Online appendix files:

The standard DICE-2016R model is listed below in section A in blue color. The model for uncertainty analysis is listed in section B in grey color. These require the GAMS code to operate.

1. Standard GAMS code for DICE-2016R model

$ontext

This is the beta version of DICE-2016R. The major changes are outlined in Nordhaus,

"Revisiting the social cost of carbon: Estimates from the DICE-2016R model,"

September 30, 2016," available from the author.

Version is DICE-2016R-091916ap.gms

$offtext

$title DICE-2016R September 2016 (DICE-2016R-091216a.gms)

set t Time periods (5 years per period) /1\*100/

PARAMETERS

\*\* Availability of fossil fuels

fosslim Maximum cumulative extraction fossil fuels (GtC) /6000/

\*\*Time Step

tstep Years per Period /5/

\*\* If optimal control

ifopt Indicator where optimized is 1 and base is 0 /1/

\*\* Preferences

elasmu Elasticity of marginal utility of consumption /1.45 /

prstp Initial rate of social time preference per year /.015 /

\*\* Population and technology

gama Capital elasticity in production function /.300 /

pop0 Initial world population 2015 (millions) /7403 /

popadj Growth rate to calibrate to 2050 pop projection /0.134 /

popasym Asymptotic population (millions) /11500 /

dk Depreciation rate on capital (per year) /.100 /

q0 Initial world gross output 2015 (trill 2010 USD) /105.5 /

k0 Initial capital value 2015 (trill 2010 USD) /223 /

a0 Initial level of total factor productivity /5.115 /

ga0 Initial growth rate for TFP per 5 years /0.076 /

dela Decline rate of TFP per 5 years /0.005 /

\*\* Emissions parameters

gsigma1 Initial growth of sigma (per year) /-0.0152 /

dsig Decline rate of decarbonization (per period) /-0.001 /

eland0 Carbon emissions from land 2015 (GtCO2 per year) / 2.6 /

deland Decline rate of land emissions (per period) / .115 /

e0 Industrial emissions 2015 (GtCO2 per year) /35.85 /

miu0 Initial emissions control rate for base case 2015 /.03 /

\*\* Carbon cycle

\* Initial Conditions

mat0 Initial Concentration in atmosphere 2015 (GtC) /851 /

mu0 Initial Concentration in upper strata 2015 (GtC) /460 /

ml0 Initial Concentration in lower strata 2015 (GtC) /1740 /

mateq Equilibrium concentration atmosphere (GtC) /588 /

mueq Equilibrium concentration in upper strata (GtC) /360 /

mleq Equilibrium concentration in lower strata (GtC) /1720 /

\* Flow paramaters

b12 Carbon cycle transition matrix /.12 /

b23 Carbon cycle transition matrix /0.007 /

\* These are for declaration and are defined later

b11 Carbon cycle transition matrix

b21 Carbon cycle transition matrix

b22 Carbon cycle transition matrix

b32 Carbon cycle transition matrix

b33 Carbon cycle transition matrix

sig0 Carbon intensity 2010 (kgCO2 per output 2005 USD 2010)

\*\* Climate model parameters

t2xco2 Equilibrium temp impact (oC per doubling CO2) / 3.1 /

fex0 2015 forcings of non-CO2 GHG (Wm-2) / 0.5 /

fex1 2100 forcings of non-CO2 GHG (Wm-2) / 1.0 /

tocean0 Initial lower stratum temp change (C from 1900) /.0068 /

tatm0 Initial atmospheric temp change (C from 1900) /0.85 /

c1 Climate equation coefficient for upper level /0.1005 /

c3 Transfer coefficient upper to lower stratum /0.088 /

c4 Transfer coefficient for lower level /0.025 /

fco22x Forcings of equilibrium CO2 doubling (Wm-2) /3.6813 /

\*\* Climate damage parameters

a10 Initial damage intercept /0 /

a20 Initial damage quadratic term

a1 Damage intercept /0 /

a2 Damage quadratic term /0.00236 /

a3 Damage exponent /2.00 /

\*\* Abatement cost

expcost2 Exponent of control cost function / 2.6 /

pback Cost of backstop 2010$ per tCO2 2015 / 550 /

gback Initial cost decline backstop cost per period / .025 /

limmiu Upper limit on control rate after 2150 / 1.2 /

tnopol Period before which no emissions controls base / 45 /

cprice0 Initial base carbon price (2010$ per tCO2) / 2 /

gcprice Growth rate of base carbon price per year /.02 /

\*\* Scaling and inessential parameters

\* Note that these are unnecessary for the calculations

\* They ensure that MU of first period's consumption =1 and PV cons = PV utilty

scale1 Multiplicative scaling coefficient /0.0302455265681763 /

scale2 Additive scaling coefficient /-10993.704/ ;

\* Program control variables

sets tfirst(t), tlast(t), tearly(t), tlate(t);

PARAMETERS

l(t) Level of population and labor

al(t) Level of total factor productivity

sigma(t) CO2-equivalent-emissions output ratio

rr(t) Average utility social discount rate

ga(t) Growth rate of productivity from

forcoth(t) Exogenous forcing for other greenhouse gases

gl(t) Growth rate of labor

gcost1 Growth of cost factor

gsig(t) Change in sigma (cumulative improvement of energy efficiency)

etree(t) Emissions from deforestation

cumetree(t) Cumulative from land

cost1(t) Adjusted cost for backstop

lam Climate model parameter

gfacpop(t) Growth factor population

pbacktime(t) Backstop price

optlrsav Optimal long-run savings rate used for transversality

scc(t) Social cost of carbon

cpricebase(t) Carbon price in base case

photel(t) Carbon Price under no damages (Hotelling rent condition)

ppm(t) Atmospheric concentrations parts per million

atfrac(t) Atmospheric share since 1850

atfrac2010(t) Atmospheric share since 2010 ;

