Decomposing the Integrated Assessment of Climate Change

Thomas F. Rutherford
ZEW and University of Colorado

Christoph Boehringer
ZEW and University of Heidelberg

June 29, 2006

Overview of Talk

1. Literature
2. A canonical model
3. The decomposition idea
4. Application to DICE
5. Using GNUPLOT with GAMS
6. Some illustrative computational exercises

The Message

- Integrated assessment models for climate policy design can be decomposed using a linear approximation of the climate system.

- This permits models of the economic and natural science components to be processed independently on different time scales.

- Turnpike properties of Ramsey growth model permit a precise representation of post-terminal emissions and to reduce the requisite economic horizon.

- Decomposition accommodates economic modelling in a complementarity format thereby providing a means of incorporating second-best effects, e.g. distortionary taxes.

Integrated Assessment

Integrated models of climate and economy:

- First appeared in the 1980s as a paradigm for integrating science and economic policy instruments to study complex environmental issues.

- Combine complementarity knowledge from various discplines to produce informed policy analysis.

- Early example: RAINS model of acidification in Europe (Alcamo et al., 1985)

- Variety of models have been developed to study greenhouse issues (see Weyant et al. 1996, Parson and Fisher-Vanden 1997 and Kelly and Kolstad 1999).

- Nordhaus, Peck and Tysberg, and Manne and Richels in the 1990s.

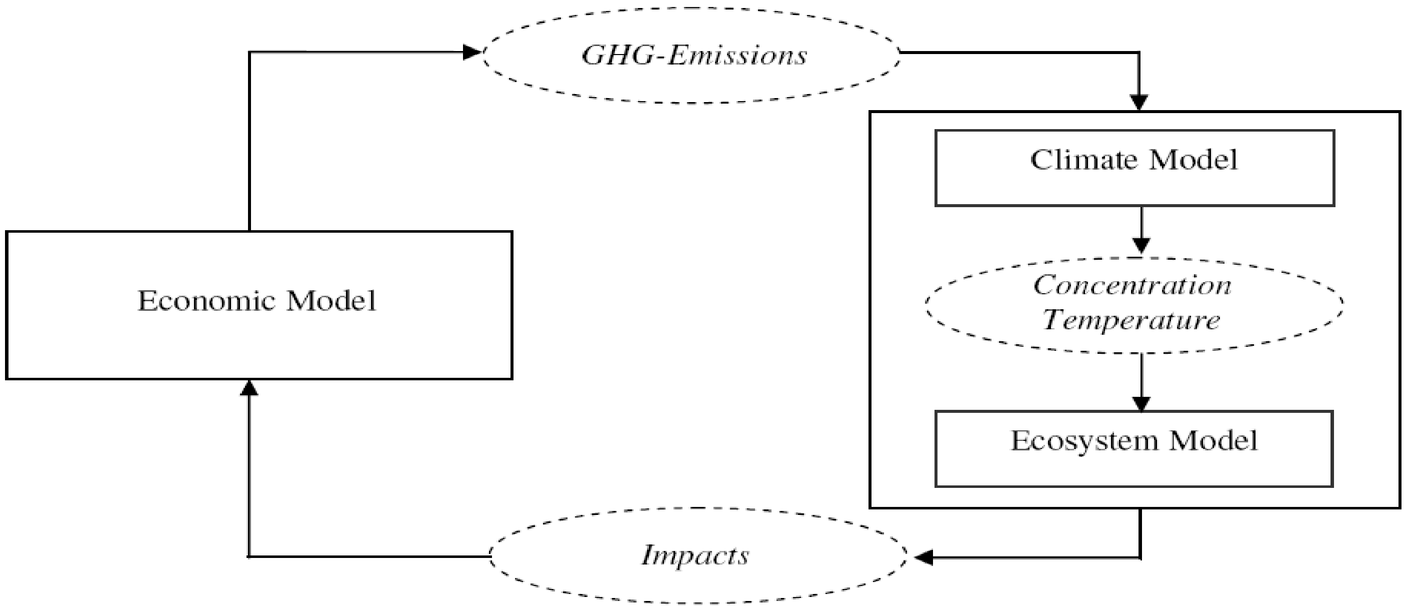


Figure 1: *Schematic Structure of Integrated Assessment Models for Climate Change*

Two Types of Integrated Assessment Models

1. *Policy simulation models* for assessment of specific measures (e.g. IMAGE by Rotmans), and

2. *Policy optimization models*, which seek to characterize optimal policies. (e.g., DICE by Nordhaus or MERGE by Manne and Richels).

Two Difficulties with Policy Optimization IAMs

1. Existing integrated assessment models must be solved over a long time horizon to provide a consistent accounting of both the costs and benefits of climate policy measures.

2. These models are typically solved as centralized planner optimization programs which do not readily admit second-best effects such as pre-existing taxation.

Our Contributions

1. We demonstrate that a tangent approximation of the the climate system provides a excellent means of decomposing integrated assessment models,

2. When the climate system is thus approximated, the economic model can be formulated:

   (a) over policy-relevant time horizon,

   (b) in a complementarity format which accomodates a wider range of econmic complexities (e.g., more goods, more regions, technical change, tax revenue recycling etc.)

3. When an IAM is decomposed, it becomes a simple matter to compare results from alternative climate/economic components.

