シミュレーションのプログラム

武田史郎*

2004 年 5 月

1　シミュレーションのためのモデル

補論でモデルの構造の説明はおこなっている。しかし、シミュレーションをおこなう GAMS プログラムでは、calibrated share form を用いてモデルを記述している。ここでは、シミュレーションのプログラムの理解を助けるために、プログラム内でのモデルの記述に沿った形でモデルを提示する。

1.1　Calibrated share form

まず、Calibrated share form について説明しておこう。Calibrated share form については **?** が詳しい。以下、一つの生産物、二つの投入物を持つ CES 生産関数を例にして説明をおこなう。q を生産量、x_1、x_2 をそれぞれ投入物 1 と 2 の投入量とすると、この生産関数は次のように表現できる。

$$q = \phi\left[\alpha x_1^{\frac{\sigma-1}{\sigma}} + (1-\alpha)x_2^{\frac{\sigma-1}{\sigma}}\right]^{\frac{\sigma}{\sigma-1}} = \left[\beta_1 x_1^{\frac{\sigma-1}{\sigma}} + \beta_2 x_2^{\frac{\sigma-1}{\sigma}}\right]^{\frac{\sigma}{\sigma-1}} \tag{1}$$

ここには三つの未知のパラメータが含まれている。すなわち、投入物間の代替の弾力性 σ、及び二つのシェアパラメータ β_1、β_2 である[1]。シミュレーションをおこなうには、全てのパラメータを特定化する必要があるが、応用一般均衡分析 (AGE 分析) では、これらのパラメータの特定化に次のような方法がとられることが多い。

- 代替の弾力性は外生的に与える。
- シェアパラメータはカリブレートする。

シェアパラメータのカリブレートは、具体的には次のようにおこなわれる。まず、AGE 分析では、基準となる時点でのデータ (ベンチマークデータ) が与えられている。$\bar{y}$、$\bar{x}_1$、$\bar{x}_2$、$\bar{p}_1$、$\bar{p}_2$ をそれぞれ基準時点での生産量、投入量、投入物の価格のベンチマークデータとしよう。さらに、このベンチマークデータは均衡条件を満たしているものと仮定される。これは、ベンチマークデータのもとで生産者が費用最小化を達成しているということを意味する。従って、上のデータは次の式を満たしていなければならない。

$$\bar{x}_i = \bar{q}\left[\frac{\beta_i}{\bar{p}_i}\left(\sum_{j=1,2}\beta_j^{\sigma}\bar{p}_j^{1-\sigma}\right)^{\frac{1}{1-\sigma}}\right]^{\sigma} \qquad i = 1,2 \tag{2}$$

*連絡先、email: <zbc08106@park.zero.ad.jp>. シミュレーションで用いたデータ、及び GAMS のプログラムは全て筆者から入手可能である。

[1]厳密には、β_i はなんらかの「シェア」を表しているわけではないが、ここでは便宜上シェアパラメータと呼ぶ。

これは、(1) 式の CES 生産関数から導かれた条件付き要素需要関数である。ここで、σ は外生的に与えられるので、上の二本の式には、二個の未知数 (β_1, β_2) しか含まれない。よって、この方程式を以下のように β_1, β_2 について解くことができる。

$$\bar{\beta}_i = \theta_i \left(\frac{\bar{x}_i}{\bar{y}} \right)^{\frac{1-\sigma}{\sigma}} \qquad i = 1, 2 \tag{3}$$

ここで、θ_i はベンチマークデータにおける投入物 i の費用シェア $\bar{p}_i\bar{x}_i / \sum_j \bar{p}_j\bar{x}_j$ である。これがカリブレートされたシェアパラメータである。カリブレートされたシェアパラメータはベンチマークデータと代替の弾力性の関数になっている。

シミュレーションをおこなう際には、生産関数に (1) 式を使い、β_i については (3) 式によって別途カリブレートすればよい。しかし、さらに一歩進めて、Calibrate share form の生産関数を使うアプローチもある。Calibrated share form の生産関数とは、(3) 式でカリブレートされた β_i を 元の生産関数 (1) に代入したもののことである。

$$\begin{aligned} y &= f(x_1, x_2) \\ &= \left[\theta_1 \left(\frac{\bar{x}_1}{\bar{y}} \right)^{\frac{1-\sigma}{\sigma}} x_1^{\frac{\sigma-1}{\sigma}} + \theta_2 \left(\frac{\bar{x}_2}{\bar{y}} \right)^{\frac{1-\sigma}{\sigma}} x_2^{\frac{\sigma-1}{\sigma}} \right]^{\frac{\sigma}{\sigma-1}} \\ &= \bar{y} \left[\theta_1 \left(\frac{x_1}{\bar{x}_1} \right)^{\frac{\sigma-1}{\sigma}} + \theta_2 \left(\frac{x_2}{\bar{x}_2} \right)^{\frac{\sigma-1}{\sigma}} \right]^{\frac{\sigma}{\sigma-1}} \end{aligned}$$

この式からわかるように、calibrated share form では、β_i の代わりに直接ベンチマークデータが生産関数に入ってくる。このため、β_i のカリブレートを別途おこなう必要はなくなる。これが Calibrated share form の利点である。

さらに、この calibrated share form の CES 生産関数から calibrated share form の (単位) 費用関数を導出することができる。

$$c(p_1, p_2) \equiv \min\{p_1 a_1 + p_2 a_2 | f(a_1, a_2) = 1\} \tag{4}$$

$$= \bar{c} \left[\theta_1 \left(\frac{p_1}{\bar{p}_1} \right)^{1-\sigma} + \theta_2 \left(\frac{p_2}{\bar{p}_2} \right)^{1-\sigma} \right]^{\frac{1}{1-\sigma}} \tag{5}$$

ただし、$\bar{c} \equiv \sum_i \bar{p}_i\bar{x}_i / \bar{y}$ はベンチマークにおける単位費用である。

これは二投入物しかない単純な CES 生産関数であるが、より複雑な多段階の CES 関数の場合でも基本的には同じ方法が適用できる。また、生産関数ではなく、効用関数の場合にも全く同じ方法が適用できる。後に、提示する GAMS のプログラムでは全ての費用関数、価格指数を calibrated share form で記述している。また、需要関数も calibrated share form の費用関数、価格指数から導出されたものを利用している。

1.2 記号

以下、モデルで利用される記号のリストである。プログラム内での変数との対応も示してあるので、プログラムを見る際に参考にして欲しい。

アクティビティーレベル

記号	意味	プログラム内の記号
Q_{si}	生産活動のレベル	`q_q(s,i)`
A_{si}	Armington 統合のレベル	`q_a(s,i)`
X_{si}	輸出活動	`q_x(s,i)`
M_{si}	輸入活動	`q_m(s,i)`
$\bar{C}_s$	消費財統合活動	`q_cc(s)`
W_s	期間効用	`q_w(s)`
G_s	一時点の政府支出	`q_g(s)`
J_s	純投資	`q_j(s)`
L_s^S	労働供給	`q_sl(s)`
U	生涯効用	`q_u`
K_s	資本ストック	`q_k(s)`
G^L	政府の生涯支出	`q_gle`
CNC_s	炭素排出 ($i \in \mathrm{ES}$)	`q_cnc(s,e)`

単位費用

記号	意味	プログラム内の記号
c_s^{CC}	消費財統合の単位費用	`c_cc(s)`
c_{si}^A	Armington 統合の単位費用	`c_a(s,i)`
c_{si}^W	期間効用の単位費用 (支出関数)	`c_w(s)`
c^U	生涯効用の単位費用 (支出関数)	`c_u`
c_{si}^Q	生産の単位費用	`c_q(s,i)`
c_{si}^E	燃焼用途の排出源財の単位費用 ($i \in \mathrm{ES}$)	`c_e(s,e)`

価格

記号	意味	プログラム内の記号
p_{si}^Q	生産物の価格指数	`p_q(s,i)`
p_{si}^F	合成本源的要素の価格指数	`p_f(s,i)`
p_{si}^{EE}	生産における合成エネルギー財の価格指数	`p_ee(s,i)`
p_{si}^{FE}	生産における合成エネルギー・本源的要素の価格指数	`p_fe(s,i)`
p_s^C	合成非エネルギー財の価格指数	`p_c(s)`
p_s^{EC}	消費における合成エネルギー財の価格指数	`p_ec(s)`
p_s^I	投資財の価格指数	`p_i(s)`
p_{si}^E	燃焼用途の排出源財の価格 ($i \in \mathrm{ES}$)	`p_e(s,e)`
p_s^{CC}	合成消費財の価格	`p_cc(s)`
p_{si}^D	国内財の価格	`p_d(s,i)`
p_{si}^X	輸出財の価格	`p_x(s,i)`
p_{si}^M	輸入財の価格	`p_m(s,i)`
p^{FX}	外国為替の価格 (為替レート)	`p_fxv`
p_s^L	労働の価格	`p_l(s)`
p_s^{LE}	家計の直面する労働の価格	`p_le(s)`
p_s^W	期間効用の価格指数	`p_w(s)`
p^U	生涯効用の価格指数	`p_u`
p_{si}^A	Armington 財の価格指数	`p_a(s,i)`
p_s^G	政府支出の価格指数	`p_g(s)`
p_s^K	資本の影の価格	`p_k(s)`
r_s^K	レンタルプライス	`p_rk(s)`
r_s^{KE}	家計の直面するレンタルプライス	`p_rke(s)`
p^{GL}	政府の生涯支出の価格指数	`p_gle`
p_s^{CA}	排出権価格	`p_ca(s)`
p_s^{KA}	調整費用プレミアム	`p_ka(s)`

単位需要・供給

記号	意味	プログラム内の記号
a_{si}^{X}	輸出財の単位供給	`a_x(s,i)`
a_{si}^{D}	国内財の単位供給	`a_d(s,i)`
a_{si}^{L}	単位労働需要	`a_l(s,i)`
a_{si}^{K}	単位資本需要	`a_k(s,i)`
a_{sji}^{E}	燃焼用途の排出源財 (除 COL、LIM) への需要 ($j \in$ EC)	`a_e(s,ec,i)`
a_{si}^{EC}	エネルギー財への消費需要 ($i \in$ ENE)	`a_ec(s,ene)`
a_{si}^{C}	非エネルギー財への消費需要 ($i \in$ NENE)	`a_c(s,nene)`
a_{s}^{CC}	合成消費財への需要	`a_cc(s)`
a_{s}^{LE}	余暇需要	`a_lei(s)`
a_{s}^{U}	期間効用に対する需要	`a_u(s)`
a_{si}^{AD}	Armington 統合における国内財への単位需要	`a_ad(s,i)`
a_{si}^{AM}	Armington 統合における輸入財への単位需要	`a_am(s,i)`
$\bar{a}_{ij}^{\mathrm{NEN}}$	非エネルギー財 (除 COL and LIM) への中間財需要 (外生変数)	
$\bar{a}_{ij}^{\mathrm{CL}}$	燃焼用途の COL と LIM への中間財需要 (外生変数)	
$\bar{a}_{ij}^{\mathrm{NC}}$	非燃焼用途の排出源財への需要 (外生変数)	
$\bar{a}_{i}^{I}$	投資需要 (外生変数)	
$\bar{a}_{i}^{G}$	政府支出のための需要 (外生変数)	

Income

記号	意味	プログラム内の記号
Y^{H}	家計の生涯所得	`v_inc_h`
Y^{G}	政府の生涯所得	`v_inc_gl`
M_{s}^{G}	政府の一時点での所得	`v_inc_g(s)`
K_{T+1}	終端後時点における資本ストック	`q_tcap`
τ	乗数	`m_tau`