\* Program control definitions

tfirst(t) = yes$(t.val eq 1);

tlast(t) = yes$(t.val eq card(t));

\* Parameters for long-run consistency of carbon cycle

b11 = 1 - b12;

b21 = b12\*MATEQ/MUEQ;

b22 = 1 - b21 - b23;

b32 = b23\*mueq/mleq;

b33 = 1 - b32 ;

\* Further definitions of parameters

a20 = a2;

sig0 = e0/(q0\*(1-miu0));

lam = fco22x/ t2xco2;

l("1") = pop0;

loop(t, l(t+1)=l(t););

loop(t, l(t+1)=l(t)\*(popasym/L(t))\*\*popadj ;);

ga(t)=ga0\*exp(-dela\*5\*((t.val-1)));

al("1") = a0; loop(t, al(t+1)=al(t)/((1-ga(t))););

gsig("1")=gsigma1; loop(t,gsig(t+1)=gsig(t)\*((1+dsig)\*\*tstep) ;);

sigma("1")=sig0; loop(t,sigma(t+1)=(sigma(t)\*exp(gsig(t)\*tstep)););

pbacktime(t)=pback\*(1-gback)\*\*(t.val-1);

cost1(t) = pbacktime(t)\*sigma(t)/expcost2/1000;

etree(t) = eland0\*(1-deland)\*\*(t.val-1);

cumetree("1")= 100; loop(t,cumetree(t+1)=cumetree(t)+etree(t)\*(5/3.666););

rr(t) = 1/((1+prstp)\*\*(tstep\*(t.val-1)));

forcoth(t) = fex0+ (1/17)\*(fex1-fex0)\*(t.val-1)$(t.val lt 18)+ (fex1-fex0)$(t.val ge 18);

optlrsav = (dk + .004)/(dk + .004\*elasmu + prstp)\*gama;

\*Base Case Carbon Price

cpricebase(t)= cprice0\*(1+gcprice)\*\*(5\*(t.val-1));

VARIABLES

MIU(t) Emission control rate GHGs

FORC(t) Increase in radiative forcing (watts per m2 from 1900)

TATM(t) Increase temperature of atmosphere (degrees C from 1900)

TOCEAN(t) Increase temperatureof lower oceans (degrees C from 1900)

MAT(t) Carbon concentration increase in atmosphere (GtC from 1750)

MU(t) Carbon concentration increase in shallow oceans (GtC from 1750)

ML(t) Carbon concentration increase in lower oceans (GtC from 1750)

E(t) Total CO2 emissions (GtCO2 per year)

EIND(t) Industrial emissions (GtCO2 per year)

C(t) Consumption (trillions 2005 US dollars per year)

K(t) Capital stock (trillions 2005 US dollars)

CPC(t) Per capita consumption (thousands 2005 USD per year)

I(t) Investment (trillions 2005 USD per year)

S(t) Gross savings rate as fraction of gross world product

RI(t) Real interest rate (per annum)

Y(t) Gross world product net of abatement and damages (trillions 2005 USD per year)

YGROSS(t) Gross world product GROSS of abatement and damages (trillions 2005 USD per year)

YNET(t) Output net of damages equation (trillions 2005 USD per year)

DAMAGES(t) Damages (trillions 2005 USD per year)

DAMFRAC(t) Damages as fraction of gross output

ABATECOST(t) Cost of emissions reductions (trillions 2005 USD per year)

MCABATE(t) Marginal cost of abatement (2005$ per ton CO2)

CCA(t) Cumulative industrial carbon emissions (GTC)

CCATOT(t) Total carbon emissions (GtC)

PERIODU(t) One period utility function

CPRICE(t) Carbon price (2005$ per ton of CO2)

CEMUTOTPER(t) Period utility

UTILITY Welfare function;

NONNEGATIVE VARIABLES MIU, TATM, MAT, MU, ML, Y, YGROSS, C, K, I;

EQUATIONS

\*Emissions and Damages

EEQ(t) Emissions equation

EINDEQ(t) Industrial emissions

CCACCA(t) Cumulative industrial carbon emissions

CCATOTEQ(t) Cumulative total carbon emissions

FORCE(t) Radiative forcing equation

DAMFRACEQ(t) Equation for damage fraction

DAMEQ(t) Damage equation

ABATEEQ(t) Cost of emissions reductions equation

MCABATEEQ(t) Equation for MC abatement

CARBPRICEEQ(t) Carbon price equation from abatement

\*Climate and carbon cycle

MMAT(t) Atmospheric concentration equation

MMU(t) Shallow ocean concentration

MML(t) Lower ocean concentration

TATMEQ(t) Temperature-climate equation for atmosphere

TOCEANEQ(t) Temperature-climate equation for lower oceans

\*Economic variables

YGROSSEQ(t) Output gross equation

YNETEQ(t) Output net of damages equation

YY(t) Output net equation

CC(t) Consumption equation

CPCE(t) Per capita consumption definition

SEQ(t) Savings rate equation

KK(t) Capital balance equation

RIEQ(t) Interest rate equation

\* Utility

CEMUTOTPEREQ(t) Period utility

PERIODUEQ(t) Instantaneous utility function equation

UTIL Objective function ;

