Tutorial for the reduced scale Developed-Developing Nations model used at 4CMR

Introduction

The model used here was developed initially in the software STELLA, so the equation format below reflects the ways in which STELLA displays equations. However, the same equations can be placed into any coding language. For example, at 4CMR we generally use a MATLAB version.

In the Results Viewers of the 4CMR website, the various scenarios are functions of nine aspects of the model that are shown in Bold, Italics below. The first is a rate constant called kbr, which is the fractional annual rate at which the birth and mortality rates are made equal and hence the population is brought under control. In the version of the model below, there is only one value of kbr that controls population growth in both the Developed (D) and Developing DG) nations, but you can easily adjust the model so these two sets of nations are controlled separately (by replacing the single kbr with two constants kbrD and kbrDG).

Four of these aspects are years in which policies are introduced. These policies control the fractional rate at which the carbon intensity of the energy system is reduced in the Developed (D) and Developing (DG) nations, and the fractional rate at which the growth in per capita energy use is slowed in the Developed (D) and Developing (DG) nations. There are then four terms that specify these fractional rates of control, one for each of the four policies mentioned.

If you are using the on-line version of the model at the 4CMR website, you will find slider controls for each of these nine features. Rather than specify the year of introduction of a policy as the number of years since start of the simulation (which is 1990), the website sliders show this as the actual year of introduction of the policy (eg, the year 2021). In addition, you will find a slider for ARr where you can adjust the amount of rainforest in the world. When this is adjusted, the equivalent change is made automatically in cropland as just one example of land use changes.

In the on-line version, graphs are provided for the amount of C in the Atmosphere and Mixing Ocean; Annual Global Carbon Emissions (Developed and Developing nations combined); per capita Carbon Emissions (separately for Developed and Developing nations); and Population of the Developed and Developing nations.

When running simulations, you might consider several possible policy aims:

- Preventing the Atmosphere from rising above 1160 billion tonnes C (550 ppm)
- Preventing the Atmosphere from rising above 950 billion tonnes (450 ppm)
- Achieving parity in per capita C emissions in Developed and Developing nations

NOTE: The model is founded on a set of coupled zeroth and first order differential equations describing the state of the 5 environmental compartments (Atmosphere, Mixing Ocean, Soil, Deep Earth and Flora) and 2 populations (Developed and Developing nations), each with initial conditions. The solutions are generated by numerical methods using Newton's method with suitably small time steps, but your code could use any of the numerical methods available. The model description below gives first the differential equations, then their discrete time numerical approximations, then the equations or parameter values providing information required by the differential equations. Since there are multiple coupled differential equations for the environmental compartments, there is no closed form, analytic solution available (at least as far as we have determined), which is why numerical solutions are generated for the amount of carbon in each compartment. Population growth could be specified as a closed form solution (it is a simple exponential function), but is represented in our version of the model as a numerical solution at discrete time points for consistency across the 7 differential equations.

The differential equations

In these equations, $N_x(t)$ is the amount of carbon in compartment x; POPx(t) is number of people in population x at time t; lxy is the first order rate constant for flow from compartment x to y.

$dN_{Atmosphere}(t)/dt$ = (Roa + Rsa + Rfa + Rde + RFF - Raf - Rao - Ras) = loa*$N_{Mixing\_Ocean}(t)$ + lsa*$N_{Soil}(t)$ + lfa*$N_{Flora}(t)$ + Rde + RFF - (ARb*NPPb + ARc*NPPc + ARd*NPPd + ARg*NPPg + ARm*NPPm + ARr*NPPr) – lao*$N_{Atmosphere}(t)$ – las*$N_{Atmosphere}(t)$ *(this third term is set equal to zero because the terms in brackets represent NET flow from Atmosphere to Flora; Rde is a specified constant; RFF is the sum of the products of per capita carbon emissions times POP(t) for the Developed and Developing nations)*

$dN_{deep\_earth}(t)/dt$ = (Rod + Rsd) = lod*$N_{Mixing\_Ocean}(t)$ + lsd*$N_{Soil}(t)$ *(this equation is not important in the model because Deep Earth is a permanent sink on the time scale of these simulations and Rde in the previous differential equation is taken as a constant)*

$dN_{Flora}(t)/dt$ = (Raf - Rfs - Rfa) = (ARb*NPPb + ARc*NPPc + ARd*NPPd + ARg*NPPg + ARm*NPPm + ARr*NPPr) – lfs*$N_{Flora}(t)$ – lfa*$N_{Flora}(t)$ *(this last term is set equal to zero because the terms in brackets represent NET flow from Atmosphere to Flora)*