その他

記号	意味	プログラム内の記号
R_s	割引要因	`p_disc(s)`
r_s	利子率	`1/p_intr(s) - 1`

シェア変数

記号	意味	プログラム内の記号
θ_{i}^{X}	生産における輸出供給のシェア	`sh_x(i)`
θ_{i}^{FE}	本源的要素・エネルギー財の投入シェア	`sh_fe(i)`
θ_{i}^{F}	本源的要素の投入シェア	`sh_f(i)`
θ_{i}^{FL}	本源的要素における労働の投入シェア	`sh_fl(i)`
θ_{ji}^{E}	生産における各エネルギー財の投入シェア ($j \in$ ENE)	`sh_e(ene)`
$\theta_{ji}^{\mathrm{NEN}}$	生産における各非エネルギー財の投入シェア ($j \in$ NEN)	`sh_ee(j,nen)`
$\theta_{ji}^{\mathrm{NC}}$	生産における非燃焼用途の排出源財の投入シェア ($j \in$ ES)	`sh_nc(e,i)`
$\theta_{ji}^{\mathrm{CL}}$	生産における燃焼用途の COL と LIM の投入シェア ($j \in$ CL)	`sh_cl(cl,i)`
θ^{NENE}	消費財統合のおける非エネルギー財のシェア	`sh_nene`
θ_{i}^{C}	非エネルギー財統合のおける各非エネルギー財のシェア ($i \in$ NENE)	`sh_c(nene)`
θ_{i}^{EC}	エネルギー財統合のおける各エネルギー財のシェア ($j \in$ ENE)	`sh_ec(ene)`
θ^{LE}	余暇のシェア	`sh_lei`
θ_{s}^{U}	期間効用のシェア	`sh_u(s)`
θ_{i}^{G}	政府支出における各財のシェア	`sh_g(i)`
θ_{i}^{AD}	Armington 統合における国内財のシェア	`sh_ad(i)`
$\theta_{i}^{\mathrm{INV}}$	投資における各財のシェア	`sh_i(i)`

政策変数

記号	意味	プログラム内の記号
t_i^L	生産における労働税率	`tl(i)`
t_i^K	生産における資本税率	`tk(i)`
t^C	消費税率	`tc`
t^I	労働所得税率	`tli`
t^A	資本所得税率	`tci`
t_i^Q	生産に対する間接税率	`tq(i)`
t_i^M	輸入関税率	`tm(i)`
s_i^C	消費補助金率	`-ts(i)`

弾力性パラメーター

記号	意味	プログラム内の記号
η	国内供給と輸出供給の間の変形弾力性	`eta`
$\sigma_{F,i}$	資本と労働の代替弾力性	`sigpf(i)`
σ_{FE}	エネルギー・本源的要素の代替弾力性	`sigpfe`
σ_{EE}	エネルギーの間の代替の弾力性	`sigee`
$\sigma_{A,i}$	Armington 統合における国内財と輸入財の代替弾力性	`siga(i)`
σ_{LC}	効用関数内の余暇と消費の代替弾力性	`sigcl`
σ_{CC}	消費財統合におけるエネルギー財と非エネルギー財の代替弾力性	`sigcc`
σ_{EC}	エネルギー財の間の代替弾力性	`sigec`
σ_C	非エネルギー財の間の代替弾力性	`sigc`
σ_U	異時点間代替の弾力性	`sigu`

部門・財のインデックス

記号		意味	プログラム内の記号
I	=	全ての財・部門の集合	`i,ii`
ES	=	{COC, SLA, COK, CRU, PET, NAT, GAS, LIM} ··· 排出源財	`e`
EC	=	{COC, SLA, CRU, PET, NAT, GAS}	`ec`
CL	=	{COK, LIM}.	`cl`
ELE	=	{ELE} ··· 電力	`ele`
ENE	=	EC ∪ ELE ··· エネルギー財	`ene`
NENE	=	$I \setminus$ ENE ··· 非エネルギー財	`nene`
EN	=	ES ∪ ELE ··· エネルギー財 + COL + LIM	`en`
NEN	=	$I \setminus$ EN ··· 非エネルギー財 – COL – LIM	`nen`

1.3 モデル

以下、モデルを提示する。

1.3.1 単位費用

消費財統合の単位費用

$$c_s^{\mathrm{CC}} = \bar{c}_t^{\mathrm{CC}} \left[\theta^{\mathrm{NENE}} \left(\frac{p_s^C}{\bar{p}_t^C} \right)^{1-\sigma_{\mathrm{CC}}} + (1-\theta^{\mathrm{NENE}}) \left(\frac{p_s^{\mathrm{EC}}}{\bar{p}_t^{\mathrm{EC}}} \right)^{1-\sigma_{\mathrm{CC}}} \right]^{\frac{1}{1-\sigma_{\mathrm{CC}}}} \qquad \{c_s^{\mathrm{CC}}\}$$

Armington 統合の単位費用

$$c_{si}^{A} = \bar{c}_{it}^{A} \left[\theta_i^{\mathrm{AD}} \left(\frac{p_{si}^{D}}{\bar{p}_{ti}^{D}} \right)^{1-\sigma_A} + (1-\theta_i^{\mathrm{AD}}) \left(\frac{p_{si}^{M}}{\bar{p}_{ti}^{M}} \right)^{1-\sigma_A} \right]^{\frac{1}{1-\sigma_A}} \qquad \{c_{si}^{A}\}$$

期間効用の単位費用 (支出関数)

$$c_s^{W} = \bar{c}_t^{W} \left[\theta^{\mathrm{LE}} \left[\frac{p_s^{\mathrm{LE}}}{\bar{p}_t^{\mathrm{LE}}} \right]^{1-\sigma_{\mathrm{LC}}} + (1-\theta^{\mathrm{LE}}) \left[\frac{(1+t_s^{C}) p_s^{\mathrm{CC}}}{(1+\bar{t}_t^{C}) \bar{p}_t^{\mathrm{CC}}} \right]^{1-\sigma_{\mathrm{LC}}} \right]^{\frac{1}{1-\sigma_{\mathrm{LC}}}} \qquad \{c_s^{W}\}$$

生涯効用の単位費用 (支出関数)

$$c^{U} = \bar{c}^{U} \left[\sum_{s=t}^{T} \theta_s^{U} \left(\frac{p_s^{W}}{\bar{p}_t^{W} p_s^{\mathrm{RF}}} \right)^{1-\sigma_U} \right]^{\frac{1}{1-\sigma_U}} \qquad \{c^{U}\}$$

生産の単位費用

$$c_{si}^{Q} = \bar{c}_{ti}^{Q} \left[\sum_{j \in \mathrm{NEN}} \theta_{ji}^{\mathrm{NEN}} \frac{p_{sj}^{A}}{\bar{p}_{tj}^{A}} + \sum_{j \in \mathrm{ES}} \theta_{ji}^{\mathrm{NC}} \frac{p_{sj}^{A}}{\bar{p}_{tj}^{A}} + \sum_{j \in \mathrm{CL}} \theta_{ji}^{\mathrm{CL}} \frac{p_{sj}^{\mathrm{EE}}}{\bar{p}_{tj}^{\mathrm{EE}}} + \theta_i^{\mathrm{FE}} \frac{p_{si}^{\mathrm{FE}}}{\bar{p}_{ti}^{\mathrm{FE}}} \right] \qquad \{c_{si}^{Q}\}$$

1.3.2 価格 (指数)

生産物の価格指数

$$p_{si}^{Q} = \bar{p}_{ti}^{Q} \left[\theta_i^{X} \left(\frac{p_{si}^{X}}{\bar{p}_{ti}^{X}} \right)^{1+\eta} + (1-\theta_i^{X}) \left(\frac{p_{si}^{D}}{\bar{p}_{ti}^{D}} \right)^{1+\eta} \right]^{\frac{1}{1+\eta}} \qquad \{p_{si}^{Q}\}$$

合成本源的要素の価格指数

$$p_{si}^{F} = \bar{p}_{ti}^{F} \left[\theta_i^{\mathrm{FL}} \left[\frac{(1+t_i^{L}) p_s^{L}}{(1+\bar{t}_i^{L}) \bar{p}_t^{L}} \right]^{1-\sigma_{F,i}} + (1-\theta_i^{\mathrm{FL}}) \left[\frac{(1+t_i^{K}) r_s^{K}}{(1+\bar{t}_i^{L}) \bar{r}_t^{K}} \right]^{1-\sigma_{F,i}} \right]^{\frac{1}{1-\sigma_{F,i}}} \qquad \{p_{si}^{F}\}$$

生産における合成エネルギー財の価格指数

$$p_{si}^{\mathrm{EE}} = \bar{p}_{ti}^{\mathrm{EE}} \left[\sum_{j \in \mathrm{EC}} \theta_{ji}^{\mathrm{EE}} \left(\frac{p_{sj}^{E}}{\bar{p}_{tj}^{E}} \right)^{1-\sigma_{\mathrm{EE}}} + \theta_{\mathrm{ELE},i}^{\mathrm{EE}} \left(\frac{p_{s,\mathrm{ELE}}^{A}}{\bar{p}_{t,\mathrm{ELE}}^{A}} \right)^{1-\sigma_{\mathrm{EE}}} \right]^{\frac{1}{1-\sigma_{\mathrm{EE}}}} \qquad \{p_{si}^{\mathrm{EE}}\}$$

生産における合成エネルギー・本源的要素の価格指数

$$p_{si}^{\mathrm{FE}} = \bar{p}_{ti}^{\mathrm{FE}} \left[\theta_i^{F} \left(\frac{p_{si}^{F}}{\bar{p}_{ti}^{F}} \right)^{1-\sigma_{\mathrm{FE}}} + (1-\theta_i^{F}) \left(\frac{p_{si}^{\mathrm{EE}}}{\bar{p}_{ti}^{\mathrm{EE}}} \right)^{1-\sigma_{\mathrm{FE}}} \right]^{\frac{1}{1-\sigma_{\mathrm{FE}}}} \qquad \{p_{si}^{\mathrm{FE}}\}$$

合成非エネルギー消費財の価格指数

$$p_s^C = \bar{p}_t^C \left[\sum_{i \in \mathrm{NENE}} \theta_i^C \left[\frac{(1-s_i^C)p_{si}^A}{(1-\bar{s}_i^C)\bar{p}_{ti}^A} \right]^{1-\sigma_C} \right]^{\frac{1}{1-\sigma_C}} \qquad \{p_s^C\}$$

合成エネルギー消費財の価格指数

$$p_s^{\mathrm{EC}} = \bar{p}_t^{\mathrm{EC}} \left[\sum_{i \in \mathrm{EC}} \theta_i^{\mathrm{EC}} \left[\frac{(1-s_i^C)p_{si}^E}{(1-\bar{s}_i^C)\bar{p}_{ti}^E} \right]^{1-\sigma_{\mathrm{EC}}} + \theta_{\mathrm{ELE}}^{\mathrm{EC}} \left[\frac{(1-s_{\mathrm{ELE}}^C)p_{s,\mathrm{ELE}}^A}{(1-\bar{s}_{\mathrm{ELE}}^C)\bar{p}_{t,\mathrm{ELE}}^A} \right]^{1-\sigma_{\mathrm{EC}}} \right]^{\frac{1}{1-\sigma_{\mathrm{EC}}}} \qquad \{p_s^{\mathrm{EC}}\}$$

燃焼用途の排出源財の単位費用

$$c_{si}^E = p_{si}^A + \gamma_i p_s^{\mathrm{CA}} \qquad \{c_{si}^E\}$$

投資財の価格指数

$$p_s^I = \bar{p}_t^I \left[\sum_i \theta_i^{\mathrm{INV}} \frac{p_{si}^A}{\bar{p}_{ti}^A} \right] \qquad \{p_s^I\}$$

調整費用プレミアム

$$p_s^{\mathrm{KA}} = p_s^I \frac{\phi}{2} \left[\frac{J_s}{K_s} \right]^2 \qquad \{p_s^{\mathrm{KA}}\}$$

家計の直面する賃金

$$p_s^{\mathrm{LE}} = (1-t^L)p_s^L \qquad \{p_s^{\mathrm{LE}}\}$$

家計の直面するレンタルプライス

$$r_s^{\mathrm{KE}} = (1-t^A)r_s^K \qquad \{r_s^{\mathrm{KE}}\}$$

割引き要因

$$R_s = \begin{cases} 1 & s = t \\ \prod_{l=t}^{s}(1+r_s)^{-1} & s > t \end{cases} \qquad \{R_s\}$$

1.3.3 ゼロ利潤条件

生産活動

$$\frac{c_{si}^Q}{c_{si}^Q} \geq \frac{(1-t_i^Q)p_{si}^Q}{(1-t_i^Q)p_{si}^Q} \qquad \{Q_{si}\}$$

Armington 統合

$$\frac{c_{si}^{A}}{c_{si}^{A}} \geq \frac{p_{si}^{A}}{p_{si}^{A}} \qquad \{A_{si}\}$$