\*\* Equations of the model

\*Emissions and Damages

eeq(t).. E(t) =E= EIND(t) + etree(t);

eindeq(t).. EIND(t) =E= sigma(t) \* YGROSS(t) \* (1-(MIU(t)));

ccacca(t+1).. CCA(t+1) =E= CCA(t)+ EIND(t)\*5/3.666;

ccatoteq(t).. CCATOT(t) =E= CCA(t)+cumetree(t);

force(t).. FORC(t) =E= fco22x \* ((log((MAT(t)/588.000))/log(2))) + forcoth(t);

damfraceq(t) .. DAMFRAC(t) =E= (a1\*TATM(t))+(a2\*TATM(t)\*\*a3) ;

dameq(t).. DAMAGES(t) =E= YGROSS(t) \* DAMFRAC(t);

abateeq(t).. ABATECOST(t) =E= YGROSS(t) \* cost1(t) \* (MIU(t)\*\*expcost2);

mcabateeq(t).. MCABATE(t) =E= pbacktime(t) \* MIU(t)\*\*(expcost2-1);

carbpriceeq(t).. CPRICE(t) =E= pbacktime(t) \* (MIU(t))\*\*(expcost2-1);

\*Climate and carbon cycle

mmat(t+1).. MAT(t+1) =E= MAT(t)\*b11 + MU(t)\*b21 + (E(t)\*(5/3.666));

mml(t+1).. ML(t+1) =E= ML(t)\*b33 + MU(t)\*b23;

mmu(t+1).. MU(t+1) =E= MAT(t)\*b12 + MU(t)\*b22 + ML(t)\*b32;

tatmeq(t+1).. TATM(t+1) =E= TATM(t) + c1 \* ((FORC(t+1)-(fco22x/t2xco2)\*TATM(t))-(c3\*(TATM(t)-TOCEAN(t))));

toceaneq(t+1).. TOCEAN(t+1) =E= TOCEAN(t) + c4\*(TATM(t)-TOCEAN(t));

\*Economic variables

ygrosseq(t).. YGROSS(t) =E= (al(t)\*(L(t)/1000)\*\*(1-GAMA))\*(K(t)\*\*GAMA);

yneteq(t).. YNET(t) =E= YGROSS(t)\*(1-damfrac(t));

yy(t).. Y(t) =E= YNET(t) - ABATECOST(t);

cc(t).. C(t) =E= Y(t) - I(t);

cpce(t).. CPC(t) =E= 1000 \* C(t) / L(t);

seq(t).. I(t) =E= S(t) \* Y(t);

kk(t+1).. K(t+1) =L= (1-dk)\*\*tstep \* K(t) + tstep \* I(t);

rieq(t+1).. RI(t) =E= (1+prstp) \* (CPC(t+1)/CPC(t))\*\*(elasmu/tstep) - 1;

\*Utility

cemutotpereq(t).. CEMUTOTPER(t) =E= PERIODU(t) \* L(t) \* rr(t);

periodueq(t).. PERIODU(t) =E= ((C(T)\*1000/L(T))\*\*(1-elasmu)-1)/(1-elasmu)-1;

util.. UTILITY =E= tstep \* scale1 \* sum(t, CEMUTOTPER(t)) + scale2 ;

\*Resource limit

CCA.up(t) = fosslim;

\* Control rate limits

MIU.up(t) = limmiu;

MIU.up(t)$(t.val<30) = 1;

\*\* Upper and lower bounds for stability

K.LO(t) = 1;

MAT.LO(t) = 10;

MU.LO(t) = 100;

ML.LO(t) = 1000;

C.LO(t) = 2;

TOCEAN.UP(t) = 20;

TOCEAN.LO(t) = -1;

TATM.UP(t) = 20;

CPC.LO(t) = .01;

TATM.UP(t) = 12;

\* Control variables

set lag10(t) ;

lag10(t) = yes$(t.val gt card(t)-10);

S.FX(lag10(t)) = optlrsav;

\* Initial conditions

CCA.FX(tfirst) = 400;

K.FX(tfirst) = k0;

MAT.FX(tfirst) = mat0;

MU.FX(tfirst) = mu0;

ML.FX(tfirst) = ml0;

TATM.FX(tfirst) = tatm0;

TOCEAN.FX(tfirst) = tocean0;

\*\* Solution options

option iterlim = 99900;

option reslim = 99999;

option solprint = on;

option limrow = 0;

option limcol = 0;

model CO2 /all/;

\* For base run, this subroutine calculates Hotelling rents

\* Carbon price is maximum of Hotelling rent or baseline price

\* The cprice equation is different from 2013R. Not sure what went wrong.

If (ifopt eq 0,

a2 = 0;

solve CO2 maximizing UTILITY using nlp;

photel(t)=cprice.l(t);

a2 = a20;

cprice.up(t)$(t.val<tnopol+1) = max(photel(t),cpricebase(t));

);

miu.fx('1')$(ifopt=1) = miu0;

solve co2 maximizing utility using nlp;

solve co2 maximizing utility using nlp;

solve co2 maximizing utility using nlp;

\*\* POST-SOLVE

\* Calculate social cost of carbon and other variables

scc(t) = -1000\*eeq.m(t)/(.00001+cc.m(t));

atfrac(t) = ((mat.l(t)-588)/(ccatot.l(t)+.000001 ));

atfrac2010(t) = ((mat.l(t)-mat0)/(.00001+ccatot.l(t)-ccatot.l('1') ));

ppm(t) = mat.l(t)/2.13;

\* Produces a file "Dice2016R-091916ap.csv" in the base directory

\* For ALL relevant model outputs, see 'PutOutputAllT.gms' in the Include folder.