$dN_{Mixing\_Ocean}(t)/dt$ = (Rao - Roa - Rod) = lao*$N_{Atmosphere}(t)$ – loa*$N_{Mixing\_Ocean}(t)$ – lod*$N_{Mixing\_Ocean}(t)$

$dN_{Soil}(t)/dt$ = (Rfs + Ras - Rsa - Rsd) = lfs*$N_{Flora}(t)$ + las*$N_{Atmosphere}(t)$ – lsa*$N_{Soil}(t)$ – lsd*$N_{Soil}(t)$

dPOPD(t)/dt = (popgrowD(t)) = (BrD*SFD – MrD)*POPD(t)

dPOPDG(t)/dt = (popgrowDG(t)) =(BrDG*SFDG – MrDG)*POPDG(t)

The discrete time step approximations to the model equations

Atmosphere(t) = Atmosphere(t - dt) + (Roa + Rsa + Rfa + Rde + RFF - Raf - Rao - Ras) * dt
(this is the numerical solution to the differential equation controlling the amount of carbon in the Atmosphere, with this amount in units of billion tonnes C – not tonnes CO2)

INIT Atmosphere = 740
(this is the amount of carbon in the Atmosphere at the start of the time period, in units of billion tonnes C – not tonnes CO2)

deep_earth(t) = deep_earth(t - dt) + (Rod + Rsd) * dt
(this is the numerical solution to the differential equation controlling the amount of carbon in the Deep Earth caused by returns from soil and oceans during the period of the simulation only, with this amount in units of billion tonnes C – not tonnes CO2; it is not necessary for the model and so can be removed if desired – we use it simply to keep track of total C in the system)

INIT deep_earth = 0
(this is the amount of carbon in the Deep Earth at the start of the time period, in units of billion tonnes C – not tonnes CO2; see the note above as to why it is 0 here; again, this term is not necessary for the model – we use it simply to keep track of total C in the system)

Flora(t) = Flora(t - dt) + (Raf - Rfs - Rfa) * dt
(this is the numerical solution to the differential equation controlling the amount of carbon in the Flora or vegetation in units of billion tonnes C – not tonnes CO2)

INIT Flora = 560
(this is the amount of carbon in the Flora or vegetation at the start of the time period, in units of billion tonnes C – not tonnes CO2)

Mixing_Ocean(t) = Mixing_Ocean(t - dt) + (Rao - Roa - Rod) * dt
(this is the numerical solution to the differential equation controlling the amount of carbon in the Mixing Ocean in units of billion tonnes C – not tonnes CO2)

INIT Mixing_Ocean = 2500
(this is the amount of carbon in the Mixing Ocean at the start of the time period, in units of billion tonnes C – not tonnes CO2)

Soil(t) = Soil(t - dt) + (Rfs + Ras - Rsa - Rsd) * dt
(this is the numerical solution to the differential equation controlling the amount of carbon in the Soil in units of billion tonnes C – not tonnes CO2)

INIT Soil = 1720
(this is the amount of carbon in the Soil at the start of the time period, in units of billion tonnes C – not tonnes CO2)

Roa = Mixing_Ocean*loa
(this is the rate of flow, in units of billion tonnes C per year, from the Mixing Ocean to the Atmosphere)

Rsa = Soil*lsa
(this is the rate of flow, in units of billion tonnes C per year, from the Soil to the Atmosphere)

Rfa = 0
(this is the rate of flow, in units of billion tonnes C per year, from the Flora to the Atmosphere; however Raf as shown below is the NET rate of flow, so Rfa is set to 0)

Rde = Rdeat
(this is the rate of flow, in units of billion tonnes C per year, from the Deep Earth to the Atmosphere, equal to another constant defined below as Rdeat – a peculiarity of the way in which the model was developed originally in STELLA)

RFF = (POPD*PCPD)+(POPDG*PCPDG)
(this is the rate of flow, in units of billion tonnes C per year, from Society to the Atmosphere, with components from the developed nations D and developing nations DG; the terms POP and PCP are defined later)

Raf = land
(this is the rate of flow, in units of billion tonnes C per year, from the Atmosphere to the Flora; the value of land is defined later)

Rao = Atmosphere*Iao
(this is the rate of flow, in units of billion tonnes C per year, from the Atmosphere to the Mixing Ocean)

Ras = (Atmosphere*Ias)
(this is the rate of flow, in units of billion tonnes C per year, from the Atmosphere to the Soil)

Rod = Mixing_Ocean*Iod
(this is the rate of flow, in units of billion tonnes C per year, from the Mixing Ocean to the Deep Earth by settling followed by subduction)

Rsd = Soil*Isd
(this is the rate of flow, in units of billion tonnes C per year, from the Soil to the Deep Earth)