輸出活動

$$p_{si}^{X} \geq p^{\mathrm{FX}} \qquad \{X_{si}\}$$

輸入活動

$$(1+t_i^{M})p^{\mathrm{FX}} \geq p_{si}^{M} \qquad \{M_{si}\}$$

消費財統合

$$\frac{c_s^{\mathrm{CC}}}{c_s^{\mathrm{CC}}} \geq \frac{p_s^{\mathrm{CC}}}{p_s^{\mathrm{CC}}} \qquad \{\bar{C}_s\}$$

期間効用

$$R_s c_s^{W} \geq p_s^{W} \qquad \{W_s\}$$

生涯効用

$$\frac{c^{U}}{c^{U}} \geq \frac{p^{U}}{p^{U}} \qquad \{U\}$$

一時点における政府支出

$$\bar{p}_t^{G}\left[\sum_i \theta_i^{G}\frac{p_{si}^{A}}{\bar{p}_{ti}^{A}}\right] \geq p_s^{G} \qquad \{G_s\}$$

投資活動

$$\frac{p_s^{I}}{\bar{p}_s^{I}}\left[1+\phi\left(\frac{J_s}{K_s}\right)\right] \geq p_s^{K} \qquad \{J_s\}$$

資本蓄積

$$(1+r_s)p_{s-1}^{K} \geq r_s^{\mathrm{KE}} + (1-\delta)p_s^{K} + p_s^{\mathrm{KA}} \qquad \{K_s\}$$

生涯政府支出

$$\bar{p}^{\mathrm{GL}}\left[\sum_{s=t}^{T}\theta_s^{G}\frac{p_s^{G}}{\bar{p}_t^{G}p_s^{\mathrm{RF}}}\right] \geq p^{\mathrm{GL}} \qquad \{G^{L}\}$$

炭素排出

$$c_{si}^{E} \geq p_{si}^{E} \qquad \{\mathrm{CNC}_i\}_{i\in\mathrm{ES}}$$

1.3.4 単位需要・供給関数

輸出供給，国内供給

$$a^X_{si} = \bar{a}^X_{ti}\left[\frac{p^X_{si}/\bar{p}^X_{ti}}{p^Q_{si}/\bar{p}^Q_{ti}}\right]^{\eta} \qquad \{a^X_{si}\}$$

$$a^D_{si} = \bar{a}^D_{ti}\left[\frac{p^D_{si}/\bar{p}^D_{ti}}{p^Q_{si}/\bar{p}^Q_{ti}}\right]^{\eta} \qquad \{a^D_{si}\}$$

労働需要

$$a^L_{si} = \bar{a}^L_{ti}\left[\frac{p^F_{si}/\bar{p}^F_{ti}}{(1+t^L_i)p^L_s/[(1+\bar{t}^L_i)\bar{p}^L_t]}\right]^{\sigma_{\mathrm{FL}}}\left[\frac{p^{\mathrm{FE}}_{si}/\bar{p}^{\mathrm{FE}}_{ti}}{p^F_{si}/\bar{p}^F_{ti}}\right]^{\sigma_{\mathrm{FE}}} \qquad \{a^L_{si}\}$$

資本需要

$$a^K_{si} = \bar{a}^K_{ti}\left[\frac{p^F_{si}/\bar{p}^F_{ti}}{(1+t^K_i)r^K_s/[(1+\bar{t}^K_i)\bar{r}^K_t]}\right]^{\sigma_{\mathrm{FL}}}\left[\frac{p^{\mathrm{FE}}_{si}/\bar{p}^{\mathrm{FE}}_{ti}}{p^F_{si}/\bar{p}^F_{ti}}\right]^{\sigma_{\mathrm{FE}}} \qquad \{a^K_{si}\}$$

燃焼用途の排出源財への需要 (除 COL、LIM)

$$a^E_{sij} = \bar{a}^E_{tij}\left[\frac{p^{\mathrm{EE}}_{sj}/\bar{p}^{\mathrm{EE}}_{tj}}{p^E_{si}/\bar{p}^E_{ti}}\right]^{\sigma_{\mathrm{EE}}}\left[\frac{p^{\mathrm{FE}}_{sj}/\bar{p}^{\mathrm{FE}}_{tj}}{p^{\mathrm{EE}}_{sj}/\bar{p}^{\mathrm{EE}}_{tj}}\right]^{\sigma_{\mathrm{FE}}} \qquad \{a^E_{sij}\}_{i\in\mathrm{EC}}$$

生産における電力需要

$$a^E_{sij} = \bar{a}^E_{tij}\left[\frac{p^{\mathrm{EE}}_{sj}/\bar{p}^{\mathrm{EE}}_{tj}}{p^A_{si}/\bar{p}^A_{ti}}\right]^{\sigma_{\mathrm{EE}}}\left[\frac{p^{\mathrm{FE}}_{sj}/\bar{p}^{\mathrm{FE}}_{tj}}{p^{\mathrm{EE}}_{sj}/\bar{p}^{\mathrm{EE}}_{tj}}\right]^{\sigma_{\mathrm{FE}}} \qquad \{a^E_{si}\}_{i\in\mathrm{ELE}}$$

排出源財 (除 LIM、COK) への消費需要

$$a^{\mathrm{EC}}_{si} = \bar{a}^{\mathrm{EC}}_{ti}\left[\frac{p^{\mathrm{EC}}_{s}/\bar{p}^{\mathrm{EC}}_{t}}{(1-s^C_i)p^E_{si}/[(1-\bar{s}^C_i)\bar{p}^E_{ti}]}\right]^{\sigma_{\mathrm{EC}}}\left[\frac{p^{\mathrm{CC}}_{s}/\bar{p}^{\mathrm{CC}}_{t}}{p^{\mathrm{EC}}_{s}/\bar{p}^{\mathrm{EC}}_{t}}\right]^{\sigma_{\mathrm{CC}}} \qquad \{a^{\mathrm{EC}}_{si}\}_{i\in\mathrm{EC}}$$

電力への消費需要

$$a^{\mathrm{EC}}_{si} = \bar{a}^{\mathrm{EC}}_{ti}\left[\frac{p^{\mathrm{EC}}_{s}/\bar{p}^{\mathrm{EC}}_{t}}{(1-s^C_i)p^A_{si}/[(1-\bar{s}^C_i)\bar{p}^A_{ti}]}\right]^{\sigma_{\mathrm{EC}}}\left[\frac{p^{\mathrm{CC}}_{s}/\bar{p}^{\mathrm{CC}}_{t}}{p^{\mathrm{EC}}_{s}/\bar{p}^{\mathrm{EC}}_{t}}\right]^{\sigma_{\mathrm{CC}}} \qquad \{a^{\mathrm{EC}}_{si}\}_{i\in\mathrm{ELE}}$$

非排出源財 (除電力)) への消費需要

$$a^C_{si} = \bar{a}^C_{ti}\left[\frac{p^C_{s}/\bar{p}^C_{t}}{(1-s^C_i)p^A_{si}/[(1-\bar{s}^C_i)\bar{p}^A_{ti}]}\right]^{\sigma_C}\left[\frac{p^{\mathrm{CC}}_{s}/\bar{p}^{\mathrm{CC}}_{t}}{p^C_{s}/\bar{p}^C_{t}}\right]^{\sigma_{\mathrm{CC}}} \qquad \{a^C_{si}\}_{i\in\mathrm{NEN}}$$

COK と LIM への消費需要

$$a^C_{si} = \bar{a}^C_{ti}\left[\frac{p^C_{s}/\bar{p}^C_{t}}{(1-s^C_i)p^E_{si}/[(1-\bar{s}^C_i)\bar{p}^E_{ti}]}\right]^{\sigma_C}\left[\frac{p^{\mathrm{CC}}_{s}/\bar{p}^{\mathrm{CC}}_{t}}{p^C_{s}/\bar{p}^C_{t}}\right]^{\sigma_{\mathrm{CC}}} \qquad \{a^C_{si}\}_{i\in\mathrm{CL}}$$

合成消費への需要

$$a_s^{\mathrm{CC}} = \bar{a}_t^{\mathrm{CC}} \left[\frac{p_s^W / \bar{p}_t^W}{(1+t^C) p_s^{\mathrm{CC}} / [(1+\bar{t}^C) \bar{p}_t^{\mathrm{CC}}]} \right]^{\sigma_{\mathrm{LC}}} \qquad \{a_s^{\mathrm{CC}}\}$$

余暇需要

$$a_s^{\mathrm{LE}} = \bar{a}_t^{\mathrm{LE}} \left[\frac{p_s^W / \bar{p}_t^W}{p_s^{\mathrm{LE}} / \bar{p}_t^{\mathrm{LE}}} \right]^{\sigma_W} \qquad \{a_s^{\mathrm{LE}}\}$$

期間効用への需要

$$a_s^U = \bar{a}_s^U \left[\frac{c^U / \bar{c}^U}{p_s^W / (\bar{p}_t^W p_s^{\mathrm{RF}})} \right]^{\sigma_U} \qquad \{a_s^U\}$$

国内財への需要，輸入財への需要

$$a_{si}^{\mathrm{AD}} = \bar{a}_{ti}^{\mathrm{AD}} \left[\frac{c_{si}^A / \bar{c}_{ti}^A}{p_{si}^D / \bar{p}_{ti}^D} \right]^{\sigma_A} \qquad \{a_{si}^{\mathrm{AD}}\}$$

$$a_{si}^{\mathrm{AM}} = \bar{a}_{ti}^{\mathrm{AM}} \left[\frac{c_{si}^A / \bar{c}_{ti}^A}{p_{si}^M / \bar{p}_{ti}^M} \right]^{\sigma_A} \qquad \{a_{si}^{\mathrm{AM}}\}$$

労働供給

$$L_s^S = \bar{L}_s - a_s^{\mathrm{LE}} W_s \qquad \{L_s^S\}$$

1.3.5 市場均衡

合成消費市場

$$\bar{C}_s \geq a_s^{\mathrm{CC}} W_s \qquad \{p_s^{\mathrm{CC}}\}$$

国内財市場

$$a_{si}^D Q_{si} \geq a_{si}^{\mathrm{AD}} A_{si} \qquad \{p_{si}^D\}$$

輸出財市場

$$a_{si}^X Q_{si} \geq X_{si} \qquad \{p_{si}^X\}$$

輸入財市場

$$M_{si} \geq a_{si}^{\mathrm{AM}} A_{si} \qquad \{p_{si}^M\}$$

経常収支の均衡 (外国為替市場)

$$\sum_{s=t}^{T}\sum_{i} p_s^{\mathrm{RF}} X_{si} \geq \sum_{s=t}^{T}\sum_{i} p_s^{\mathrm{RF}} M_{si} \qquad \{p^{\mathrm{FX}}\}$$

労働市場

$$L_s^S \geq \sum_i a_{si}^L Q_{si} \qquad \{p_s^L\}$$

労働供給

$$\bar{L}_s \geq L_s^S + a_s^{\mathrm{LE}} W_s \qquad \{p_s^{\mathrm{LE}}\}$$

期間効用

$$W_s \geq a_s^U U \qquad \{p_s^W\}$$

生涯効用

$$M^H \geq p^U U \qquad \{p^U\}$$

Armington 財

非エネルギー財 (除 COK と LIM)。

$$A_{si} \geq a_{si}^C \bar{C}_s + \sum_j \bar{a}_{ij}^Q Q_{sj} + \bar{a}_i^I I_s + \bar{a}_i^G G_s \qquad \{p_{si}^A\}_{i\in \mathrm{NENE}}$$

LIM と COK の市場。

$$A_{si} \geq a_{si}^C \bar{C}_s + \sum_j \bar{a}_{ij}^{\mathrm{CL}} Q_{sj} + \sum_j \bar{a}_{ij}^{\mathrm{NC}} Q_{sj} + \bar{a}_i^I I_s + \bar{a}_i^G G_s \qquad \{p_{si}^A\}_{i\in \mathrm{CL}}$$

電力市場

$$A_{si} \geq a_{si}^{\mathrm{EC}} \bar{C}_s + \sum_j a_{sij}^E Q_{sj} + \bar{a}_i^I I_s + \bar{a}_i^G G_s \qquad \{p_{si}^A\}_{i\in \mathrm{ELE}}$$