\* The statement at the end of the \*.lst file "Output..." will tell you where to find the file.

file results /Dice2016R-091916ap.csv/; results.nd = 10 ; results.nw = 0 ; results.pw=20000; results.pc=5;

put results;

put /"Results of DICE-2016R model run using model Dice2016R-091916ap.csv";

put /"This is optimal if ifopt = 1 and baseline if ifopt = 0";

put /"ifopt =" ifopt;

put // "Period";

Loop (T, put T.val);

put / "Year" ;

Loop (T, put (2010+(TSTEP\*T.val) ));

put / "Industrial Emissions GTCO2 per year" ;

Loop (T, put EIND.l(T));

put / "Atmospheric concentration C (ppm)" ;

Loop (T, put (MAT.l(T)/2.13));

put / "Atmospheric Temperature " ;

Loop (T, put TATM.l(T));

put / "Output Net Net) " ;

Loop (T, put Y.l(T));

put / "Climate Damages fraction output" ;

Loop (T, put DAMFRAC.l(T));

put / "Consumption Per Capita " ;

Loop (T, put CPC.l(T));

put / "Carbon Price (per t CO2)" ;

Loop (T, put cprice.l(T));

put / "Emissions Control Rate" ;

Loop (T, put MIU.l(T));

put / "Social cost of carbon" ;

Loop (T, put scc(T));

put / "Interest Rate " ;

Loop (T, put RI.l(T));

put / "Population" ;

Loop (T, put L(T));

put / "TFP" ;

Loop (T, put AL(T));

put / "Output gross,gross" ;

Loop (T, put YGROSS.L(t));

put / "Change tfp" ;

Loop (T, put ga(t));

put / "Capital" ;

Loop (T, put k.l(t));

put / "s" ;

Loop (T, put s.l(t));

put / "I" ;

Loop (T, put I.l(t));

put / "Y gross net" ;

Loop (T, put ynet.l(t));

put / "damages" ;

Loop (T, put damages.l(t));

put / "damfrac" ;

Loop (T, put damfrac.l(t));

put / "abatement" ;

Loop (T, put abatecost.l(t));

put / "sigma" ;

Loop (T, put sigma(t));

put / "Forcings" ;

Loop (T, put forc.l(t));

put / "Other Forcings" ;

Loop (T, put forcoth(t));

put / "Period utilty" ;

Loop (T, put periodu.l(t));

put / "Consumption" ;

Loop (T, put C.l(t));

put / "Objective" ;

put utility.l;

put / "Land emissions" ;

Loop (T, put etree(t));

put / "Cumulative ind emissions" ;

Loop (T, put cca.l(t));

put / "Cumulative total emissions" ;

Loop (T, put ccatot.l(t));

put / "Atmospheric concentrations Gt" ;

Loop (T, put mat.l(t));

put / "Atmospheric concentrations ppm" ;

Loop (T, put ppm(t));

put / "Total Emissions GTCO2 per year" ;

Loop (T, put E.l(T));

put / "Atmospheric concentrations upper" ;

Loop (T, put mu.l(t));

put / "Atmospheric concentrations lower" ;

Loop (T, put ml.l(t));

put / "Atmospheric fraction since 1850" ;

Loop (T, put atfrac(t));

put / "Atmospheric fraction since 2010" ;

Loop (T, put atfrac2010(t));

putclose;

display pbacktime;

1. Code for uncertainty runs

$ontext

This is DICE-2016R. It builds on DICE-2016 from May 2016. It updates the damage function

based on the work of Moffatt and Nordhaus.

This does uncertainty loop with new approach.

$offtext

$title DICE-2016 February 2016

set t Time periods (5 years per period) /1\*100/

sets tfirst(t), tlast(t), tearly(t), tlate(t)

\*\* Random sets

set kkdam conditional for damage /1\*5/

set kkprod conditional for output /1\*5/

set kkets conditional for temp sens /1\*5/

set kkcarb conditional for carbon cycle /1\*5/

set kksig conditional for emissions intensity /1\*5/

parameters

\*\* Availability of fossil fuels

fosslim Maximum cumulative extraction fossil fuels (GtC) /6000/

\*\*Time Step

tstep Years per Period /5/

\*\* If optimal control

ifopt Indicator where optimized is 1 and base is 0 /0/

\*\* Preferences

elasmu Elasticity of marginal utility of consumption / 1.45 /

prstp Initial rate of social time preference per year / .015 /

\*\* Population and technology

gama Capital elasticity in production function /.300 /

pop0 Initial world population 2015 (millions) /7403 /

popadj Growth rate to calibrate to 2050 pop projection /0.134 /

popasym Asymptotic population (millions) /11500 /

dk Depreciation rate on capital (per year) /.100 /

q0 Initial world gross output 2015 (trill 2010 USD) /105.5 /

k0 Initial capital value 2015 (trill 2010 USD) /223. /

a0 Initial level of total factor productivity /5.115 /

ga0 Initial growth rate for TFP per 5 years /0.076 /

dela Decline rate of TFP per 5 years /0.005 /

\*\* Emissions parameters

gsigma1 Initial growth of sigma (per year) /-0.0152 /

dsig Decline rate of decarbonization (per period) /-0.001 /

eland0 Carbon emissions from land 2015 (GtCO2 per year) / 2.6 /

deland Decline rate of land emissions (per period) / .115 /

e0 Industrial emissions 2015 (GtCO2 per year) /35.85 /

miu0 Initial emissions control rate for base case 2015 /.03 /

\*\* Carbon cycle

\* Initial Conditions

mat0 Initial Concentration in atmosphere 2015 (GtC) /851 /

mu0 Initial Concentration in upper strata 2015 (GtC) /460 /

ml0 Initial Concentration in lower strata 2015 (GtC) /1740 /

mateq Equilibrium concentration atmosphere (GtC) /588 /

mueq Equilibrium concentration in upper strata (GtC) /360 /

mleq Equilibrium concentration in lower strata (GtC) /1720 /

\* Flow paramaters

b12 Carbon cycle transition matrix /.12 /

b23 Carbon cycle transition matrix /0.007 /

\* These are for declaration and are defined later

b11 Carbon cycle transition matrix

b21 Carbon cycle transition matrix

b22 Carbon cycle transition matrix

b32 Carbon cycle transition matrix

b33 Carbon cycle transition matrix

sig0 Carbon intensity 2010 (kgCO2 per output 2005 USD 2010)