Rfs = Flora*Ifs
(this is the rate of flow, in units of billion tonnes C per year, from the Flora to the Soil)

land = ARb*NPPb+ARc*NPPc+ARd*NPPd+ARg*NPPg+ARm*NPPm+ARr*NPPr
(this is the NET rate of flow from Flora or vegetation into the Atmosphere, in units of billion tonnes C per year – not tonnes CO2)

Iao = 0.125
(this is the first order rate constant for flow of C from Atmosphere to Mixing Ocean; it is the instantaneous fraction of the Atmosphere content that flows to Mixing Ocean per year)

las = 0
(this is the first order rate constant for flow of C from Atmosphere to Soil; it is poorly established at present so is not reflected here, although this can be changed as better data become available)

lfs = 0.0982
(this is the first order rate constant for flow of C from Flora to Soil; it is the instantaneous fraction of the Flora content that flows to Soil per year)

loa = 0.036
(this is the first order rate constant for flow of C from Mixing Ocean to Atmosphere; it is the instantaneous fraction of the Mixing Ocean content that flows to Atmosphere per year)

lod = 1.2E-03
(this is the first order rate constant for flow of C from Mixing Ocean to Deep Earth; it is the instantaneous fraction of the Mixing Ocean content that flows to Deep Earth per year)

lsa = 0.03139
(this is the first order rate constant for flow of C from Soil to Atmosphere; it is the instantaneous fraction of the Soil content that flows to Atmosphere per year)

lsd = 5.81E-04
(this is the first order rate constant for flow of C from Soil to Deep Earth; it is the instantaneous fraction of the Soil content that flows to Deep Earth per year)

Rdeat = 1.5
(this is the zeroth order rate constant for flow of C from Deep Earth to; in units of tonnes C per year – not CO_2)

POPD(t) = POPD(t - dt) + (popgrowD) * dt
(this is the numerical solution to the differential equation controlling the population size in the Developed nations)

INIT POPD = 1.13e9
(this is the population size in the Developed nations at the start of the time period)

popgrowD = (BrD*SFD-MrD)*POPD
(this is the rate of growth of the population in Developed nations at any time during the simulation; the terms in parentheses are defined later)

POPDG(t) = POPDG(t - dt) + (popgrowDG) * dt
(this is the numerical solution to the differential equation controlling the population size in the Developing nations)

INIT POPDG = 4.46e9
(this is the population size in the Developing nations at the start of the time period)

popgrowDG = (BrDG*SFDG-MrDG)*POPDG
(this is the rate of growth of the population in Developing nations at any time during the simulation; the terms in parentheses are defined later)

BrD = 0.013-(0.013-0.010070493)*(1-exp(-kbr*TIME))
(this is the first order birth rate constant for the Developed nations at the start of the simulation; TIME is the year of calculation; if population control is applied, this rate constant is not truly a first order rate constant, but rather changes – declines – in time)

BrDG = 0.038-(0.038-0.013186813)*(1-exp(-kbr*time))
(this is the first order birth rate constant for the Developing nations at the start of the simulation; TIME is the year of calculation; if population control is applied, this rate constant is not truly a first order rate constant, but rather changes – declines – in time)

kbr = 0.03
(this is a rate constant that represents the fractional rate of decline in the birth rate)

MrD = 0.01
(this is the mortality rate – fraction of people who die annually – in the Developed nations; here, it is made a constant but this can be adjusted to be a function of time if desired)

MrDG = 0.012
(this is the mortality rate – fraction of people who die annually – in the Developing nations; here, it is made a constant but this can be adjusted to be a function of time if desired)

SFD = 0.993
(this is the neonatal survival fraction – fraction of newborns who survive the neonatal period – in the Developed nations; here, it is made a constant but this can be adjusted to be a function of time if desired; if the birth rate and mortality rate is taken from a database that already reflects neonatal survival, this value should be set to 1)

SFDG = 0.91
(this is the neonatal survival fraction – fraction of newborns who survive the neonatal period – in the Developing nations; here, it is made a constant but this can be adjusted to be a function of time if desired; if the birth rate and mortality rate is taken from a database that already reflects neonatal survival, this value should be set to 1)

EEND = 39e-6*exp(EEN_growth_D*TIME)

(this is the existential energy need in the Developed nations, defined as millions of barrels of oil equivalent per person per year obtained at the point of use as available end-use energy)

EENDG = 13e-7*exp(EEN_growth_DG*TIME)
(this is the existential energy need in the Developing nations, defined as millions of barrels of oil equivalent per person per year obtained at the point of use as available end-use energy)

EEN_growth_D = IF(TIME<Year_of_Reduction_Policy_D)THEN(0.02)ELSE(0.02*exp(-Reduction_rate_D_growth*(TIME-Year_of_Reduction_Policy_D)))
(this is a rate constant equal to the fractional rate of growth per year in EEND for the Developed nations; it is controlled by the policy defined below)