A Canonical Integrated Assessment Model

A stylized optimizing IAM model:

$$\max \sum_{t=0}^{\infty} \left(\frac{1}{1+\rho}\right)^t U(C_t, D_t) \tag{1}$$

$$\begin{aligned} \text{s.t.} \qquad C_t &= F(K_t, D_t, E_t) - I_t \\ K_{t+1} &= (1-\delta)K_t + I_t \\ \\ K_0 &= \bar{K}_0 \end{aligned}$$

In which:

ρ is the discount rate,

U denotes instantaneous utility reflecting both final consumption and the disutility of climate damages,

C_t represents consumption in period t,

F characterizes aggregate production in period t as a function of capital, damages (with potentially adverse effects on productivity), and emissions,

K_t is the capital stock in period t (with $K_0 = \bar{K}_0$ as the initial capital stock),

E_t are emissions in period t,

I_t is investment in period t,

$$
\begin{aligned}
T_t^E &= H(S_t) \\
S_{t+1} &= G(S_t, E_t) \\
D_t &= D_t(T_t^E) \\
S_0 &= \bar{S}_0
\end{aligned}
$$

In which:

T_t^E is the global mean temperature in period t,

H describes the functional relationship between the climate state and temperature,

S_t is a vector of the climate state (with

D_t denotes damages of climate change in period t, $S_0 = \bar{S}_0$ as the initial climate state), and

G characterizes the motion of the climate state as a function of the previous climate state and current anthropogenic emissions.

Approximation

Merge the relationships $T_t^E = H(S_t)$ and $S_{t+1} = G(S_t, E_t)$ into a single equivalent equation

$$T_t^E = \Gamma_t(S_0, E_0, E_1, ..., E_{t-1}),$$

where Γ_t relates temperature in period t as a function of the initial climate state and emissions in all previous periods.

Then compute a *linear approximation* of the climate response:

$$T_t^E \approx \bar{T}_t^E + \sum_{\tau=0}^{t} \gamma_{t\tau}(E_\tau - \bar{E}_\tau)$$

For a simple application, we can do this by "brute force":

$$\gamma_{t\tau} \approx \frac{\bar{T}_t^E - \Gamma_t(S_0, E_0, ..., \bar{E}_\tau + \epsilon, ..., \bar{E}_{t-1})}{\epsilon}.$$

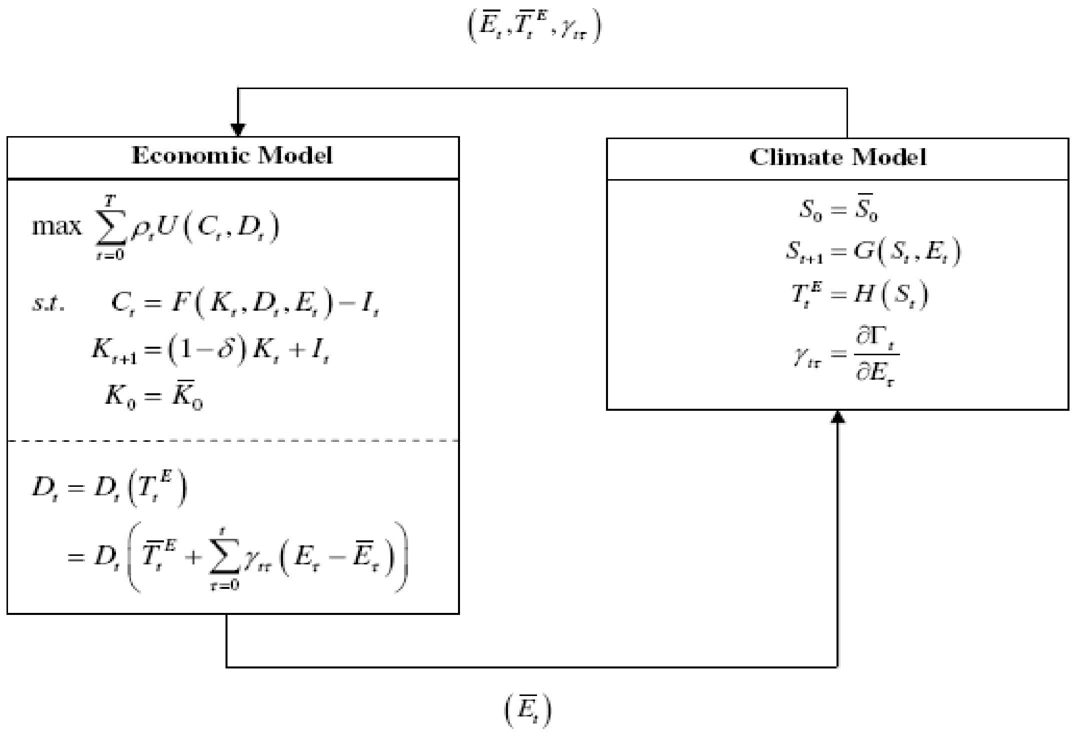


Figure 2: *Basic Decomposition Approach*

Comments

- Numerical differencing is only computationally tractable for small-scale climate models with solution times measured in seconds, however for larger scale models *adjoint codes* may be used for the same purpose.

- A benefit we perceive is that we can use this approach to provide a decomposition of the relative importance of climate and economic models in a given policy assessment.

- A more subtle advantage of the decomposition relates to differences in the nature of time scales for economic and climate models.

The DICE Climate Model

```
parameters
        E(t)               Anthropogenic carbon emissions (economic input)

        m(t)               CO2-equiv concentration billion t
        forc(t)            Radiative forcing - W per m2
        forcoth(tc)        Exogenous forcings from other greenhouse gases,
        te(t)              Temperature - atmosphere C
        tl(t)              Temperature - lower ocean C
        termv              Terminal value of atmophere
        deltaE             Difference iterval /0.001/;

m(t) = m0;  te(t) = t0;  tl(t) = tl0;  forcoth(tc) = 1.42;

loop(t,
   m(t)     = 590 + atret*E(t) + (1-deltam)*(m(t-1)-590) + m0$tfirst(t);
   forc(t)  = 4.1*(LOG(m(t)/590)/LOG(2)) + forcoth(t);
   te(t)    = te(t-1)+c1*(forc(t-1)-lam*te(t-1)-c3*(te(t-1)-tl(t-1))) + t0$tfirst(t);
   tl(t)    = tl(t-1)+c4*(te(t-1)-tl(t-1)) + tl0$tfirst(t);
   teref(t) = te(t);
);
```

Post-Terminal Projection

The Ramsey model, which provides the basis for nearly all policy-oriented IAMs, is an "exogenous growth model" (see Barro and Sala-i-Martin, Chapter 2).