\*\* Climate model parameters

fex0 2015 forcings of non-CO2 GHG (Wm-2) / 0.5 /

fex1 2100 forcings of non-CO2 GHG (Wm-2) / 1.0 /

tocean0 Initial lower stratum temp change (C from 1900) /.0068 /

tatm0 Initial atmospheric temp change (C from 1900) /0.85 /

c1 Climate equation coefficient for upper level /0.1005 /

c3 Transfer coefficient upper to lower stratum /0.088 /

c4 Transfer coefficient for lower level /0.025 /

fco22x Forcings of equilibrium CO2 doubling (Wm-2) /3.6813 /

\*\* Climate damage parameters

a10 Initial damage intercept /0 /

a20 Initial damage quadratic term

a1 Damage intercept /0 /

a3 Damage exponent /2.00 /

\*\* Abatement cost

expcost2 Exponent of control cost function / 2.6 /

pback Cost of backstop 2010$ per tCO2 2015 / 550 /

gback Initial cost decline backstop cost per period / .025 /

limmiu Upper limit on control rate after 2150 / 1.2 /

tnopol Period before which no emissions controls base / 45 /

cprice0 Initial base carbon price (2010$ per tCO2) / 2 /

gcprice Growth rate of base carbon price per year /.02 /

\*\* Participation parameters

periodfullpart Period at which have full participation /21 /

partfract2010 Fraction of emissions under control in 2010 / 1 /

partfractfull Fraction of emissions under control at full time / 1 /

\*\* Scaling and inessential parameters

\* Note that these are unnecessary for the calculations

\* They ensure that MU of first period's consumption =1 and PV cons = PV utilty

scale1 Multiplicative scaling coefficient /0.0302455265681763 /

scale2 Additive scaling coefficient /-10993.704/

\* Loop scalars

t2xco2 /3.1/

a2base Damage quadratic term /0.00236 /

a2 Damage coef in equation

\* lam Climate model parameter

;

PARAMETERS

\* Loop vectors

lam(kkets)

loopdamcoef(kkdam)

loopprodAL(kkprod,t)

loopgA(kkprod,t)

loopetscoef(kkets)

loopsig(kksig)

loopcarb(kkcarb)

loopscc(kkdam,kkets,kkprod,kkcarb,kksig,t)

loopsigma(kkdam,kkets,kkprod,kkcarb,kksig,t)

looptemp(kkdam,kkets,kkprod,kkcarb,kksig,t)

loopmat(kkdam,kkets,kkprod,kkcarb,kksig,t)

loopdamf(kkdam,kkets,kkprod,kkcarb,kksig,t)

loopy(kkdam,kkets,kkprod,kkcarb,kksig,t)

loopem(kkdam,kkets,kkprod,kkcarb,kksig,t)

loopr(kkdam,kkets,kkprod,kkcarb,kksig,t)

loopmiu(kkdam,kkets,kkprod,kkcarb,kksig,t)

loopobjective(kkdam,kkets,kkprod,kkcarb,kksig)

scc(t)

;

parameter loopdamcoef(kkdam) Coefficients for damage equation

/ 1 0.00071 , 2 0.00173 , 3 0.00236 , 4 0.00299 , 5 0.00401 / ;

parameter loopGA0(kkprod) Coefficients for PROD equation

/ 1 0.00609 , 2 0.04940 , 3 0.07600 , 4 0.10254 , 5 0.14594 / ;

parameter loopetscoef(kkets) Coefficients for CLIMATE equation

/ 1 2.02848 , 2 2.60288 , 3 3.10000 , 4 3.45421 , 5 4.27501 / ;

parameter loopsig(kksig) Coefficients for emissions intensity

/ 1 -0.01932 , 2 -0.01680 , 3 -0.01520 , 4 -0.01340 , 5 -0.01048 / ;

parameter loopcarb(kkcarb) Coefficients for carbon cycle

/ 1 246.45146 , 2 305.72038 , 3 360.00000 , 4 398.67917 , 5 496.57460 / ;

;

loop(kkets, lam(kkets) = fco22x/ loopetscoef(kkets); );

loop(kkprod, loopga(kkprod,t) = loopGA0(kkprod)\*exp(-dela\*5\*((t.val-1))); );

loopprodAL(kkprod,'1') = a0;

\*loop(kkprod, loopprodAL(kkprod,t+1) = loopprodAL(kkprod,t)/ ((1-loopga(kkprod,t))););

\*loop(kkprod,loop(t, loopprodAL(kkprod,t+1) = loopprodAL(kkprod,t) ;););

loop(kkprod,loop(t, loopprodAL(kkprod,t+1) = loopprodAL(kkprod,t)/ ((1-loopga(kkprod,t)));););

\* ga(t)=ga0\*exp(-dela\*5\*((t.val-1)));

\* al("1") = a0; loop(t, al(t+1)=ALPROD(t)/((1-ga(t))););

