab1,ylab=ylab1,xlim=xlim1,
    ylim=ylim1,xaxs = "i",yaxs = "i")
lines(Xval,Smeth,   col=cols4[1],lty=types4[1])
lines(Xval,SeqDiff, col=cols4[2],lty=types4[2])
lines(Xval,SeqRM,   col=cols4[3],lty=types4[3])
lines(Xval,SeqDeriv,col=cols4[4],lty=types4[4])
lines(Xval,SeqGWP,  col=cols4[5],lty=types4[5])
   legend(x="topright",names4a,col=cols4, lty=types4)
# Set up right axis
  par(new=TRUE)
  plot(c(0),type="n", xlim=xlim1, yaxs= "i",  ylim= yrange2,
     axes=FALSE,xlab=NA,ylab=NA)
  axis(side=4, at= pretty(yrange2))
  corners = par("usr"); par(xpd=TRUE);
  text(x=(corners[2]+72.5), y=mean(corners[3:4]),"Mt CH4/year",srt=270)
}

ForcingRun <- function(){
fname <<- "eqforce.eps"
ylab1 <<- "Mt CO2-eq"
xlab1 <- "Time (years)"
mairPS <<- 1.8; mairX11 <<- 0.8 # Right margins
set_box0(mairCurr)
CO2min <- -10 ; CO2max <- 80; ylim1 <- c(CO2min,CO2max)
xlim1 <- c(0,300)
yrange2 <- c(CO2min/GWPfac,CO2max/GWPfac)
plot(c(0),type="n",xlab=xlab1,ylab=ylab1,xlim=xlim1,ylim=ylim1,xaxs = "i",yaxs = "i")
lines(Xval,Mmeth,   col=cols4[1],lty=types4[1])
lines(Xval,MeqDiff, col=cols4[2],lty=types4[2])
lines(Xval,MeqRM,   col=cols4[3],lty=types4[3])
lines(Xval,MeqDeriv,col=cols4[4],lty=types4[4])
lines(Xval,MeqGWP,  col=cols4[5],lty=types4[5])
   legend("topleft",names4a,col=cols4, lty=types4)
par(new= TRUE)
# specifying yaxs="i" seems to be critical for not having location rounded
plot(c(0),type="n", xlim=xlim1, ylim= yrange2, yaxs="i" ,
    axes=FALSE,xlab=NA,ylab=NA)
axis(side=4) # ,at= pretty(yrange2))
corners = par("usr"); par(xpd=TRUE);
text(x=(corners[2]+72.5), y=mean(corners[3:4]),"Mt CH4",srt=270)
}

LtRun <- function(){
fname <<- "eqLT.eps"
mairPS <<- 0.3; mairX11 <<- 0.3 # Right margins
set_box0()
xlim1 <- c(0,0.05); ylim1 <- c(0,0.8)
xlab2 <- "e-folding rate, p  (1/years)";
ylab2 <- "Ratio of airborne fractions"
names4 <- c("FEI", "Diff","RM", "Deriv", "GWP")
plot(c(0),type="n",xlab=xlab2,ylab=ylab2,xlim=xlim1,ylim=ylim1,
    xaxs = "i",yaxs = "i")
 curve(FEI,lty=types4[1],add=TRUE,xlim=c(0,0.2))
 curve(Gdiff,lty=types4[2],add=TRUE ,  xlim=c(0,0.2))
 curve(RM,lty=types4[3],add=TRUE)
 curve(Deriv,lty=types4[4],add=TRUE)
 curve(GWP,lty=types4[5],add=TRUE)
   legend("topleft",names4,col=cols4, lty=types4)
```

```
}


FreqRun <- function(){
fname <<- "eqFreq.eps"
set_box0(mairCurr)
xlim1  <- c(0,0.06); ylim1 <- c(0,3.0)
mairPS <<- 0.3; mairX11 <<- 0.3 # Right margins
xlab2  <- "Cycles per year"; ylab2 <- "Frequency response"
names3 <- c("FEI", "GWP", "Deriv", "Gstar", "Gdiff","RM")
names4 <- c("FEI", "Gdiff","RM", "Deriv", "GWP")
plot(c(0),type="n",xlab=xlab2,ylab=ylab2,xlim=xlim1,
      ylim=ylim1,xaxs = "i",yaxs = "i")
 curve(ModFEI,lty=types4[1],add=TRUE) #   ,xlim=c(0,0.2))
 curve(ModGdiff,lty=types4[2] , add=TRUE ) # ,  xlim=c(0,0.2))
 curve(ModRM,lty=types4[3],add=TRUE)
 curve(ModDeriv,lty=types4[4],add=TRUE)
 curve(GWP,lty=types4[5],add=TRUE)
   legend("topright",names4,col=cols4, lty=types4)
}
#
# Swap these around as required
# CurrentRun <- LtRun
# CurrentRun <- FreqRun
# CurrentRun <- SourceRun
 CurrentRun <- ForcingRun