排出源財 (除 LIM and COK)。

$$A_{si} \geq a_{si}^{\mathrm{EC}} \bar{C}_s + \sum_j a_{sij}^E Q_{sj} + \sum_j \bar{a}_{ij}^{\mathrm{NC}} Q_{sj} + \bar{a}_i^I I_s + \bar{a}_i^G G_s \qquad \{p_{si}^A\}_{i\in \mathrm{EC}}$$

投資財

$$J_{s-1} + (1-\delta) K_{s-1} \geq K_s \qquad \{p_s^K\}_{s>t}$$

レンタル資本市場

$$K_s^S \geq \sum_i a_{ti}^K Q_{si} \qquad \{r_s^K\}$$

一時点での政府支出

$$G_s \geq \bar{a}_s^{\mathrm{GL}} G^L \qquad \{p^{\mathrm{GL}}\}$$

政府の生涯支出

$$Y^G \geq p^{\mathrm{GL}} G^L \qquad \{p^{\mathrm{GL}}\}$$

排出権市場

$$\mathrm{CA}_s \geq \sum_{i \in E} \gamma_i \mathrm{CNC}_{si} \qquad \{p_s^{\mathrm{CA}}\}$$

排出源財の価格

$$\begin{aligned} \mathrm{CNC}_{si} &\geq a_{si}^C \bar{C}_s + \sum_j \bar{a}_{ij}^{\mathrm{CL}} Q_{sj} \qquad \{p_{si}^E\}_{i \in \mathrm{CL}} \\ \mathrm{CNC}_{si} &\geq a_{si}^{\mathrm{EC}} \bar{C}_s + \sum_j a_{sij}^E Q_{sj} \qquad \{p_{si}^E\}_{i \in \mathrm{EC}} \end{aligned}$$

1.3.6 所得制約

家計の生涯所得

$$\begin{aligned} Y^H = &\left[(1-t^A) r_t^K + (1-\delta) p_t^K + p_t^{\mathrm{KA}}\right] K_t - R_T p_T^K K_{T+1} \\ &+ \sum_{s=t}^{T} R_s \left[(1-t^I) p_s^L \bar{L}_s + p_s^G T_s\right] \qquad \{Y^H\} \end{aligned}$$

政府の一時点での収入

$$\begin{aligned} M_s^G = &\sum_i \left[t_i^L p_s^L a_{si}^L + t_i^K r_s^K a_{si}^K\right] Q_{si} + \sum_i t_i^Q p_{si}^Q Q_{si} \\ &+ t^C p_s^{\mathrm{CC}} a_s^{\mathrm{CC}} W_s + t^I p_s^L L_s^S + t^A r_s^K K_s \\ &+ \sum_i t_i^M p^{\mathrm{FX}} p_s^{\mathrm{RF}} M_{si} - \sum_i s_i^C p_{si}^A a_{si}^{\mathrm{CC}} \mathrm{CC}_s - p_s^G T_s \qquad \{M_s^G\} \end{aligned}$$