Policy measures affect *levels* but not *growth rates*.

We can therefore easily extrapolate carbon emissions from the terminal period off the end of the model:

$$-p_t \frac{\partial F}{\partial E_t} = \sum_{\tau=t}^{\infty} \frac{\partial \Gamma_\tau}{\partial E_t} p_\tau^D \approx \sum_{\tau=t}^{T} \frac{\partial \Gamma_\tau}{\partial E_t} p_\tau^D + \frac{\tilde{p}_T^D}{(1+r)^{\tau-T}} \sum_{\tau=T+1}^{\infty} \frac{\partial \Gamma_\tau}{\partial E_t}$$

and

$$E_t \approx E_T \frac{L_t}{L_T} \quad \forall t > T$$

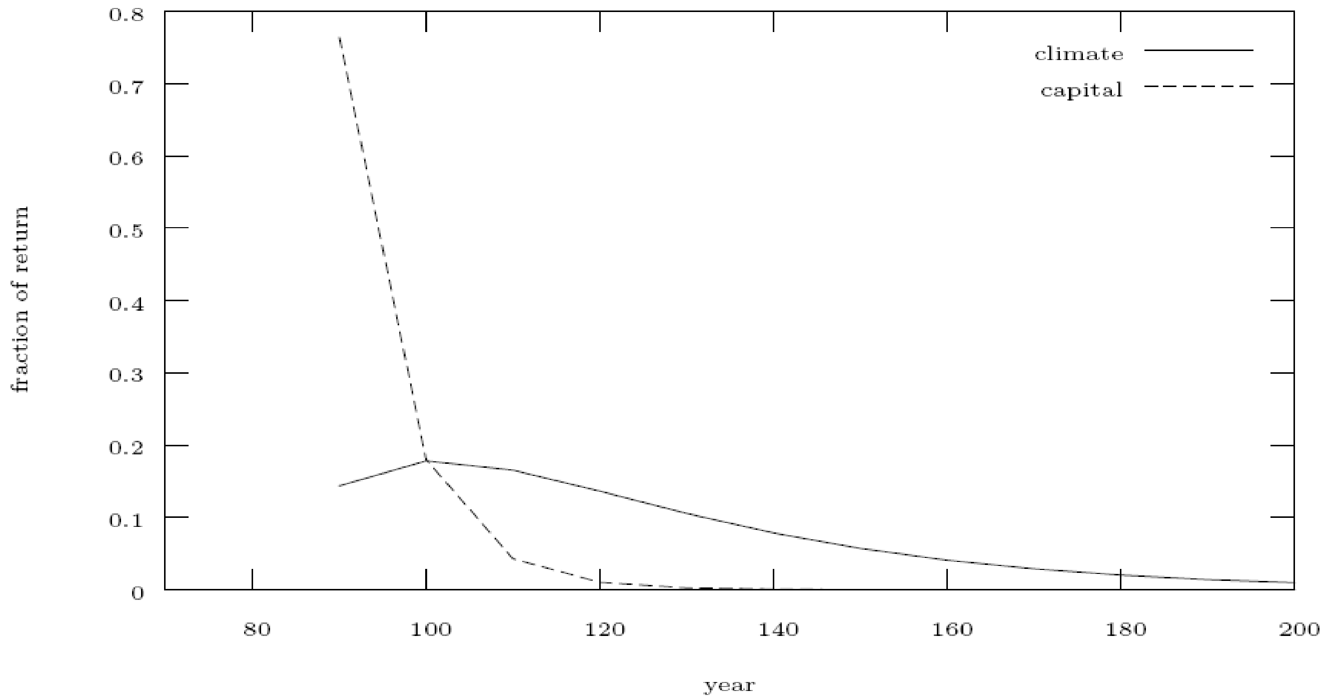


Figure 3: *Time Structure of Returns to Economic and Climate Investments*

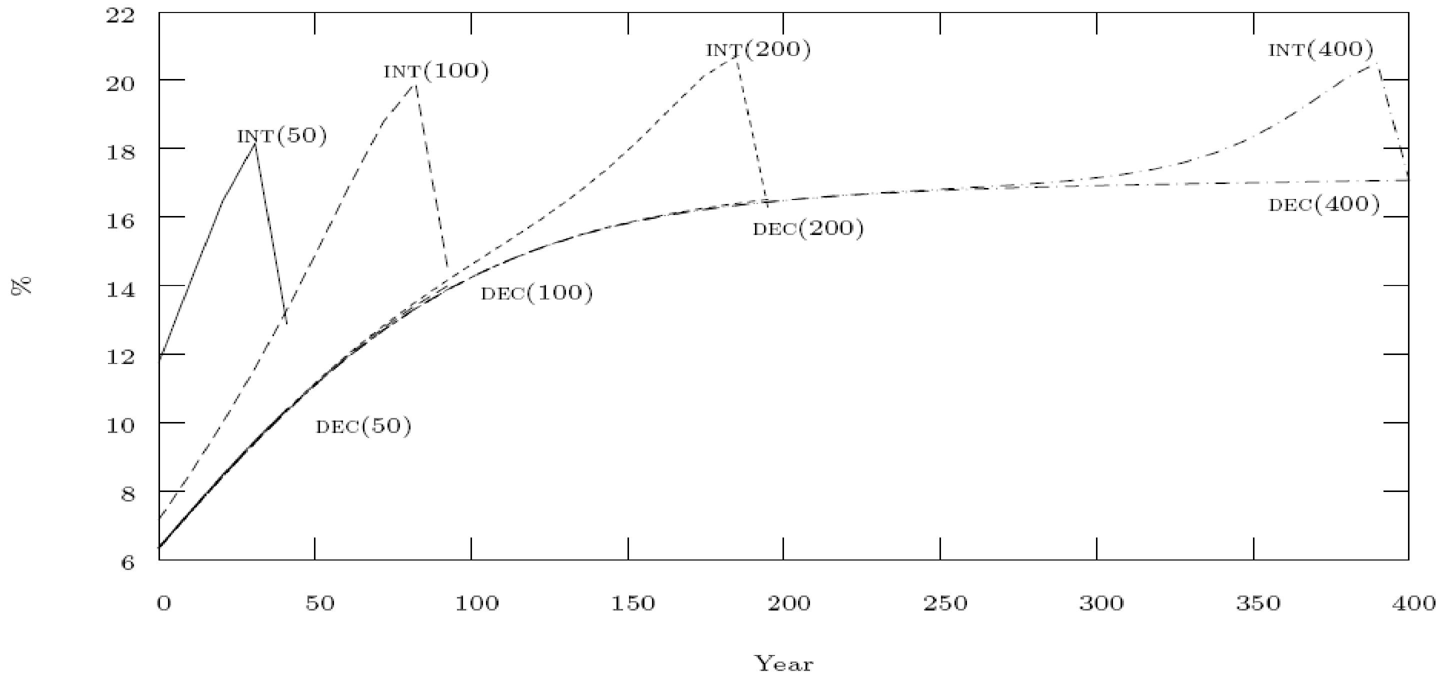


Figure 4: *Sensitivity of Emission Control Rate*

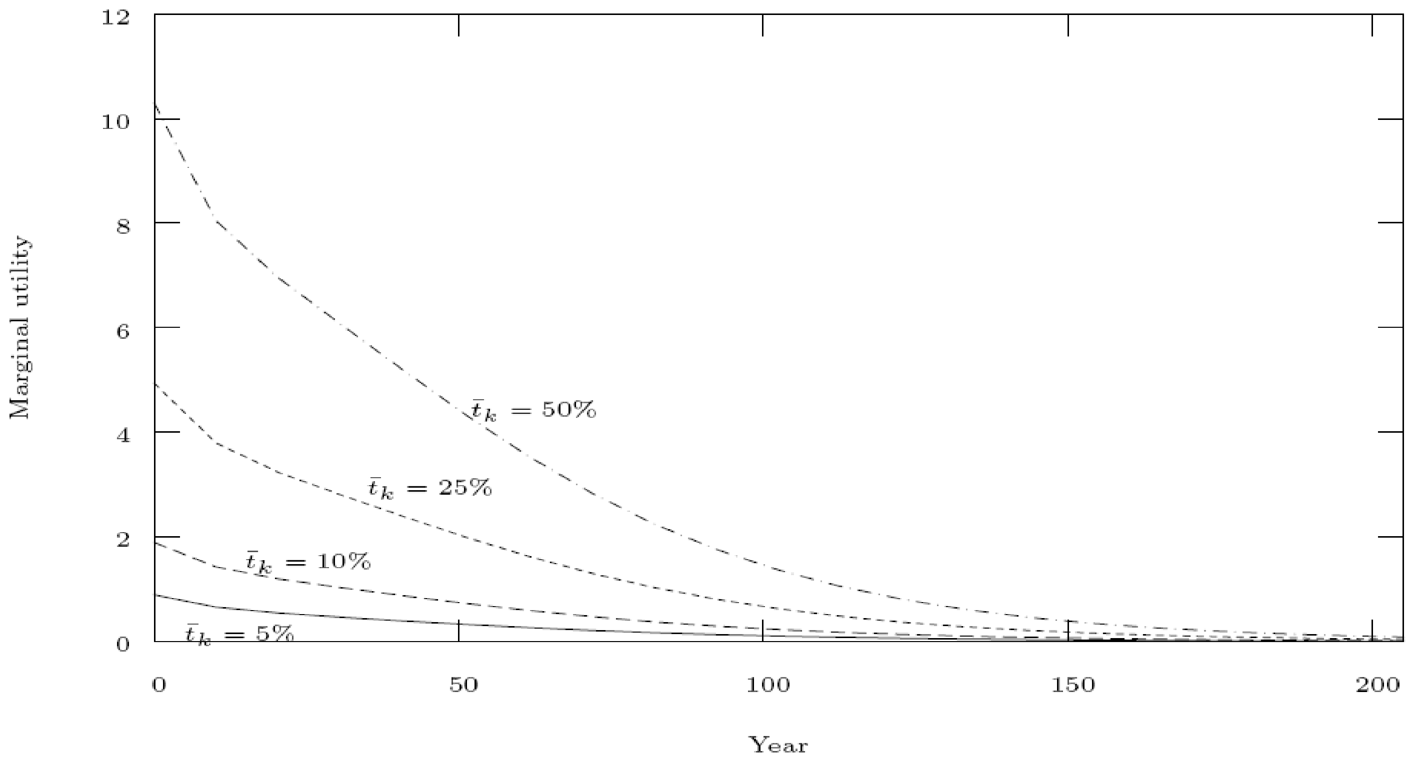
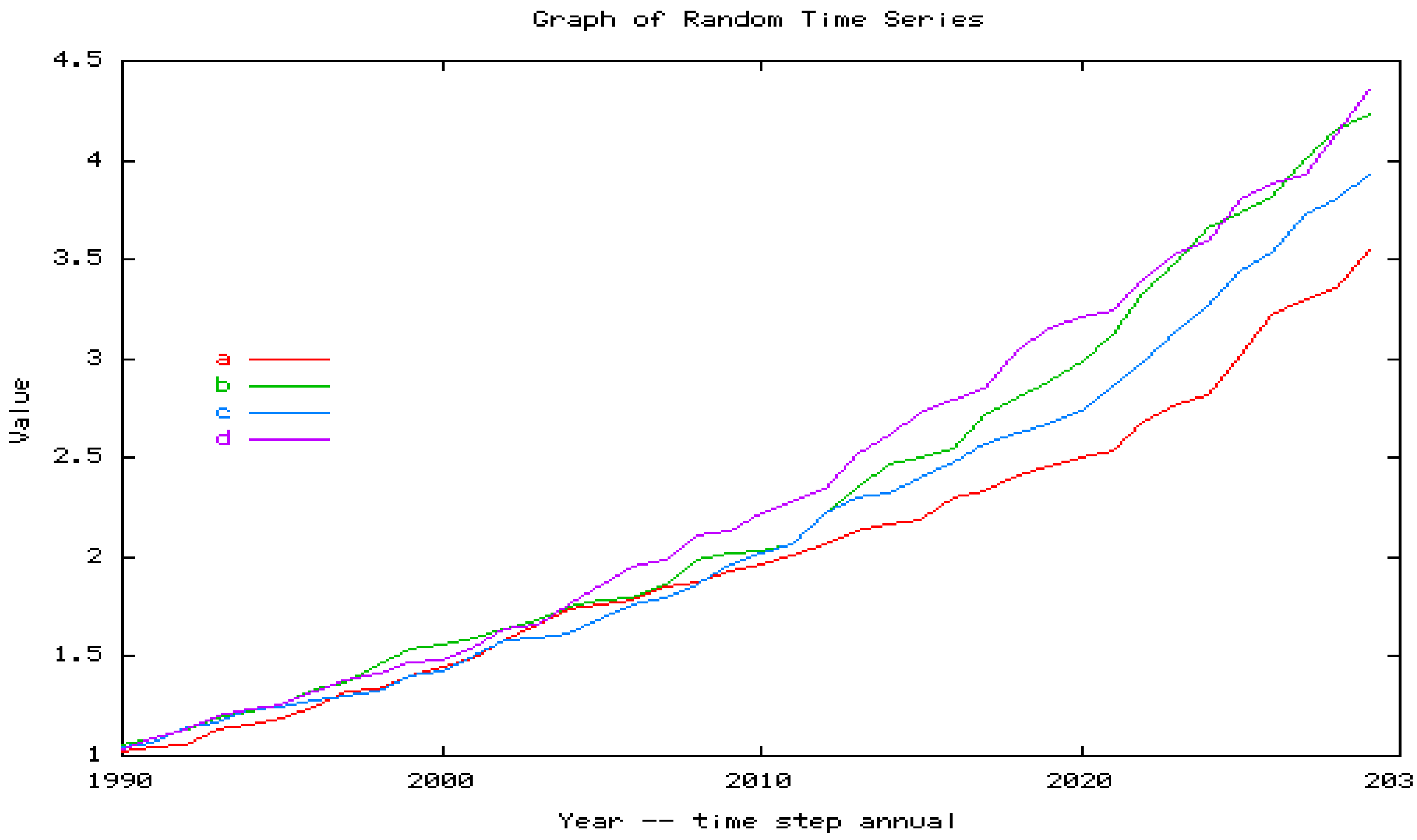