X11()
  mairCurr <- 0.3
CurrentRun() # Role is to set fname
  postscript(fname, width = 6.5, height = 5.0,
            horizontal = FALSE, pointsize = 15)
# Note: horizontal = FALSE for consistent orientation on portrait docs
mairCurr <-  mairPS
CurrentRun() # Generate output file
  dev.off() # Close output file
X11()
mairCurr <- mairX11
CurrentRun() # Now have a look at it
#
# On exit to console mode, everything: functions, constants and series
#   remains accessible if it was defined at top level.
# ..................................................................
```

References

M. Cain, J. Lynch, M. R. Allen, J. S. Fugelstvedt, D. J. Frame, and A. M. Macey. Improved calculation of warming-equivalent emissions for short-lived climate pollutants. *npj: Climate and Atmospheric Science*, 2:29, 2019. doi: 10.1038/s41612-019-0084-4.

I. G. Enting and N. Clisby. Technical note on comparing greenhouse gas emission metrics. *Submitted to Atmospheric Chemistry and Physics Discussions*, pages acp–2020–996, 2020.

F. Joos, R. Roth, J. S. Fuglestvedt, G. P. Peters, I. G. Enting, W. von Bloh, V. Brovkin, E. J. Burke, M. Eby, N. R.

Edwards, T. Friedrich, T. L. Frölicher, P. R. Halloran, P. B. Holden, C. Jones, T. Kleinen, F. Mackenzie, K. Matsumoto, M. Meinshausen, G.-K. Plattner, A. Reisinger, J. Segschneider, G. Shaffer, M. Steinacher, K. Strassmann, K. Tanaka, A. Timmermann, and A. J. Weaver. Carbon dioxide and climate impulse response functions for the computation of greenhouse gas metrics: a multi-model analysis. *Atmos. Chem. Phys.*, 13:2793–2825, 2013.

E. T. Sundquist. Geological perspectives on carbon dioxide and the carbon cycle. In E. T. Sundquist and W. S. Broecker, editors, *The Carbon Cycle and Atmospheric CO_2: Natural Variations Archean to Present. Geophysical Monograph 32.*, pages 5–59. AGU, Washington, 1985.

.

I. Enting and N. Clisby. Typeset October 18, 2020.

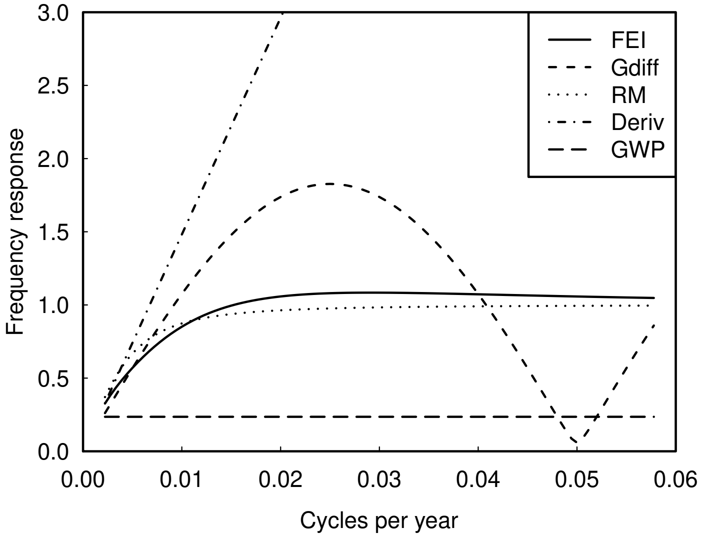

Figure 1: Frequency response for various metrics analysed by Enting and Clisby [2020].