PARAMETERS

l(t) Level of population and labor

ALPROD(t) Level of total factor productivity

sigma(t) CO2-equivalent-emissions output ratio

rr(t) Average utility social discount rate

ga(t) Growth rate of productivity from

forcoth(t) Exogenous forcing for other greenhouse gases

gl(t) Growth rate of labor

gcost1 Growth of cost factor

gsig(t) Change in sigma (cumulative improvement of energy efficiency)

etree(t) Emissions from deforestation

cumetree(t) Cumulative from land

cost1(t) Adjusted cost for backstop

partfract(t) Fraction of emissions in control regime

gfacpop(t) Growth factor population

pbacktime(t) Backstop price

optlrsav Optimal long-run savings rate used for transversality

scc(t) Social cost of carbon

cpricebase(t) Carbon price in base case

photel(t) Carbon Price under no damages (Hotelling rent condition)

ppm(t) Atmospheric concentrations parts per million

atfrac(t) Atmospheric share since 1850

atfrac2010(t) Atmospheric share since 2010

;

\* Program control definitions

tfirst(t) = yes$(t.val eq 1);

tlast(t) = yes$(t.val eq card(t));

\* Parameters for long-run consistency of carbon cycle

b11 = 1 - b12;

b21 = b12\*MATEQ/MUEQ;

b22 = 1 - b21 - b23;

b32 = b23\*mueq/mleq;

b33 = 1 - b32 ;

\* Further definitions of parameters

a20 = a2base;

sig0 = e0/(q0\*(1-miu0));

l("1") = pop0;

loop(t, l(t+1)=l(t););

loop(t, l(t+1)=l(t)\*(popasym/L(t))\*\*popadj ;);

gsig("1")=gsigma1; loop(t,gsig(t+1)=gsig(t)\*((1+dsig)\*\*tstep) ;);

sigma("1")=sig0; loop(t,sigma(t+1)=(sigma(t)\*exp(gsig(t)\*tstep)););

pbacktime(t)=pback\*(1-gback)\*\*(t.val-1);

cost1(t) = pbacktime(t)\*sigma(t)/expcost2/1000;

etree(t) = eland0\*(1-deland)\*\*(t.val-1);

cumetree("1")= 100; loop(t,cumetree(t+1)=cumetree(t)+etree(t)\*(5/3.666););

rr(t) = 1/((1+prstp)\*\*(tstep\*(t.val-1)));

forcoth(t) = fex0+ (1/17)\*(fex1-fex0)\*(t.val-1)$(t.val lt 18)+ (fex1-fex0)$(t.val ge 18);

optlrsav = (dk + .004)/(dk + .004\*elasmu + prstp)\*gama;

partfract(t)$(ord(T)>periodfullpart) = partfractfull;

partfract(t)$(ord(T)<periodfullpart+1) = partfract2010+(partfractfull-partfract2010)\*(ord(t)-1)/periodfullpart;

partfract("1")= partfract2010;

\*Base Case Carbon Price

cpricebase(t)= cprice0\*(1+gcprice)\*\*(5\*(t.val-1));

VARIABLES

MIU(t) Emission control rate GHGs

FORC(t) Increase in radiative forcing (watts per m2 from 1900)

TATM(t) Increase temperature of atmosphere (degrees C from 1900)

TOCEAN(t) Increase temperatureof lower oceans (degrees C from 1900)

MAT(t) Carbon concentration increase in atmosphere (GtC from 1750)

MU(t) Carbon concentration increase in shallow oceans (GtC from 1750)

ML(t) Carbon concentration increase in lower oceans (GtC from 1750)

E(t) Total CO2 emissions (GtCO2 per year)

EIND(t) Industrial emissions (GtCO2 per year)

C(t) Consumption (trillions 2005 US dollars per year)

K(t) Capital stock (trillions 2005 US dollars)

CPC(t) Per capita consumption (thousands 2005 USD per year)

I(t) Investment (trillions 2005 USD per year)

S(t) Gross savings rate as fraction of gross world product

RI(t) Real interest rate (per annum)

Y(t) Gross world product net of abatement and damages (trillions 2005 USD per year)

YGROSS(t) Gross world product GROSS of abatement and damages (trillions 2005 USD per year)

YNET(t) Output net of damages equation (trillions 2005 USD per year)

DAMAGES(t) Damages (trillions 2005 USD per year)

DAMFRAC(t) Damages as fraction of gross output

ABATECOST(t) Cost of emissions reductions (trillions 2005 USD per year)

MCABATE(t) Marginal cost of abatement (2005$ per ton CO2)

CCA(t) Cumulative industrial carbon emissions (GTC)

CCATOT(t) Total carbon emissions (GtC)

PERIODU(t) One period utility function

CPRICE(t) Carbon price (2005$ per ton of CO2)

CEMUTOTPER(t) Period utility

UTILITY Welfare function

;

NONNEGATIVE VARIABLES MIU, TATM, MAT, MU, ML, Y, YGROSS, C, K, I;

EQUATIONS

\*Emissions and Damages

EEQ(t) Emissions equation

EINDEQ(t) Industrial emissions

CCACCA(t) Cumulative industrial carbon emissions

CCATOTEQ(t) Cumulative total carbon emissions

FORCE(t) Radiative forcing equation

DAMFRACEQ(t) Equation for damage fraction

DAMEQ(t) Damage equation

ABATEEQ(t) Cost of emissions reductions equation

MCABATEEQ(t) Equation for MC abatement

CARBPRICEEQ(t) Carbon price equation from abatement

\*Climate and carbon cycle

MMAT(t) Atmospheric concentration equation

MMU(t) Shallow ocean concentration

MML(t) Lower ocean concentration

TATMEQ(t) Temperature-climate equation for atmosphere

TOCEANEQ(t) Temperature-climate equation for lower oceans

\*Economic variables

YGROSSEQ(t) Output gross equation

YNETEQ(t) Output net of damages equation

YY(t) Output net equation

CC(t) Consumption equation

CPCE(t) Per capita consumption definition

SEQ(t) Savings rate equation

KK(t) Capital balance equation

RIEQ(t) Interest rate equation

\* Utility

CEMUTOTPEREQ(t) Period utility

PERIODUEQ(t) Instantaneous utility function equation

UTIL Objective function ;