Figure 5: *Welfare Impact of 1% Increase in Abatement*

Using GNUPLOT from GAMS

```
set     t       Time periods in the model /0*40/;
        tlbl(t) Labels for the graph / 0 1990, 10 2000, 20 2010, 30 2020, 40 2030/;

parameter a(t,*)  My model output;

... a(t,*) is loaded from your GAMS model.

$setglobal labels decade

*       Here are the GNUPLOT commands:

$setglobal gp_opt1 "set size 0.8,0.8"
$setglobal gp_opt2 "set key  5,3"
$setglobal gp_opt3 "set title 'Graph of Random Time Series'"
$setglobal gp_opt4 "set yrange [0:3]"
$setglobal gp_opt5 "set xlabel 'Year -- time step annual'"
$setglobal gp_opt6 "set ylabel 'Value'"

$setglobal domain t
$setglobal labels tlbl
$batinclude plot A
```



Graph of Random Time Series
4.5
4
3.5
3
2.5
2
1.5
1
Value
a
b
c
d
1990
2000
2010
2020
203
Year -- time step annual

Climate 1-2-3

GAMS programs `climate1.gms`, `climate2.gms`, and `climate3.gms` provides a simplified illustration of ideas in our paper. Your progamming task involves *replicating* the graphs which illustrate these calculation.

Here are a few *extra credit* questions for students who are able to produce the plots with relative ease:

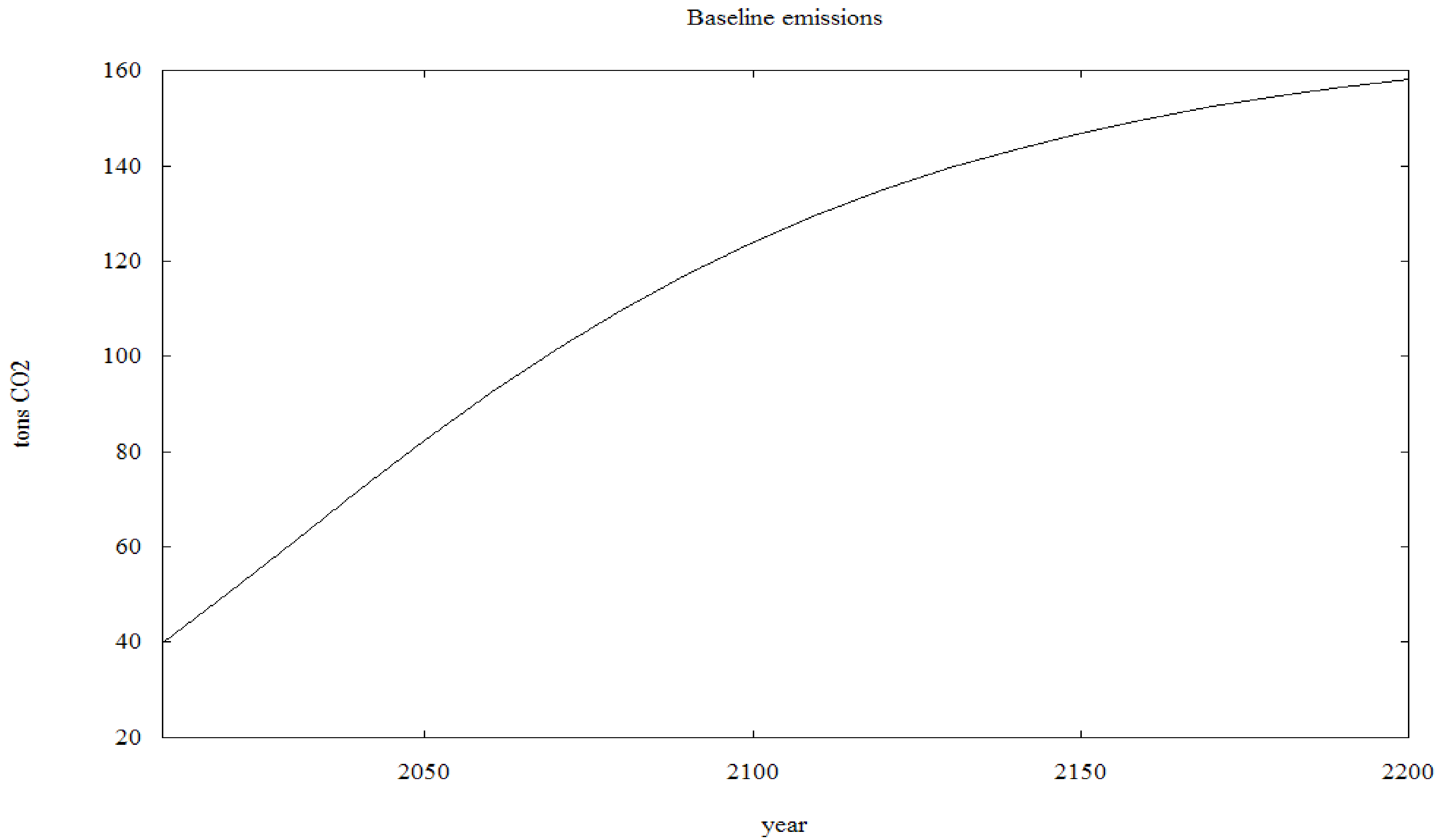
1. The abatement timing model presented here embodies the assumption that in the initial year of climate action abatement cannot exceed 10% of baseline emissions, and thereafter it may growh at only 20% per decade. From a qualitative perspective,how are the optimal policies affected by these assumptions?

2. How are economic costs affected when the stabilization target (specified here as 2 degrees) varies from 1.5 to 5 degrees?

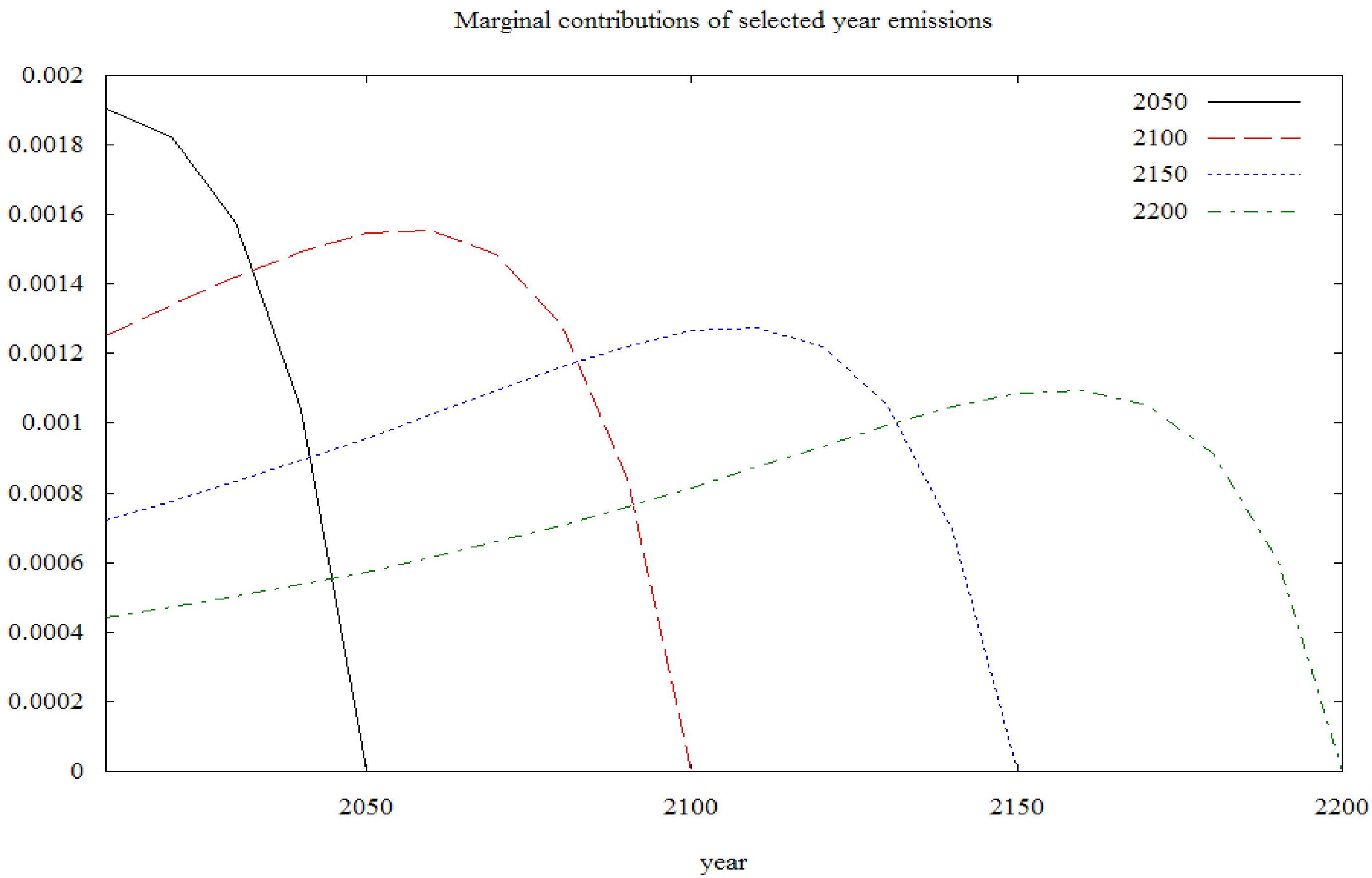
3. How sensitive is the optimal abatement policy affected by the intertemporal discount rate?