政府の生涯所得

$$Y^G = \sum_{s=t}^{T} R_s Y_s^G \qquad \{Y_s^G\}$$

終端条件

$$\frac{W_T}{W_{T-1}} = \frac{J_T}{J_{T-1}} \qquad \{K_{T+1}\}$$

2 プログラム

以下は、シミュレーションをおこなうための GAMS のプログラムである。

```
$title  Dynamic model for double dividend analysis.
display "com: Dynamic model for double dividend analysis.";
$ontext
$Id: dynamic_mcp.gms,v 1.8 2004/03/15 12:29:56 st Exp $
Copyright (C)   2004 Shiro Takeda
Time-stamp:     <2004-05-15 18:31:22 Shiro Takeda>
Author:         Shiro Takeda <zbc08106@park.zero.ad.jp>
e-mail:         <zbc08106@park.zero.ad.jp>
First-written:  <2003/04/24>

$offtext
*       ---------------------------------------------------------------------
*       Important parameters.

*       The number of periods.
$if not setglobal tyear $setglobal tyear 100

*       Emission reduction rate (%).
$if not setglobal rdrate $setglobal rdrate 100

*       Trade balance.  Non-zero means balanced trade at each period.
$if not setglobal bopcon $setglobal bopcon 0

*       Government budget balance.  Non-zero means balanced budget at each
*       period.
$if not setglobal gbb $setglobal gbb 0

*       Quadratic adjustment cost.
$if not setglobal phi $setglobal phi 0.5

*       Labor growth rate
$if not setglobal gr $setglobal gr 0.01

*       Solver choice for calibration.
$if not setglobal mcp $setglobal mcp 1

*       Period for plotting.
$if not setglobal syear $setglobal syear 50

*       Suffix for plotting
$if not setglobal suffix $setglobal suffix %rdrate%

*       Solver choice
option mcp = path;
* option mcp = miles;

*       End-of-line comment symbol.
$eolcom !
*       Inline comment symbol.
$inlinecom { }

*       Digit.
option decimals = 8;
```

```
option sysout = on;

*       Define parameters.
parameter
    sol_mcp     Flag for quadratic programming
    rdrate      Reduction rate;
sol_mcp = %mcp%;
rdrate = %rdrate%;
display sol_mcp, rdrate;

*       ---------------------------------------------------------------
*       Define elasticity parameters.
display "com: Define elasticity parameters";

*       NB: EOS means elasticity of substitution, EOT means elasticity of
*       transformation.
parameter
    eta     EOT: export and domestic supply                 / 4 /
    sigpfe  EOS: Primary factor and energy                  / 0.5 /
    sigpf   EOS: Primary factor
    sigee   EOS: energy input                               / 0.5 /
    siga    Armington elasticity
    sigu    EOS: Intertemporal elasticity                   / 0.5 /
    sigcl   EOS: consumption and leisure
    sigcc   EOS: energy vs non-energy consumption           / 0.3 /
    sigc    EOS: non-energy consumption goods               / 1 /
    sigec   EOS: energy consumption goods                   / 2 /
    eols    Elasticity of labor supply                      / 0.15 /
;
display eta, sigpfe, sigee, sigu, sigcc, sigc, sigec;

$ontext
The values of Armington elasticity (elasticity of substitution between
domestic good and import) and elasticity of substitution between primary
factors are taken from GTAP version 5 data.
$offtext

display "com: Defining sets and parameters for dynamic model.";
set
    t           Period index            / 1*%tyear% /
    tfirst(t)   First period
    tlast(t)    Last period
    tsub(t)     Period for plotting     / 1*%syear% /
;
tfirst(t) = yes$(ord(t) eq 1);
tlast(t) =  yes$(ord(t) eq card(t));
display t, tfirst, tlast;
alias (t,tt);

*       ---------------------------------------------------------------
*       Import set definitions.
display "com: Import set definitions.";
$include set_definition

*       ---------------------------------------------------------------
*       Import benchmark data.
display "com: Import benchmark data.";
$include dataset.dat
```

```
*       Endogenous sector.
display "Endogenous sector";
display iov, iovc, iovnc, ioqc, ioqnc;

*       Value added sector.
display "Value added sector";
display nlabinc, ncapinc, labtax, captax, nitax;

*       Final demand sector.
display "Final demand sector";
display cons, gexp, inve, expo, impo, impt, ctax, csub;

*       Household data.
display "Household data";
display lincome, cincome, linctax, cinctax, hctax, hcsub, hsave, hntrans;

*       ---------------------------------------------------------------
*       Check imported data.
display "com: Check imported data.";

parameter
    ioq         Quantity data.
    chk_fdva    Check FD and VA data
    chk_out     Check output data
    chk_out_p   -1 if no output;

*       Quantity data.
ioq(e,i) = ioqc(e,i) + ioqnc(e,i);
display ioq;

*       Final demand and value added.
chk_fdva(i,"va") = nlabinc(i) + labtax(i) + ncapinc(i) + captax(i)
    + nitax(i);
chk_fdva("sum","va") = round(sum(i, chk_fdva(i,"va")), 6);
chk_fdva(i,"fd")
    = cons(i) + csub(i) + gexp(i) + inve(i)
    + expo(i) + impo(i) + impt(i);
chk_fdva("sum","fd") = round(sum(i, chk_fdva(i,"fd")), 6);

chk_fdva("diff","fd-va")
    = round(chk_fdva("sum","fd")
    - chk_fdva("sum","va"), 6);
display chk_fdva;
abort$chk_fdva("diff","fd-va") "FD and VA are inconsistent.";

*       Ouput data.
chk_out(i,"sec")
    = sum(ii, iov(ii,i))
    + nlabinc(i) + labtax(i) + ncapinc(i) + captax(i)
    + nitax(i);
chk_out(i,"com")
    = sum(ii, iov(i,ii))
    + cons(i) + csub(i) + gexp(i) + inve(i)
    + expo(i) + impo(i) + impt(i);
chk_out("sum","sec") = sum(i, chk_out(i,"sec"));
chk_out("sum","com") = sum(i, chk_out(i,"com"));
chk_out(i,"sec_com") = round(chk_out(i,"sec") - chk_out(i,"com"), 6);
chk_out("sum","sec_com")
    = round(chk_out("sum","sec")
```

```
    - chk_out("sum","com"), 6);
display chk_out;
abort$chk_out("sum","sec_com") "Output data are inconsistent.";

chk_out(i,"sec") = round(chk_out(i,"sec"), 6);
chk_out(i,"com") = round(chk_out(i,"com"), 6);
chk_out_p(i,"sec") = -1;
chk_out_p(i,"com") = -1;
chk_out_p(i,"sec")$chk_out(i,"sec") = 1;
chk_out_p(i,"com")$chk_out(i,"com") = 1;
option chk_out_p:0;
display chk_out_p;

$ontext
The values of Armington elasticity (elasticity of substitution between
domestic good and import) and elasticity of substitution between value added
are taken from GTAP version 5 data.
$offtext
siga("agr") = 2.2;
siga("omi") = 2.8;
siga("lim") = 2.8;
siga("coc") = 2.8;
siga("sla") = 2.8;
siga("cru") = 2.8;
siga("nat") = 2.8;
siga("foo") = 2.2;
siga("tet") = 2.2;
siga("ppp") = 1.8;
siga("chm") = 1.9;
siga("pet") = 1.9;
siga("opp") = 1.9;
siga("cok") = 1.9;
siga("csc") = 1.9;
siga("iam") = 2.8;
siga("mac") = 2.8;
siga("oip") = 2.8;
siga("con") = 1.9;
siga("ele") = 2.8;
siga("gas") = 2.8;
siga("sww") = 2.8;
siga("com") = 1.9;
siga("res") = 1.9;
siga("tcb") = 1.9;
siga("pub") = 1.9;
siga("ser") = 1.9;
display siga;

sigpf("agr") = 0.237;
sigpf("omi") = 0.2;
sigpf("lim") = 0.2;
sigpf("coc") = 0.2;
sigpf("sla") = 0.2;
sigpf("cru") = 0.2;
sigpf("nat") = 0.2;
sigpf("foo") = 0.237;
sigpf("tet") = 1.26;
sigpf("ppp") = 1.26;
sigpf("chm") = 1.26;
sigpf("pet") = 1.26;
```

```
sigpf("opp") = 1.26;
sigpf("cok") = 1.26;
sigpf("csc") = 1.26;
sigpf("iam") = 1.26;
sigpf("mac") = 1.26;
sigpf("oip") = 1.26;
sigpf("con") = 1.4;
sigpf("ele") = 1.26;
sigpf("gas") = 1.26;
sigpf("sww") = 1.26;
sigpf("com") = 1.68;
sigpf("res") = 1.26;
sigpf("tcb") = 1.26;
sigpf("pub") = 1.26;
sigpf("ser") = 1.26;
display sigpf;

*       ---------------------------------------------------------------
*       Production side
display "com: Production side";
parameter
    int0(i,ii)          Intermediate inputs
    vld0(i)             Value of labor inputs (net of tax)
    vkd0(i)             Value of capital inputs (net of tax)
    vpf0(i)             Value of primary factor (value added)
    ve0(i)              Value of energy inputs
    vpfe0(i)            Value of primary factor and energy inputs

    outs0(i)            Sector output (gross of tax)
    outc0(i)            Commodity output (gross of tax)

    d0(i)               Value of domestic supply
    x0(i)               Value of export supply
    out0(i)             Value of output (gross of tax)

    tq0(i)              Net indirect tax rate on production
    tl0(i)              Labor tax rate on production
    tk0(i)              Capital tax rate on production
;
*       Intermediate inputs.
int0(i,ii) = iov(i,ii);

*       Primary factor input (value).
vld0(i) = nlabinc(i);
vkd0(i) = ncapinc(i);

*       Sector output.
outs0(i) = sum(ii, iov(ii,i))
    + nlabinc(i) + ncapinc(i)
    + labtax(i) + captax(i) + nitax(i);
outs0(i) = outs0(i);

*       Commodity output.
outc0(i) = sum(ii, iov(i,ii))
    + cons(i) + csub(i) + gexp(i)
    + inve(i) + expo(i) + impo(i) + impt(i);
outc0(i) = outc0(i);

*       Output to export market.
```

```
x0(i) = expo(i);

*       Output to domestic market.
d0(i) = round(outc0(i) - x0(i), 8);

*       Total ouput.
out0(i) = d0(i) + x0(i);

display int0, vld0, vkd0, outs0, outc0, x0, d0, out0;

chk_out_p(i,"sec") = -1;
chk_out_p(i,"com") = -1;
chk_out_p(i,"sec")$outs0(i) = 1;
chk_out_p(i,"com")$outc0(i) = 1;
display chk_out_p;

*       Gross value added.
vpf0(i)$outs0(i) = ncapinc(i) + captax(i) + nlabinc(i) + labtax(i);
display vpf0;

*       Labor tax rate.
tl0(i)$vld0(i) = labtax(i) / vld0(i);

*       Capital tax rate.
tk0(i)$vkd0(i) = captax(i) / vkd0(i);

*       Indirect tax rate.
tq0(i)$outs0(i) = nitax(i) / outs0(i);
display tl0, tk0, tq0;

*       ---------------------------------------------------------------------
*       Investment
display "com: Investment";
parameter
    i0          Total value of gross investment
    ishr(i)     Share of each good in investment
    id0(i)      Gross investment demand;

id0(i) = inve(i);
i0 = sum(i, id0(i));
ishr(i) = id0(i)/i0;
display id0, i0, ishr;

*       ---------------------------------------------------------------------
*       Government demand
display "com: Government demand";
parameter
    gd0(i)      Government demand (including gov investment)
    g0          Government total expenditure;

gd0(i) = gexp(i);
g0 = sum(i, gd0(i));
display gd0, g0;

*       ---------------------------------------------------------------------
*       Trade and Armington.
display "com: Trade and Armington";
parameter
    a0(i)       Armington supply
```

```
    m0(i)       Demand for imported good (net of import tax)
    tm0(i)      Import tax rate
    bop0        Balance of payments of each good (ex - im)
    bopt0       Balance of payments (ex - im);

m0(i) = - impo(i);
tm0(i)$m0(i) = - impt(i)/m0(i);
a0(i) = d0(i) + (1+tm0(i)) * m0(i);
bop0(i) = x0(i) - m0(i);
bopt0 = round(sum(i, bop0(i)), 8);
display m0, tm0, a0, bop0, bopt0;

*       ---------------------------------------------------------------
*       Household data
display "com: Household data";

parameter
    epi0        Expanded disposal income
    wg0         Household gross current expenditure
    wn0         Household net current expenditure
    c0          Household consumption expenditure
    ce0         Energy good consumption
    cne0        Non-energy good consumption
    cg0         Household gross consumption expenditure
    cd0         Household consumption demand

    li0         Value of labor income (gross of tax)
    ki0         Value of capital income (gross of tax)

    tli0        Labor income tax rate
    tci0        Capital income tax rate
    ts0         Consumption subsidy rate
    tc0         Consumption tax rate

    vos         Value of savings
    vot         Value of net transfer
;
li0 = lincome;
ki0 = cincome;
vos = hsave;
vot = hntrans;
cd0(i) = cons(i) + csub(i);
display li0, ki0, vos, cd0, vot;

ts0(i)$cd0(i) = - csub(i) / cd0(i);
tc0(i)$cons(i) = ctax(i) / cons(i);
c0 = sum(i, cons(i));
cg0 = sum(i, (1+tc0(i)) * (1+ts0(i)) * cd0(i));
tli0 = linctax / li0;
tci0 = cinctax / ki0;
display ts0, tc0, cg0, tli0, tci0;

cne0 = sum(nene, cons(nene));
ce0 = sum(ene, cons(ene));
display cne0, ce0;

*       ---------------------------------------------------------------
*       Total labor endowment.
display "com: Total labor endowment.";
```

```
$ontext
Total labor endowment = a * labor supply
Leisure = total labor endowment - labor supply

Assumptions:

* Labor hours per month are 190 hours.
* Time available for leisure and work is 16 hours per day.

Then, total time available for leisure and work per month is given by

        16 * 30 = 480

Thus, parameter a is

        a = 480 / 190 = 2.53

$offtext
parameter
    lratio      Labor supply ratio              / 2.53 /
    endl        Benchmark total labor endowment