\*\* Equations of the model

\*Emissions and Damages

eeq(t).. E(t) =E= EIND(t) + etree(t);

eindeq(t).. EIND(t) =E= sigma(t) \* YGROSS(t) \* (1-(MIU(t)));

ccacca(t+1).. CCA(t+1) =E= CCA(t)+ EIND(t)\*5/3.666;

ccatoteq(t).. CCATOT(t) =E= CCA(t)+cumetree(t);

force(t).. FORC(t) =E= fco22x \* ((log((MAT(t)/588.000))/log(2))) + forcoth(t);

damfraceq(t) .. DAMFRAC(t) =E= (a1\*TATM(t))+(a2\*TATM(t)\*\*a3) ;

dameq(t).. DAMAGES(t) =E= YGROSS(t) \* DAMFRAC(t);

abateeq(t).. ABATECOST(t) =E= YGROSS(t) \* cost1(t) \* (MIU(t)\*\*expcost2) \* (partfract(t)\*\*(1-expcost2));

mcabateeq(t).. MCABATE(t) =E= pbacktime(t) \* MIU(t)\*\*(expcost2-1);

carbpriceeq(t).. CPRICE(t) =E= pbacktime(t) \* (MIU(t)/partfract(t))\*\*(expcost2-1);

\*Climate and carbon cycle

mmat(t+1).. MAT(t+1) =E= MAT(t)\*b11 + MU(t)\*b21 + (E(t)\*(5/3.666));

mml(t+1).. ML(t+1) =E= ML(t)\*b33 + MU(t)\*b23;

mmu(t+1).. MU(t+1) =E= MAT(t)\*b12 + MU(t)\*b22 + ML(t)\*b32;

tatmeq(t+1).. TATM(t+1) =E= TATM(t) + c1 \* ((FORC(t+1)-(fco22x/t2xco2)\*TATM(t))-(c3\*(TATM(t)-TOCEAN(t))));

toceaneq(t+1).. TOCEAN(t+1) =E= TOCEAN(t) + c4\*(TATM(t)-TOCEAN(t));

\*Economic variables

ygrosseq(t).. YGROSS(t) =E= (ALPROD(t)\*(L(t)/1000)\*\*(1-GAMA))\*(K(t)\*\*GAMA);

yneteq(t).. YNET(t) =E= YGROSS(t)\*(1-damfrac(t));

yy(t).. Y(t) =E= YNET(t) - ABATECOST(t);

cc(t).. C(t) =E= Y(t) - I(t);

cpce(t).. CPC(t) =E= 1000 \* C(t) / L(t);

seq(t).. I(t) =E= S(t) \* Y(t);

kk(t+1).. K(t+1) =L= (1-dk)\*\*tstep \* K(t) + tstep \* I(t);

rieq(t+1).. RI(t) =E= (1+prstp) \* (CPC(t+1)/CPC(t))\*\*(elasmu/tstep) - 1;

\*Utility

cemutotpereq(t).. CEMUTOTPER(t) =E= PERIODU(t) \* L(t) \* rr(t);

periodueq(t).. PERIODU(t) =E= ((C(T)\*1000/L(T))\*\*(1-elasmu)-1)/(1-elasmu)-1;

util.. UTILITY =E= tstep \* scale1 \* sum(t, CEMUTOTPER(t)) + scale2 ;

\*Resource limit

CCA.up(t) = fosslim;

\* Control rate limits

MIU.up(t) = limmiu\*partfract(t);

MIU.up(t)$(t.val<30) = 1;

\*\* Upper and lower bounds for stability

K.LO(t) = 1;

MAT.LO(t) = 10;

MU.LO(t) = 100;

ML.LO(t) = 1000;

C.LO(t) = 2;

TOCEAN.UP(t) = 20;

TOCEAN.LO(t) = -1;

TATM.UP(t) = 20;

CPC.LO(t) = .01;

TATM.UP(t) = 12;

\* Control variables

\* Set savings rate for steady state for last 10 periods

set lag10(t) ;

lag10(t) = yes$(t.val gt card(t)-10);

S.FX(lag10(t)) = optlrsav;

\*s.fx(t)=0.26;

\* Initial conditions

CCA.FX(tfirst) = 400;

K.FX(tfirst) = k0;

MAT.FX(tfirst) = mat0;

MU.FX(tfirst) = mu0;

ML.FX(tfirst) = ml0;

TATM.FX(tfirst) = tatm0;

TOCEAN.FX(tfirst) = tocean0;

\*\* Solution options

option iterlim = 99900;

option reslim = 99999;

option solprint = on;

option limrow = 0;

option limcol = 0;

model CO2 /all/;

\* For base run, this subroutine calculates Hotelling rents

\* Carbon price is maximum of Hotelling rent or baseline price

\* The cprice equation is different from 2013R. Not sure what went wrong.

miu.fx('1')$(ifopt=1) = miu0;

loop(kkets,

loop(kkprod,

loop(kkdam,

loop(kkcarb,

loop(kksig,

a2=loopdamcoef(kkdam);

t2xco2=loopetscoef(kkets);

ALPROD(t)= loopprodAL(kkprod,t);

mueq=loopcarb(kkcarb);

b11 = 1 - b12;

b21 = b12\*MATEQ/MUEQ;

b22 = 1 - b21 - b23;

b32 = b23\*mueq/mleq;

b33 = 1 - b32 ;

gsigma1=loopsig(kksig);

gsig("1")=gsigma1; loop(t,gsig(t+1)=gsig(t)\*((1+dsig)\*\*tstep) ;);

sigma("1")=sig0; loop(t,sigma(t+1)=(sigma(t)\*exp(gsig(t)\*tstep)););