EEN_growth_DG = IF(TIME<Year_of_Reduction_Policy_DG)THEN(0.04)ELSE(0.04*exp(-Reduction_rate_DG_growth*(TIME-Year_of_Reduction_Policy_DG)))
(this is a rate constant equal to the fractional rate of growth per year in EEND for the Developing nations; it is controlled by the policy defined below)

EFFD = 0.6
(this is the average efficiency of energy generation and use in Developed nations; in this version of the model it does not change in time, but can be made to do so if you wish)

EFFDG = 0.4
(this is the average efficiency of energy generation and use in Developing nations; in this version of the model it does not change in time, but can be made to do so if you wish)

PCPD = EEND*RFD/EFFD
(this is the per capita production of C in units of tonnes C per person per year in the Developed nations)

PCPDG = EENDG*RFDG/EFFDG
(this is the per capita production of C in units of tonnes C per person per year in the Developing nations)

Reduction_rate_D = 0.03
(this is the fractional rate of reduction per year in the carbon intensity of energy production, or RFD, in Developed nations due to the policy mentioned below)

Reduction_rate_D_growth = 0.04
(this is the fraction rate of reduction per year in the rate of growth of EEND, in Developed nations due to the policy mentioned below)

Reduction_rate_DG = 0.025
(this is the fractional rate of reduction per year in the carbon intensity of energy production, or RFDG, in Developing nations due to the policy mentioned below)

Reduction_rate_DG_growth = 0.02
(this is the fraction rate of reduction per year in the rate of growth of EENDG, in Developing nations due to the policy mentioned below)

RFD = IF(TIME<Year_of_Policy_D)THEN(5e-5)ELSE(5e-5*EXP(-Reduction_rate_D*(TIME-Year_of_Policy_D)))
(this is the value of the Release Factor or carbon intensity of energy production in Developed nations as a function of time; it has units of billion tonnes of C per million barrels of oil equivalent)

RFDG = IF(TIME<Year_of_Policy_DG)THEN(1E-4)ELSE(1e-4*exp(-Reduction_rate_DG*(TIME-Year_of_Policy_DG)))
(this is the value of the Release Factor or carbon intensity of energy production in Developing nations as a function of time; it has units of billion tonnes of C per million barrels of oil equivalent)

Year_of_Policy_D = 25
(this is the numbers of years since year 0 at the start of the simulation – which is 1990 in this model – in which a policy is introduced for Reduction_rate_D)

Year_of_Policy_DG = 30
(this is the number of years since year 0 at the start of the simulation – which is 1990 in this model – in which a policy is introduced for Reduction_rate_DG)

Year_of_Reduction_Policy_D = 30
(this is the number of years since year 0 at the start of the simulation – which is 1990 in this model – in which a policy is introduced for Reduction_rate_D_growth)

Year_of_Reduction_Policy_DG = 60
(this is the number of years since year 0 at the start of the simulation – which is 1990 in this model – in which a policy is introduced for Reduction_rate_DG_growth)

ARb = 50
(this is the global land area devoted to barren in units of trillions of square meters)

ARc = 14
(this is the global land area devoted to cropland in units of trillions of square meters)

ARd = 31.5
(this is the global land area devoted to deciduous forest in units of trillions of square meters)

ARg = 32
(this is the global land area devoted to grassland in units of trillions of square meters)

ARm = 4.5
(this is the global land area devoted to marshland in units of trillions of square meters)

ARr = 17
(this is the global land area devoted to rainforest in units of trillions of square meters)

NPPb = .0018
(this is the net primary productivity of barren in units of billions of tonnes C per year per 10^{12} square meters)

NPPc = 0.33
(this is the net primary productivity of cropland in units of billions of tonnes C per year per 10^{12} square meters)

NPPd = 0.6
(this is the net primary productivity of deciduous forest in units of billions of tonnes C per year per 10^{12} square meters)

NPPg = 0.25
(this is the net primary productivity of grassland in units of billions of tonnes C per year per 10^{12} square meters)

NPPm = 1.24
(this is the net primary productivity of marshland in units of billions of tonnes C per year per 10^{12} square meters)

NPPr = 1
(this is the net primary productivity of rainforest in units of billions of tonnes C per year per 10^{12} square meters)

goalC = 1160
(this is the global amount of C in the Atmosphere, in billion tonnes C – not CO2, corresponding to a doubling of the pre-industrial revolution level; it corresponds to approximately 550 ppm; adjust if other targets are selected - for example 450 ppm corresponds to approximately 950 billion tonnes C)