    lei0        Benchmark leisure;
endl = lratio * li0;
lei0 = endl - li0;
display lratio, endl, lei0;

*       ---------------------------------------------------------------
*       Defining sets and parameters for dynamic model.
parameter
    bopcon              Balance of payments restriction / %bopcon% /
    capflo              Free capital flow
    phi                 Adjustment speed                / %phi% /
    gbb                 Government balanced budget      / %gbb% /;
*       Capital flow constraint: If balance of payment restriction is not
*       imposed, capital flow is free.
bopcon = 1$%bopcon%;
capflo = yes$(not bopcon);
display bopcon, capflo, phi;

*       ---------------------------------------------------------------
*       Calibrate base year variables.
$ontext
Calibrate benchmark and baseline variables to make the baseline equilibrium
the steady state.
$offtext
parameter
    vks0        Benchmark total value of capital stock
    vai0        Benchamrk value of capital income (net of tax)
    vat0        Benchmark capital income tax;
vks0 = sum(i, vkd0(i));
vat0 = cinctax;
vai0 = cincome - vat0;
display vks0, vat0, vai0;

*       Exogenous parameter.
parameter
    gr          Baseline labor growth rate      / %gr% /;
display gr;
```

```
*       Calibrated parameters
parameter
    r           Baseline interest rate
    delta       Depreciation rate
    j0          Benchmark net investment
    jd0         Benchmark net investment demand

    rk0         Benchmark rental price for producer
    rke0        Benchmark rental price for household
    ks0         Benchmark capital stock
    pk0         Benchmark capital price
    pka0        Reduced cost of investment (adjustment premium)

    ld0         Benchmark labor input (quantity)
    kd0         Benchmark capital input (quantity);

*       Target values for calibrated variables.
parameter
    rt0         Interest rate (target level)        / 0.03 /
    delt0       Depreciation rate (target level)    / 0.07 /;
display rt0, delt0;

*       Variable definitions.
variables
    rt          Interest rate
    delt        Depreciation rate
    rket        Rental rate for household
    kt          Capital stock

    jt          Net investment
    loss        Loss function

    lam_1       Multiplier
    lam_2       Multiplier
    lam_3       Multiplier;

*       Equation definitions.
equations
    eq_obj      Objetive function
    eq_con1     Constraint 1
    eq_con2     Constraint 2
    eq_con3     Constraint 3
    eq_con4     Constraint 4

    eq_rt       FOC
    eq_delt     FOC
    eq_rket     FOC
    eq_kt       FOC;
*       ---------------------------------------------------------------
*       NLP model.
eq_obj .. loss =e=
          power((rt - rt0)/rt0, 2) + power((delt - delt0)/delt0, 2);

eq_con1 .. rket =e= (1+phi*(gr+delt))*(rt+delt)
           - phi*(delt+gr)**2 / 2;

eq_con2 .. rket*kt =e= vai0;

eq_con3 .. i0 =e= (delt+gr) * kt * (1+phi*(delt+gr)/2);
```

```
model calib_nlp NLP model / eq_obj, eq_con1, eq_con2, eq_con3 /;

*       ---------------------------------------------------------------
*       MCP model.
eq_rt .. 2*(rt-rt0)/(rt0**2) - lam_1*(1+phi*(gr+delt)) =e= 0;

eq_delt .. 2*(delt-delt0)/(delt0**2)
           + lam_1*( phi*(gr-rt) - (1+phi*(gr+delt)) )
           - lam_3*kt*(1+phi*(delt+gr)) =e= 0;

eq_rket .. lam_1 + lam_2*kt =e= 0;

eq_kt .. lam_2*rket - lam_3*(delt+gr)*(1 + phi*(delt+gr)/2) =e= 0;

model calib_mcp MCP model / eq_con1, eq_con2, eq_con3, eq_rt, eq_delt,
                            eq_rket, eq_kt /;

*       Initial values.
rt.l = rt0;
delt.l = delt0;
kt.l = vks0 / (delt.l + rt.l);
rket.l = rt.l+delt.l;
lam_1.l = 0;
lam_2.l = 0;
lam_3.l = 0;

parameter       calibres        Calibrted results;
if(sol_mcp,
    display "com: Calibration by MCP";
    solve calib_mcp using mcp;
    calibres("r","cal") = rt.l;
    calibres("rke","cal") = rket.l;
    calibres("delta","cal") = delt.l;
    calibres("k0","cal") = kt.l;
    calibres("pk0","cal") = 1+phi*(delt.l+gr);
    calibres("vks0","cal")
        = calibres("k0","cal") * calibres("rke","cal");
    calibres("j0","cal") = (delt.l+gr)*kt.l;
    calibres("pka0","cal") = phi*(delt.l+gr)**2 / 2;
    calibres("adjc","cal") = phi*(delt.l+gr) / 2;
    display calibres;
    else
    display "com: Calibration by NLP";
    solve calib_nlp using nlp minimizing loss;
    calibres("r","cal") = rt.l;
    calibres("rke","cal") = rket.l;
    calibres("delta","cal") = delt.l;
    calibres("k0","cal") = kt.l;
    calibres("pk0","cal") = 1+phi*(delt.l+gr);
    calibres("vks0","cal")
        = calibres("k0","cal") * calibres("rke","cal");
    calibres("j0","cal") = (delt.l+gr)*kt.l;
    calibres("pka0","cal") = phi*(delt.l+gr)**2 / 2;
    calibres("adjc","cal") = phi*(delt.l+gr) / 2;
    display calibres;
);
r = rt.l;
delta = delt.l;
```

```
rke0 = rket.l;
rk0 = rke0/(1-tci0);
ks0 = kt.l;
pk0 = calibres("pk0","cal");
pka0 = calibres("pka0","cal");
j0 = calibres("j0","cal");
display "Display calibrated parameters";
display r, delta, rke0, rk0, ks0, pk0, pka0, j0;

*       Benchmark capital inputs.
kd0(i) = ks0*vkd0(i)/vks0;
display kd0;

*       Benchmark net investment demand.
jd0(i) = ishr(i)*j0;
display jd0;

*       wrt labor, quantity = value.
ld0(i) = vld0(i);
display ld0;

*       ---------------------------------------------------------------
*       Reference prices.
display "com: Reference prices.";
parameter
    pl0         Reference wage rate for industry
    rk00        Reference rental price for industry
    rk000       Reference rental price for industry

    ple0        Reference labor price for household (price of leisure)
    rke0        Reference rental price for household
    pm0         Reference price of imported goods
    pc0         Reference good price for household
    pc00        Reference good price for household

    pd0         Reference price for domestic supply
    px0         Reference price for export supply
    pq0         Reference price for output;

pl0(i) = 1+tl0(i);
rk00(i) = (1+tk0(i))*rk0;
rk000(i) = (1+tk0(i));

ple0 = 1-tli0;
rke0 = (1-tci0)*rk0;

pc0(i) = (1+ts0(i));
pc00 = 1.03;
pm0(i) = 1+tm0(i);

px0(i) = 1-tq0(i);
pd0(i) = 1-tq0(i);
pq0(i) = 1-tq0(i);
display pl0, rk00, ple0, rke0, pc0, pm0, px0, pd0;

*       Benchmark current expenditure.
wg0 = cg0 + ple0*lei0;
display wg0;
```

```
*       Benchmark expanded disposal income = labor income + capital income -
*       saving.
epi0 = ple0*endl + rke0*ks0 - vos + vot;
display epi0;
parameter       chk_revex       Check revenue and expenditure consistency;
chk_revex("rev-exp") = round(wg0 - epi0, 3);
display chk_revex;
abort$chk_revex("rev-exp") "Household revenue and expenditure are inconsistent";

*       ---------------------------------------------------------------------
*       Baseline path.
display "com: Baseline path.";
parameter
    qref(t)     Reference quantity path
    pref(t)     Reference price path
    wref(t)     Reference (period) utility path;

*       Reference quantity path.
qref(t) = (1+gr)**(ord(t)-1);

*       Reference price path.
pref(t) = (1/(1+r))**(ord(t)-1);

*       Reference utility path
wref(t) = wg0 * qref(t);

display qref, pref, wref;

parameter
    gqref(t)    Reference government period expenditure
    gbar        Benchmark government lifetime expenditure;
gqref(t) = g0*qref(t);
gbar = sum(t, pref(t) * gqref(t));
display gqref, gbar;

parameter    ubar       Benchmark lifetime utility;
*       Prsigent value of utility over the model horizon:
ubar = sum(t, wref(t) * pref(t));
display ubar;

parameter       adj0;
adj0 = phi*(delta+gr)/2 + sum(i, ishr(i)*phi*(delta+gr)/2);
display adj0;

*       ---------------------------------------------------------------------
*       Baseline tax rate.
display "com: Baseline tax rate..";
parameter
    tq(i)       Indirect tax rate
    tm(i)       Import tax rate
    tl(i)       Tax rate on labor input in production
    tk(i)       Tax rate on capital input in production

    tli         Labor income tax rate
    tci         Capital income tax rate
    ts(i)       Consumption subsidy rate
    tc          Consumption tax rate

    tca(t,e)    Carbon tax rate;
```

```
*       Initial values.
tq(i) = tq0(i);
tm(i) = tm0(i);
tl(i) = tl0(i);
tk(i) = tk0(i);
tc = 0.03;
tli = tli0;
tci = tci0;
ts(i) = ts0(i);
tca(t,e) = 0;
display tq, tm, tl, tk, tc, tli, tci, ts, tca;

*       ---------------------------------------------------------------
*       Calibrate leisure-consumtpion elasticity of substitution.
display "com: Calibrate leisure-consumtpion elasticity of substitution.";
$ontext
Elasticity of substitution between consumption and leisure is calibrated from
labor supply elasticity.
$offtext

sigcl = ( eols * li0 / lei0 + ple0 * li0 / epi0 )
    / (1 - ple0 * lei0 / epi0 );
display eols, sigcl;

*       Check calibration.
parameter sigcl_, sh_l, sh_le, sh_led;

sh_l = ple0*li0 / epi0;
sh_led = ple0*endl / epi0;
sh_le = ple0*lei0 / epi0;
sigcl_("sigcl") = ( eols*sh_l/(sh_led - sh_l) + sh_l ) / (1 - sh_le);
sigcl_("diff") = round(sigcl_("sigcl") - sigcl, 6);
display sh_l, sh_le, sh_led, sigcl_;
abort$sigcl_("diff") "Inconsistent in calibration";

*       ---------------------------------------------------------------
*       Check data.
display "com: Check data.";

*       Check final demand and value added.
chk_fdva(i,"va") = rk00(i)*kd0(i) + pl0(i)*ld0(i)
    + tq0(i)*outs0(i);
chk_fdva("sum","va") = round(sum(i, chk_fdva(i,"va")), 6);

chk_fdva(i,"fd") = cd0(i) + id0(i) + gd0(i) + bop0(i) - tm0(i)*m0(i);
chk_fdva("sum","fd") = round(sum(i, chk_fdva(i,"fd")), 6);

chk_fdva("diff","fd-va")
    = round(chk_fdva("sum","fd") - chk_fdva("sum","va"), 6);
display chk_fdva;
abort$chk_fdva("diff","fd-va") "FD and VA are inconsistent.";

*       Check output data.
chk_out(i,"sec") = sum(ii, int0(ii,i)) + chk_fdva(i,"va");
chk_out(i,"com") = sum(ii, int0(i,ii)) + chk_fdva(i,"fd");
chk_out("sum","sec") = sum(i, chk_out(i,"sec"));
chk_out("sum","com") = sum(i, chk_out(i,"com"));
chk_out(i,"sec_com") = round(chk_out(i,"sec") - chk_out(i,"com"), 6);
```

```
chk_out("sum","sec_com")
    = round(chk_out("sum","sec") - chk_out("sum","com"), 6);
display chk_out;
abort$chk_out("sum","sec_com") "Output data are inconsistent.";

*       ---------------------------------------------------------------------
*       Check investment and savings consistency.
parameter chk_is        check investment and saving;
chk_is("save") = vos;
chk_is("ginv") = i0;
*       Saving = investment because benchmark balance of payment and
*       government fiscal balance are zero.
chk_is("save-ginv")
    = round(chk_is("save") - chk_is("ginv") - bopt0, 6);
display chk_is;
abort$chk_is("save-ginv") "Inconsistent";

*       ---------------------------------------------------------------------
*       Check household income and expenditure.
parameter chk_ie        Check household income and expenditure;
*       Expanded income.
chk_ie("inc") = ple0*endl + rke0*ks0 - vos + vot;
*       Expanded expenditure.
chk_ie("exp") = cg0 + ple0*lei0;
chk_ie("inc-exp")
    = round(chk_ie("inc") - chk_ie("exp"), 6);
display chk_ie;
abort$chk_ie("inc-exp") "Income and expenditure are inconsistent!";

*       ---------------------------------------------------------------------
*       Check government revenue and expenditure
parameter chk_gov       Check government revenue and expenditure;

*       Government revenue.
chk_gov("rev")
    = sum(i, tk0(i) * rk0 * kd0(i)
    + tl0(i) * ld0(i)
    + tq0(i) * (x0(i)+d0(i)))
    + tli0 * li0
    + tci0 * rk0 * ks0
    + sum(i, tm0(i) * m0(i))
    + sum(i, ts0(i) * cd0(i))
    + sum(i, tc0(i) * (1 + ts0(i)) * cd0(i))
    - vot;
chk_gov("exp")= g0;
chk_gov("rev-exp") = round(chk_gov("rev") - chk_gov("exp"), 6);
display chk_gov;
abort$chk_gov("rev-exp") "Government expenditure and revenue are inconsistent";

*       ---------------------------------------------------------------------
*       Adjust energy (quantity) data.
display "com: Adjust energy (quantity) data.";
*       ---------------------------------------------------------------------
*       Check imported energy data.
display "com: Check imported energy data.";
display ioqc, ioqnc, fdq;

parameter cacoef "Carbon coefficient (t-C/10^7kcal)" /
          lim   1.2000000000000E-01
```

```
          coc   1.0451149310000E+00
          sla   1.0154523000000E+00