If (ifopt eq 0,

a2 = 0;

solve CO2 maximizing UTILITY using nlp;

photel(t)=cprice.l(t);

a2 = loopdamcoef(kkdam);

cprice.up(t)$(t.val<tnopol+1) = max(photel(t),cpricebase(t));

);

solve co2 maximizing utility using nlp;

\*\* POST-SOLVE

\* Calculate social cost of carbon

loopscc(kkdam,kkets,kkprod,kkcarb,kksig,t) = -1000\*eeq.m(t)/(.00001+cc.m(t));

loopsigma(kkdam,kkets,kkprod,kkcarb,kksig,t) = sigma(t);

looptemp(kkdam,kkets,kkprod,kkcarb,kksig,t) = tatm.l(t);

loopmat(kkdam,kkets,kkprod,kkcarb,kksig,t) = mat.l(t);

loopy(kkdam,kkets,kkprod,kkcarb,kksig,t) = y.l(t);

loopem(kkdam,kkets,kkprod,kkcarb,kksig,t) = eind.l(t);

loopr(kkdam,kkets,kkprod,kkcarb,kksig,t) = ri.l(t);

loopmiu(kkdam,kkets,kkprod,kkcarb,kksig,t) = miu.l(t);

loopdamf(kkdam,kkets,kkprod,kkcarb,kksig,t) = damfrac.l(t);

loopobjective(kkdam,kkets,kkprod,kkcarb,kksig) = utility.l;

scc(t) = -1000\*eeq.m(t)/(.00001+cc.m(t));

);

);

);

);

);

\*\* Display at bottom of output for visual inspection

option decimals=4;

option decimals=6;

file results /DiceResults-loopv23I.csv/; results.nd = 10 ; results.nw = 0 ; results.pw=20000; results.pc=5;

put results;

put /"Results of DICE-2016R model with uncertainty DICE2016R-091216-loop-v23.gms";

put /"This is optimal if ifopt = 1 and baseline if ifopt = 0";

put /"ifopt =" ifopt;

put /

put "damcoef";

put "etscoef";

put "prodcoef(2100)";

put "carbcoef";

put "sigcoef";

put "scc(2015)";

put "scc(2020)";

put "temp(2050)";

put "temp(2100)";

put "temp(2200)";

put "carbconc((2050)";

put "carbconc((2100)";

put "carbconc((2200)";

put "output(2050)";

put "output(2100)";

put /

loop(kkdam,loop(kkets,loop(kkprod,loop(kkcarb, loop(kksig,

put loopdamcoef(kkdam);

put loopetscoef(kkets);

put loopga0(kkprod);

put loopcarb(kkcarb);

put loopsig(kksig);

put loopscc(kkdam,kkets,kkprod,kkcarb,kksig,'1');

put loopscc(kkdam,kkets,kkprod,kkcarb,kksig,'2');

put looptemp(kkdam,kkets,kkprod,kkcarb,kksig,'8');

put looptemp(kkdam,kkets,kkprod,kkcarb,kksig,'18');

put looptemp(kkdam,kkets,kkprod,kkcarb,kksig,'38');

put loopmat(kkdam,kkets,kkprod,kkcarb,kksig,'8');

put loopmat(kkdam,kkets,kkprod,kkcarb,kksig,'18');

put loopmat(kkdam,kkets,kkprod,kkcarb,kksig,'28');

put loopy(kkdam,kkets,kkprod,kkcarb,kksig,'8');

put loopy(kkdam,kkets,kkprod,kkcarb,kksig,'18');

put /

);););););

put /

file results2 /DiceResults-loopv23II.csv/; results2.nd = 10 ; results2.nw = 0 ; results2.pw=20000; results2.pc=5;

put results2;

put /"Results of DICE-2016R model with uncertainty DICE2016R-091216-loop-v23.gms";

put /"This is optimal if ifopt = 1 and baseline if ifopt = 0";

put /"ifopt =" ifopt;

put /

put "emis(2050)";

put "emis(2100)";

put "emis(2200)";

put "damfrac((2050)";

put "damfrac((2100)";

put "damfrac((2200)";

put "sigma((2050)";

put "sigma((2100)";

put "sigma((2200)";

put "r(2015)"

put "r(2100)"

put "Objective";

put /

loop(kkdam,loop(kkets,loop(kkprod,loop(kkcarb, loop(kksig,

put loopem(kkdam,kkets,kkprod,kkcarb,kksig,'8');

put loopem(kkdam,kkets,kkprod,kkcarb,kksig,'18');

put loopem(kkdam,kkets,kkprod,kkcarb,kksig,'28');

put loopdamf(kkdam,kkets,kkprod,kkcarb,kksig,'8');

put loopdamf(kkdam,kkets,kkprod,kkcarb,kksig,'18');

put loopdamf(kkdam,kkets,kkprod,kkcarb,kksig,'28');

put loopsigma(kkdam,kkets,kkprod,kkcarb,kksig,'1');

put loopsigma(kkdam,kkets,kkprod,kkcarb,kksig,'18');

put loopsigma(kkdam,kkets,kkprod,kkcarb,kksig,'28');

put loopr(kkdam,kkets,kkprod,kkcarb,kksig,'1');

put loopr(kkdam,kkets,kkprod,kkcarb,kksig,'18');

put loopobjective(kkdam,kkets,kkprod,kkcarb,kksig);

put /

);););););

put /