4. How do changes in the baseline emissions assumptions affect the estimated cost of climate stabilization?

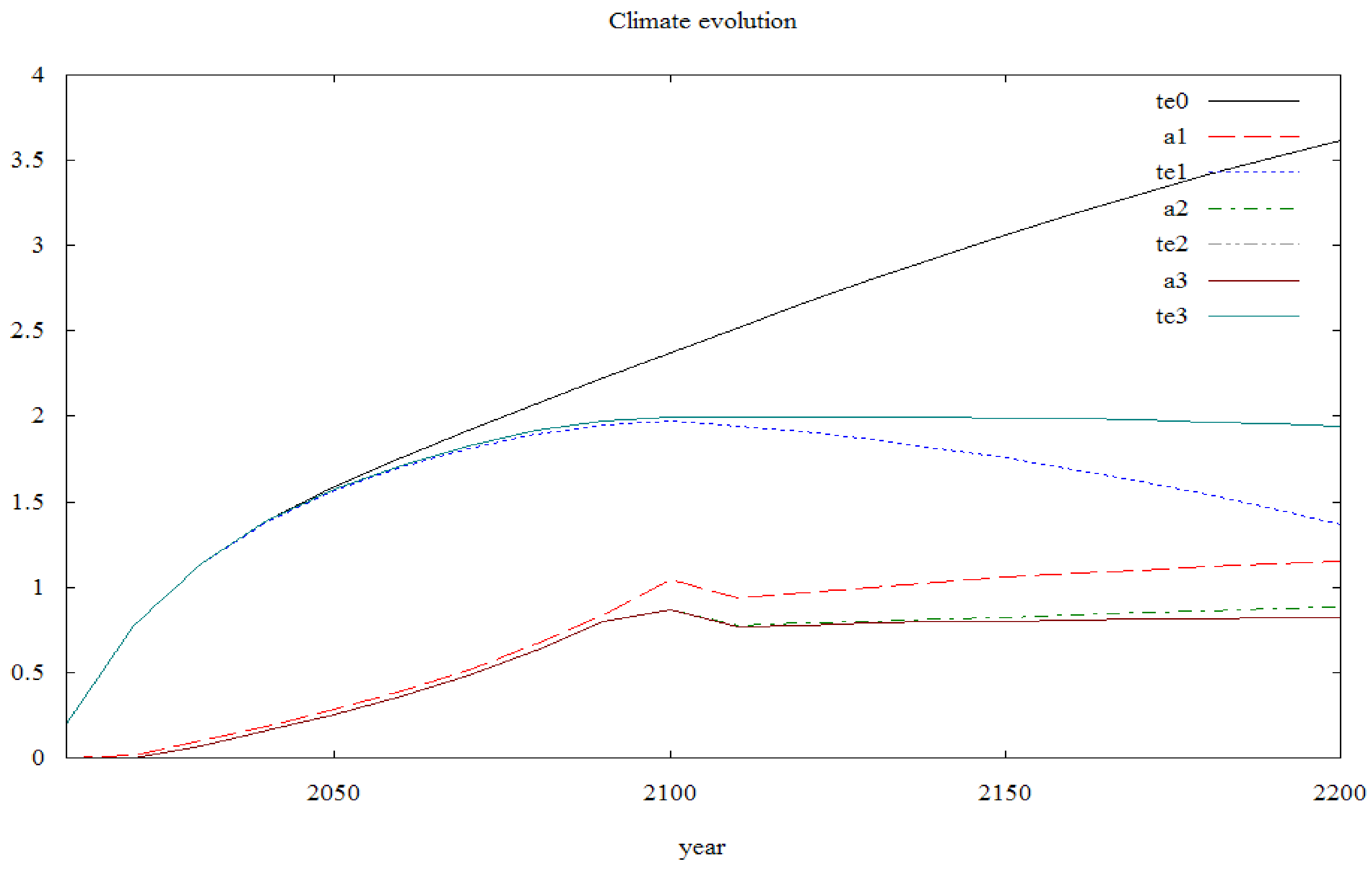
In `climate1.gms` you will need to introduce four `$setglobal` statements and one `$batinclude plot` statement to produce the following graph:



In `climate2.gms` you will need to declare and define a parameter in order to produce the following graph:



In `climate3.gms` you will need to declare and define a reporting parameter and include two additional solve statements to produce the following figure. `a1`, `a2` and `a3` report abatement as a fraction of baseline emisions in the first, second and third optimization problems:



Climate evolution
4
3.5
3
2.5
2
1.5
1
0.5
0
2050
2100
2150
2200
year
te0
a1
te1
a2
te2
a3
te3

climate1.gms

```
$title  climate1.gms    Data and an emissions growth path

SET     t               Time periods /1*40/;

parameter
        m0              CO2-equiv concent. 1965 billion tons carbon     /677/,
        tl0             Lower stratum temperature (C) 1965              /.10/,
        t0              Atmospheric temperature (C) 1965                /.2/,
        atret           Marginal atmospheric retention rate             /.64/,
        c1              Coefficient for upper level                     /.226/,
        lam             Climate feedback factor                         /1.41/,
        c3              Coefficient trans upper to lower stratum        /.440/,
        c4              Coeff of transfer for lower level               /.02/,

        r               Rate of social time preference per year         /.03/,
        gl0             Growth rate of population per decade            /.223/,
        dlab            Decline rate of population growth per dec       /.195/,
        deltam          Removal rate carbon per decade                  /.0833/,
        ga0             Initial growth rate for technology per decade   /.15/,
        dela            Decline rate of technology per decade           /.11/,
```

```
        gsigma          Growth of sigma per decade                    /-.1168/,
        sig0            CO2-equiv-GWP ratio                           /.519/,
        sigma(t)        Emissions-output ratio,
        L0              1965 world population millions                /3369/,
        k0              1965 value capital billions 1989 US dollars   /16.03/,
        gamma           Capital elasticity in output                  /.25/,

        a0              Initial level of total factor productivity    /.00963/,
        L(t)            Level of population and labor,
        al(t)           Level of total factor productivity (TFP),
        ga(t)           Growth rate of TFP from 0 to T,
        gl(t)           Growth rate of labor 0 to T,
        gsig(t)         Cumulative improvement of energy efficiency
        ebau(t)         Baseline emissions;

gsig(t) = (gsigma/dela)*(1-EXP(-dela*(ORD(t)-1)));
sigma(t)=sig0*EXP(gsig(t));
gl(t) = (gl0/dlab)*(1-EXP(-dlab*(ORD(t)-1)));
L(t)=L0*EXP(gl(t))*.9;
ga(t)= (ga0/dela)*(1-EXP(-dela*(ORD(t)-1)));
al(t) =a0*EXP(ga(t));
```

```
ebau(t) = 10 * sigma(t) * al(t) * (k0*L(t)/L0)**gamma * L(t)**(1-gamma);

set     t200(t)         /1*20/,
        tlbl(t)         /5 2050, 10 2100, 15 2150, 20 2200/;
```

climate2.gms

```
$title  climate2.gms    Computation of Climate Response

*       Include the preceding file:

$include climate1

parameter
        m(t)            CO2-equiv concentration billion t
        forc(t)         Radiative forcing - W per m2
        forcoth(t)      Exogenous forcings from other greenhouse gases,
        te(t)           Temperature - atmosphere C
        teref(t)        Reference temperature path
        tl(t)           Temperature - lower ocean C
        termv           Terminal value of atmophere
        deltaE          Difference iterval /0.001/;

set     tfirst(t)       The first time period;   tfirst(t) = yes$(ord(t)=1);

*       Initial conditions for climate model:
```

```
m(tfirst) = m0;  te(tfirst) = t0;  tl(tfirst) = tl0;  forcoth(t) = 1.42;

parameter       climate         Climate evolution
                eref(t)         Reference emissions
                E(t)            Currently estimated emissions path,
                grad            Temperature gradient,
                teinit          Initial temperature path;

*       Write out two "subroutines" for computing the climate model:

$onecho >climatemodel.gms
loop(t,

'Atmospheric carbon accumulation:

  m(t)    = 590 + atret*eref(t) + (1-deltam)*(m(t-1)-590) + m0$tfirst(t);

* This equation relates the stock of atmospheric carbon to
* forcing, with a climate sensitivity of 4.1 and a
* pre-industrial carbon concentration of 590 parts per million:
```

```
  forc(t) = 4.1*(LOG(m(t)/590)/LOG(2)) + forcoth(t);

* Thes equations relate forcing to climate change.  Higher
* radiative forcings warm the atmospheric layer:

  te(t)   = te(t-1)+c1*(forc(t-1)-lam*te(t-1)-c3*(te(t-1)-tl(t-1))) + t0$tfirst(t);

* The atmosphere then  warms the upper ocean, gradually
* warming the deep oceans:

  tl(t)   = tl(t-1)+c4*(te(t-1)-tl(t-1)) + tl0$tfirst(t);

  teref(t) = te(t);
);
$offecho

alias (t,tp);

$onecho >jacobian.gms
        eref(t) = E(t);
$include climatemodel
```

```
        teinit(t) = teref(t);
        grad(t,tp) = 0;
        loop(tp,eref(tp) = eref(tp) + deltaE;
$include climatemodel
          grad(t,tp)= (teref(t)-teinit(t)) / deltaE;
          eref(tp)  = eref(tp) - deltaE;);
          teref(t)  = teinit(t);
$offecho

E(t) = ebau(t);
$include jacobian
```

climate3.gms

```
$title  climate3.gms    Optimal Abatement

*       Include both preceding files:

$include climate2

parameter               pv(t)           Present value cost;

pv(t) = 1/(1+r)**(ord(t)-1);

variables       obj             Objective function
                abate(t)        Abatement measures;

positive variable abate;

equations       objdef, avetemp, ratelimit;

objdef..                obj =e= sum(t, pv(t) * ABATE(t));

ratelimit(t+1)..        ABATE(t+1) =l= 1.2 * ABATE(t) + 0.10*Ebau(t);
```

```
avetemp(t)..            teref(t) + sum(tp, grad(t,tp)*(Ebau(tp)-ABATE(tp)-eref(tp))) =l=

model optabate /objdef, avetemp, ratelimit/;

*       Solve iteration 1:

solve optabate using nlp minimizing obj;
E(t) = ebau(t) - ABATE.L(t);
$include jacobian
```