          cru   7.9152314900000E-01
          nat   5.8507665700000E-01
          pet   7.8289055874027E-01
          cok   1.2310959200000E+00
          gas   5.9652317200000E-01
          /
          ioca  Carbon emissions (MtC);

*       Calculate carbon emissions.
ioca(e,i) = cacoef(e) * ioqc(e,i);
ioca(e,"cons") = cacoef(e) * fdq(e,"cons");

ioca(e,"end") = sum(i, ioca(e,i));
ioca(e,"fd") = ioca(e,"cons");
ioca(e,"sum") = ioca(e,"end") + ioca(e,"fd");

ioca("sum",i) = sum(e, ioca(e,i));
ioca("sum","cons") = sum(e, ioca(e,"cons"));
ioca("sum","sum") = sum(e, ioca(e,"sum"));
display ioca;

*       ---------------------------------------------------------------
*       Check consistency between value and quantity data.
display "com: Check consistency between value and quantity data.";
*       ---------------------------------------------------------------
*       Check the value and quantity consistency
display "com: Check the value and quantity consistency";
$ontext
There are three cases:

(1) Both value data and quantity data are positive.
(2) Value data are positive, but quantity data are zero.
(3) Value data are zero, but quantity data are positive.

Set quantity data to zero for case (2) and (3).

By this adjustment, carbon emissions become less than the real value.
$offtext
parameter
    ch_iov
    ch_iovc
    ch_iovnc

    ch_ioq
    ch_ioqc     Int input for combustion
    ch_ioqnc    Int input for non-combustion
    ch_fdq      Final demand (quantity)

    ch_flag     Flag for check: (0 or 1)
    ch_flag_c   Flag for check: (0 or 1)
    ch_flag_nc  Flag for check: (0 or 1)
    ch_flag_fd  Flag for check: (0 or 1);

*       Check sign consistency.
ch_iov(e,i) = iov(e,i);
ch_iovc(e,i) = iovc(e,i);
ch_iovnc(e,i) = iovnc(e,i);
```

```
ch_ioq(e,i) = ioq(e,i);
ch_ioqc(e,i) = ioqc(e,i);
ch_ioqnc(e,i) = ioqnc(e,i);

ch_flag(e,i) = 0;
ch_flag_c(e,i) = 0;
ch_flag_nc(e,i) = 0;

*       Intermediate inputs.
loop((e,i)$(ch_iov(e,i) or ch_ioq(e,i)),
    if((ch_iov(e,i) * ch_ioq(e,i) > 0),
        ch_flag(e,i) = 0;
        elseif (ch_iov(e,i) * ch_ioq(e,i) = 0),
        ch_flag(e,i)$ch_iov(e,i) = 1;
        ch_flag(e,i)$ch_ioq(e,i) = -1;
    ));
option ch_flag:0;
display ch_flag;

*       Combustion purpose.
loop((e,i)$(ch_iovc(e,i) or ch_ioqc(e,i)),
    if((ch_iovc(e,i) * ch_ioqc(e,i) > 0),
        ch_flag_c(e,i) = 0;
        elseif (ch_iovc(e,i) * ch_ioqc(e,i) = 0),
*               Value data only.
        ch_flag_c(e,i)$ch_iovc(e,i) = 1;
        ioqc(e,i)$ch_iovc(e,i) = 0;
*               Quantity data only.
        ch_flag_c(e,i)$ch_ioqc(e,i) = -1;
        ioqc(e,i)$ch_ioqc(e,i) = 0;
    ));

*       Non-combustion purpose.
loop((e,i)$(ch_iovnc(e,i) or ch_ioqnc(e,i)),
    if((ch_iovnc(e,i) * ch_ioqnc(e,i) > 0),
        ch_flag_nc(e,i) = 0;
        elseif (ch_iovnc(e,i) * ch_ioqnc(e,i) = 0),
*               Value data only.
        ch_flag_nc(e,i)$ch_iovnc(e,i) = 1;
        ioqnc(e,i)$ch_iovnc(e,i) = 0;
*               Quantity data only.
        ch_flag_nc(e,i)$ch_ioqnc(e,i) = -1;
        ioqnc(e,i)$ch_ioqnc(e,i) = 0;
    ));

option ch_flag_c:0, ch_flag_nc:0;
display ch_flag_c, ch_flag_nc;
display "After adjustment", ioqc, ioqnc;

*       Calculate carbon emissions.
ioca(e,i) = cacoef(e) * ioqc(e,i);
ioca(e,"cons") = cacoef(e) * fdq(e,"cons");

ioca(e,"end") = sum(i, ioca(e,i));
ioca(e,"fd") = ioca(e,"cons");
ioca(e,"sum") = ioca(e,"end") + ioca(e,"fd");

ioca("sum",i) = sum(e, ioca(e,i));
```

```
ioca("sum","cons") = sum(e, ioca(e,"cons"));
ioca("sum","sum") = sum(e, ioca(e,"sum"));
display ioca;

*       ---------------------------------------------------------------
*       Contribution ratio.
display "com: Contribution ratio";
parameter       rat_ene         Contribution ratio;
rat_ene(e,i)$(ioqc(e,i)+ioqnc(e,i))
    = ioqc(e,i)/(ioqc(e,i)+ioqnc(e,i));
display rat_ene;

*       ---------------------------------------------------------------
*       Define energy inputs and consumption.
display "com: Define energy inputs and consumption.";
parameter
    veic        Value of energy input (combustion purpose)
    veict       Sumal value of energy input (combustion purpose)
    veinc       Value of energy input (non-combustion purpose)
    vec         Value of energy consumption
    vect        Value of energy for combustion purpose;

veic(e,i) = rat_ene(e,i) * int0(e,i);
veic("ele",i) = int0("ele",i);
veic(e,"sum") = sum(i, veic(e,i));
veinc(e,i) = int0(e,i) - veic(e,i);
display veic, veinc;

vec(e) = cd0(e);
display vec;

veict(i) = sum(ec, veic(ec,i)) + veic("ele",i);
display veict;

vect(e) = veic(e,"sum") + vec(e);
display vect;

*       Value of energy inputs
ve0(i)$outs0(i) = sum(ec, veic(ec,i)) + sum(ele, int0(ele,i));
display ve0;

*       Primary factor and energy.
vpfe0(i)$outs0(i) = vpf0(i) + ve0(i);
display vpfe0;

*       ---------------------------------------------------------------
*       Deriving carbon emission data.
display "com: Deriving carbon emission data.";

parameter
    carbi0      Carbon emissions from intermediate inputs (MtC)
    carbc0      Carbon emissions from final consumption (MtC)
    carbt0      Carbon emissions (MtC)

    carbl       Carbon emission restriction (MtC)
    carbl0      Benchmark carbon emissions (MtC)

    scale_ca    Scale factor    / 1000 /
;
```

```
*       Carbon emissions from intermediate inputs.
carbi0(e,i) = cacoef(e)*ioqc(e,i) / scale_ca;
carbi0(e,"sum") = sum(i, carbi0(e,i));
carbi0("sum",i) = sum(e, carbi0(e,i));
carbi0("sum","sum") = sum((e,i), carbi0(e,i));

*       Carbon emissions from final demand (consumption).
carbc0(e) = cacoef(e)*fdq(e,"cons") / scale_ca;
carbc0("sum") = sum(e, carbc0(e));
display carbi0, carbc0;

carbt0(e) = carbi0(e,"sum") + carbc0(e);
carbt0("sum") = sum(e, carbt0(e));
display carbt0;

*       Benchmark carbon emissions.
carbl0 = sum((e,i), carbi0(e,i)) + sum(e, carbc0(e));
*       Carbon emission restriction is set to zero tentatively.
* carbl(t) = 0;
carbl(t) = carbl0 * 1e+6;
option carbl0:6, carbl:6;
display carbl0, carbl;

parameter       chk_carbon      Check carbon emissions (MtC);

*       Carbon emissions from intermediate inputs.
chk_carbon(e,i) = carbi0(e,i);
chk_carbon(e,"int") = sum(i, chk_carbon(e,i));
chk_carbon("sum","int") = sum(e, chk_carbon(e,"int"));
chk_carbon("sum",i) = sum(e, chk_carbon(e,i));
*       Carbon emissions from consumption.
chk_carbon(e,"con") = carbc0(e);
chk_carbon("sum","con") = sum(e, chk_carbon(e,"con"));
chk_carbon(e,"sum") = chk_carbon(e,"int") + chk_carbon(e,"con");
*       Total carbon emissions.
chk_carbon("sum","sum") = sum(e, chk_carbon(e,"sum"));
option chk_carbon:3;
display chk_carbon;

*       ---------------------------------------------------------------
*       Scale parameter
display "com: Scale parameter";
parameter
    scale_k     Scale for benchmark capital stock               / 1 /
    scale_l     Scale for labor endowment                       / 1 /
    scale_t     Scale for transfer to household                 / 1 /
    scale_g     Scale for government lifetime expenditure       / 1 /;
display scale_k, scale_l, scale_t, scale_g;

*       ---------------------------------------------------------------
*       Tax instruments.
display "com: Tax instruments.";
set     taxinst         Tax instruments
        / bau   No carbon limit
          lum   Lump-sum tax
          lin   Labor income tax
          lab   Labor tax on production
          cin   Capital income tax
          cap   Capital tax on production
```

```
          ctx   Consumption tax
          imp   Import tax /
        tins(taxinst)   Tax instrument;
tins(taxinst) = no;
tins("lum") = yes;
display taxinst, tins;

*       ---------------------------------------------------------------
*       MCP code
display "com: MCP code";

*       ---------------------------------------------------------------
*       Share parameters
display "com: Share parameters";

parameter
    cost_q(i)           Unit cost for production

    sh_x(i)             Share of export in output
    sh_fe(i)            Share of primary factor and energy in production cost
    sh_f(i)             Share of primary factor in PF-E aggregation
    sh_fl(i)            Share of labor in primary factor
    sh_e                Share of energy goods in production
    sh_nen(ii,i)        Share of non-energy intermediate inputs
    sh_nc(e,i)          Share of energy for non-combustion purpose
    sh_cl(cl,i)         Share of cok and lim for combustion purpose

    sh_nene             Share of non-energy goods in consumption.
    sh_c(i)             Share of each good in non-energy consumption.
    sh_ec(i)            Share of each good in energy consumption.
    sh_lei              Share of leisure
    sh_u(t)             Share of period utility
    sh_g(t)             Share of each good in government expenditure

    sh_ad(i)            Share of domestic good in Armington aggregation
    sh_inv(i)           Share of input in investment activity;

*       Production cost (net of tax).
cost_q(i)$out0(i) = (1-tq0(i))*out0(i);

*       Share of export.
sh_x(i)$out0(i) = x0(i) / out0(i);

*       Share of primary factor-energy composite
sh_fe(i)$out0(i) = vpfe0(i) / cost_q(i);

*       Share of PF in PFE composite
sh_f(i)$vpfe0(i) = vpf0(i)/ vpfe0(i);

*       Share of labor in PF composite
sh_fl(i)$vpf0(i) = pl0(i)*ld0(i) / vpf0(i);

*       Share of energy in energy composite.
sh_e(ec,i)$veict(i) = veic(ec,i) / veict(i);

*       Share of electricity in energy composite
sh_e(ele,i)$veict(i) = veic(ele,i) / veict(i);

*       Share of non-energy inputs.
```

```
sh_nen(nen,i)$out0(i) = int0(nen,i) / cost_q(i);

*       Share of energy input for non-combustion purpose.
sh_nc(e,i)$out0(i) = veinc(e,i) / cost_q(i);

*       Share of cok and lim for combustion purpose.
sh_cl(cl,i)$out0(i) = veic(cl,i) / cost_q(i);

*       Government expenditure
sh_g(t) = pref(t) * qref(t) / sum(tt, pref(tt) * qref(tt));

display cost_q, sh_x, sh_fe, sh_f, sh_fl, sh_e, sh_nen, sh_nc, sh_cl;

*       Share of domestic goods in Armington aggregation.
sh_ad(i)$a0(i) = d0(i) / a0(i);
display sh_ad;

*       Share of each good in investment.
sh_inv(i)$j0 = jd0(i) / j0;
display sh_inv;

*       Share of leisure in total expenditure.
sh_lei = ple0*lei0 / wg0;

*       Share of non-energy goods in consumption.
sh_nene = cne0 / c0;

*       Share of each good in non-energy consumption.
sh_c(nene) = pc0(nene) * cd0(nene) / cne0;

*       Share of each good in energy consumption.
sh_ec(ene) = pc0(ene) * cd0(ene) / ce0;
display sh_lei, sh_nene, sh_c, sh_ec;

*       Share of each period utility.
sh_u(t) = pref(t)*wref(t) / ubar;
display sh_u;

*       ---------------------------------------------------------------
*       MCP formulation
display "com: MCP formulation";
variables
*       Unit demand and supply
    a_x(t,i)            Export supply
    a_d(t,i)            Domestic supply
    a_l(t,i)            Demand for labor
    a_k(t,i)            Demand for capital
    a_e(t,ii,i)         Demand for energy (exc. cok and lim) in production

    a_ec(t,i)           Consumption demand for energy goods
    a_c(t,i)            Consumption demand for non-energy goods
    a_cc(t)             Demand for aggregate consumption
    a_lei(t)            Demand for leisure
    a_u(t)              Demand for period utility

    a_ad(t,i)           Demand for domestic good from Armington activity
    a_am(t,i)           Demand for imported good from Armington activity

*       Activity variables
```

```
    q_q(t,i)            Activity level for production
    q_a(t,i)            Armington aggregation
    q_x(t,i)            Export activity
    q_m(t,i)            Import activity
    q_cc(t)             Consumption aggregation activity
    q_w(t)              Period utility
    q_j(t)              Net investment
    q_u                 Lifetime utility
    q_k(t)              Capital accumulation (investment) activity
    q_g(t)              Government expenditure at each period
    q_gle               Government lifetime expenditure
    q_cnc               Carbon emissions
    q_sl(t)             Labor supply

*       Unit cost.
    c_cc(t)             Unit cost of consumption aggregation
    c_a(t,i)            Unit cost of Armington aggregation
    c_w(t)              Unit cost of period utility (expenditure function)
    c_u                 Unit cost of lifetime utility (expenditure function)
    c_q(t,i)            Unit cost of production
    c_e(t,i)            Price index of emission sources for combustion purpose

*       Price variables
    p_q(t,i)            Price index (unit revenue) for output
    p_f(t,i)            Price index of primary factor composite
    p_ee(t,i)           Price index of energy composite in production
    p_fe(t,i)           Price index of primary factor-energy composite
    p_c(t)              Price index of non-energy composite in consumption
    p_ec(t)             Price index of energy composite in consumption
    p_i(t)              Price index of investment good
    p_e(t,i)            Price index of emission sources for combustion purpose

    p_cc(t)             Price index of aggregate consumption
    p_d(t,i)            Price of domestic good
    p_x(t,i)            Price of export good
    p_m(t,i)            Price of import good
    p_fxv               Exchange rate (intertemporal)
    p_fx                Exchange rate
    p_l(t)              Wage rate for producer
    p_le(t)             Wage rate for household
    p_w(t)              Price of period utility
    p_u                 Price of lifetime utility
    p_a(t,i)            Price of Armington good
    p_g(t)              Price of government expenditure
    p_k(t)              Shadow price of capital
    p_rk(t)             Rental price for industry
    p_rke(t)            Rental price for household
    p_gle               Price index for government lifetime consumption
    p_ca(t)             Price of emission permits
    p_ka(t)             Adjustmnet premium
    p_disc              Discount factor
    p_intr              "Interest rate 1/(1+r)"

*       Income variables
    v_inc_h             Extended lifetime income of household
    v_inc_g(t)          Government income (period)
    v_inc_gl            Government income (lifetime)

    q_tcap              post-terminal capital stock
```

```
    m_tau               Tax multiplier
;
equations
*       Unit demand and supply
    e_a_x(t,i)          Export supply
    e_a_d(t,i)          Domestic supply
    e_a_l(t,i)          Demand for labor
    e_a_k(t,i)          Demand for capital
    e_a_e(t,ii,i)       Demand for energy (exc. cok and lim) in production

    e_a_ec(t,i)         Consumption demand for energy goods
    e_a_c(t,i)          Consumption demand for non-energy goods
    e_a_cc(t)           Demand for aggregate consumption
    e_a_lei(t)          Demand for leisure
    e_a_u(t)            Demand for period utility

    e_a_ad(t,i)         Demand for domestic good from Armington activity
    e_a_am(t,i)         Demand for imported good from Armington activity

*       Zero profit conditions
    e_q_q(t,i)          Activity level for production
    e_q_a(t,i)          Armington aggregation
    e_q_x(t,i)          Export activity
    e_q_m(t,i)          Import activity
    e_q_cc(t)           Consumption aggregation activity
    e_q_w(t)            Period utility
    e_q_g(t)            government expenditure at each period
    e_q_j(t)            Net investment
    e_q_u               Lifetime utility
    e_q_k(t)            Capital accumulation (investment) activity
    e_q_gle             Government lifetime expenditure
    e_q_cnc             Carbon emissions
    e_q_sl(t)           Labor supply

*       Unit cost.
    e_c_cc(t)           Unit cost of consumption aggregation
    e_c_a(t,i)          Unit cost of Armington aggregation
    e_c_w(t)            Unit cost of utility (expenditure function)
    e_c_u               Unit cost of lifetime utility (expenditure function)
    e_c_q(t,i)          Unit cost of production
    e_c_e(t,e)          Price index for energy

*       Price variables
    e_p_q(t,i)          Price index (unit revenue) for activity
    e_p_f(t,i)          Price index of primary factor composite
    e_p_ee(t,i)         Price index of energy composite in production
    e_p_fe(t,i)         Price index of primary factor-energy composite
    e_p_c(t)            Price index of non-energy composite in consumption
    e_p_ec(t)           Price index of energy composit in consumption
    e_p_i(t)            Price index for investment
    e_p_e(t,i)          Price index of emission sources for combustion purpose

    e_p_cc(t)           Price index for aggregate consumption
    e_p_d(t,i)          Price of domestic good
    e_p_x(t,i)          Price of export good
    e_p_m(t,i)          Price of import good
    e_p_fxv             Exchange rate (intertemporal)
    e_p_fx              Exchange rate
    e_p_l(t)            Wage rate for producer
```

```
    e_p_le(t)           Wage rate for household
    e_p_w(t)            Price of utility
    e_p_u               Price of lifetime utility
    e_p_a(t,i)          Price of Armington good
    e_p_g(t)            Price of government expenditure
    e_p_k(t)            Shadow price of capital
    e_p_rk(t)           Rental price for industry
    e_p_rke(t)          Rental price for household
    e_p_gle             Price index for government lifetime consumption
    e_p_ca(t)           Price of emission permits
    e_p_ka(t)           Adjustmnet premium
    e_p_disc            Discount factor
    e_p_intr            Interest rate

*       Income balance
    e_v_inc_h           Extended lifetime income of household
    e_v_inc_g           Government income (period)
    e_v_inc_gl          Government income (lifetime)

    e_q_tcap            Post-terminal capital stock
    e_m_tau             Tax multiplier;

*       ---------------------------------------------------------------
*       Unit cost and price index.

*       Unit cost of consumption aggregation.
e_c_cc(t) ..

    c_cc(t) =e=
    ((sh_nene * p_c(t)**(1-sigcc) + (1-sh_nene) * p_ec(t)**(1-sigcc)
    )**(1/(1-sigcc)))$(sigcc ne 1)
    +
    ( p_c(t)**sh_nene * p_ec(t)**(1-sh_nene) )$(sigcc = 1);

*       Unit cost of Armington aggregation
e_c_a(t,i)$a0(i) ..

    c_a(t,i) =e=
    ( (1$(1-d0(i)) + p_d(t,i)$d0(i))**sh_ad(i)
    * (1$(1-m0(i)) + (p_m(t,i))$m0(i))**(1-sh_ad(i)) )$(siga(i) = 1)
    +
    ( ( (sh_ad(i) * p_d(t,i)**(1-siga(i)) )$d0(i)
    + ( (1-sh_ad(i)) * (p_m(t,i))**(1-siga(i)) )$m0(i)
    )**(1/(1-siga(i))) )$(siga(i) ne 1);

*       Unit cost of period utility
e_c_w(t) ..

    c_w(t) =e=
    ( (p_le(t))**(sh_lei)
    * ((1 + tc*(1-m_tau$tins("ctx"))) * p_cc(t) / pc00)**(1-sh_lei)
    )$(sigcl = 1)
    +
    ((sh_lei * p_le(t)**(1-sigcl)
    + (1-sh_lei) * ((1 + tc*(1-m_tau$tins("ctx"))) * p_cc(t)
    / pc00)**(1-sigcl))**(1/(1-sigcl))
    )$(sigcl ne 1);

*       Unit cost of lifetime utility
```

```
e_c_u ..

    c_u =e=
    ( prod(t, (p_w(t) / pref(t))**sh_u(t) ) )$(sigu = 1)
    +
    ( ( sum(t, sh_u(t) * (p_w(t) / pref(t))**(1-sigu) ) )**(1/(1-sigu))
    )$(sigu ne 1);

*       Unit cost of production.
e_c_q(t,i)$out0(i) ..

    c_q(t,i) =e=
    sum(nen$sh_nen(nen,i), sh_nen(nen,i) * p_a(t,nen))
    + sum(e$sh_nc(e,i), sh_nc(e,i) * p_a(t,e))
    + sum(cl$sh_cl(cl,i), sh_cl(cl,i) * p_e(t,cl))
    + sh_fe(i) * p_fe(t,i);

*       Price index of output.
e_p_q(t,i)$out0(i) ..

    p_q(t,i) =e=
    ( sh_x(i) * p_x(t,i)**(1+eta)
    + (1-sh_x(i)) * p_d(t,i)**(1+eta) )**(1/(1+eta));

*       Price index of primary factor composite.
e_p_f(t,i)$vpf0(i) ..

    p_f(t,i) =e=
    ( ((1+tl(i)*(1-m_tau$tins("lab"))) * p_l(t) / pl0(i))**sh_fl(i)
    * ((1+tk(i)*(1-m_tau$tins("cap"))) * p_rk(t) / rk000(i))**(1-sh_fl(i))
    )$(sigpf(i) = 1)
    +
    ( ( sh_fl(i) * ((1+tl(i)*(1-m_tau$tins("lab")))
    * p_l(t) / pl0(i))**(1-sigpf(i))
    + (1-sh_fl(i)) * ((1+tk(i)*(1-m_tau$tins("cap")))
    * p_rk(t) / rk000(i))**(1-sigpf(i))
    )**(1/(1-sigpf(i))) )$(sigpf(i) ne 1);

*       Price index for energy composite in production.
e_p_ee(t,i)$veict(i) ..

    p_ee(t,i) =e=
    (( sum(ec$sh_e(ec,i), sh_e(ec,i) * p_e(t,ec)**(1-sigee))
    + sh_e("ele",i) * p_a(t,"ele")**(1-sigee))**(1/(1-sigee)) )$(sigee ne 1)
    +
    ( prod(ec$sh_e(ec,i), p_e(t,ec)**(sh_e(ec,i)))
    * p_a(t,"ele")**(sh_e("ele",i)) )$(sigee = 1);

*       Price index for primary factor-energy composite
e_p_fe(t,i)$vpfe0(i) ..

    p_fe(t,i) =e=
    ( (p_f(t,i))**sh_f(i) * p_ee(t,i)**(1-sh_f(i)))$(sigpfe = 1)
    +
    ( (sh_f(i) * p_f(t,i)**(1-sigpfe)
    + (1-sh_f(i)) * p_ee(t,i)**(1-sigpfe))**(1/(1-sigpfe)) )$(sigpfe ne 1);

*       Price index for non-energy composite in consumption.
e_p_c(t) ..
```

```
    p_c(t) =e=
    ( prod(nene$sh_c(nene),
    ((1+ts(nene))*p_a(t,nene) / pc0(nene))**sh_c(nene)) )$(sigc = 1)
    +
    (( sum(nene$sh_c(nene),
    sh_c(nene) * ((1+ts(nene))*p_a(t,nene) / pc0(nene))**(1-sigc))
    )**(1/(1-sigc)) )$(sigc ne 1);

*       Price index for energy composite in consumption.
e_p_ec(t) ..

    p_ec(t) =e=
    ( prod(ec$sh_ec(ec),
    ((1+ts(ec))*p_e(t,ec) / pc0(ec))**sh_ec(ec))
    * (((1+ts("ele"))*p_a(t,"ele") / pc0("ele"))**sh_ec("ele"))$sh_ec("ele")
    )$(sigec = 1)
    +
    (( sum(ec$sh_ec(ec), sh_ec(ec)
    * ((1+ts(ec))*p_e(t,ec) / pc0(ec))**(1-sigec))
    + (sh_ec("ele") * ((1+ts("ele"))*p_a(t,"ele") / pc0("ele"))**(1-sigec)
    )$sh_ec("ele")
    )**(1/(1-sigec)) )$(sigec ne 1);

*       Price index of emission sources.
e_c_e(t,e) ..

    c_e(t,e) =e= p_a(t,e) + carbt0(e) * p_ca(t) / vect(e);

*       Price of investment good.
e_p_i(t) ..

    p_i(t) * j0 =e= sum(i, p_a(t,i) * jd0(i));

*       Wage rate for household.
e_p_le(t) ..

    ple0 * p_le(t) =e= (1-tli*(1-m_tau$tins("lin"))) * p_l(t);

*       Rental price for household.
e_p_rke(t) ..

    rke0 * p_rke(t) =e=
    (1-tci*(1-m_tau$tins("cin"))) * rk0 * p_rk(t);

*       Adjustment premium.
e_p_ka(t)$phi ..

    p_ka(t) * pka0 =e= phi * (q_j(t)*j0/(q_k(t)*ks0))**2 * p_i(t) / 2;

*       ---------------------------------------------------------------
*       Zero profit conditions

*       Production activity.
e_q_q(t,i)$out0(i) ..

    c_q(t,i) =g= (1-tq(i)) * p_q(t,i) / pq0(i);

*       Armington aggregation activity.
```

```
e_q_a(t,i)$a0(i) ..

    c_a(t,i) =g= p_a(t,i);

*       Export activity.
e_q_x(t,i)$x0(i) ..

    p_x(t,i) =g= p_fxv$capflo + p_fx(t)$bopcon;

*       Import activity.
e_q_m(t,i)$m0(i) ..

    ( (1+tm(i)*(1-m_tau$tins("imp"))) * p_fxv  )$capflo
    +
    ( (1+tm(i)*(1-m_tau$tins("imp"))) * p_fx(t) )$bopcon
    =g= pm0(i) * p_m(t,i);

*       Consumption aggregation activity.
e_q_cc(t) ..

    c_cc(t) =g= p_cc(t);

*       Period utility production activity.
e_q_w(t) ..

    p_disc(t) * c_w(t) =g= p_w(t);

*       Lifetime utility production activity.
e_q_u$ubar ..

    c_u =g= p_u;

*       Government expenditure (period)
e_q_g(t) ..

    sum(i, p_a(t,i) * gd0(i)) =g= p_g(t) * g0;

*       Government expenditure (lifetime)
e_q_gle$(not gbb) ..

    sum(t, sh_g(t) * p_disc(t) * p_g(t) / pref(t)) =g= p_gle;

*       Investment good production.
e_q_j(t) ..

    p_i(t) * ( 1 + (phi * (q_j(t)*j0/(q_k(t)*ks0)))$phi ) =g= pk0 * p_k(t);

*       Capital accumulation (investment) activity.
e_q_k(t)$(not tfirst(t)) ..

    pk0 * p_k(t-1) / p_intr(t) =g=
    rke0 * p_rke(t) + (1-delta) * pk0 * p_k(t) + (p_ka(t) * pka0)$phi;

*       Carbon emissions.
e_q_cnc(t,e)$vect(e) ..

    c_e(t,e) =g= p_e(t,e);

*       ---------------------------------------------------------------
```

```
*       Unit demand and supply

*       Export supply.
e_a_x(t,i)$x0(i) ..

    a_x(t,i) =e= x0(i) * (p_x(t,i) / p_q(t,i))**eta;

*       Domestic supply.
e_a_d(t,i)$d0(i) ..

    a_d(t,i) =e= d0(i) * (p_d(t,i) / p_q(t,i))**eta;

*       Labor demand.
e_a_l(t,i)$ld0(i) ..

    a_l(t,i) =e=
    ld0(i) * (p_f(t,i)
    / ((1+tl(i)*(1-m_tau$tins("lab")))*p_l(t) / pl0(i)))**sigpf(i)
    * (p_fe(t,i) / p_f(t,i))**sigpfe;

*       Capital demand.
e_a_k(t,i)$kd0(i) ..

    a_k(t,i) =e=
    kd0(i) * (p_f(t,i)
    / ((1+tk(i)*(1-m_tau$tins("cap")))*p_rk(t) / rk000(i)))**sigpf(i)
    * (p_fe(t,i) / p_f(t,i))**sigpfe;

*       Demand for energy (exc. cok and lim) in production.
e_a_e(t,i,ii)$(ene(i) and veic(i,ii)) ..

    a_e(t,i,ii) =e=
    (veic(i,ii) * (p_ee(t,ii) / p_e(t,i))**sigee
    * (p_fe(t,ii) / p_ee(t,ii))**sigpfe)$ec(i)
    +
    (veic(i,ii) * (p_ee(t,ii) / p_a(t,i))**sigee
    * (p_fe(t,ii) / p_ee(t,ii))**sigpfe)$ele(i);

*       Consumption demand for energy goods
e_a_ec(t,i)$(ene(i) and cd0(i))..

    a_ec(t,i) =e=
    (cd0(i) * (p_ec(t) / ((1+ts(i))*p_e(t,i) / pc0(i)))**sigec
    * (p_cc(t) / p_ec(t))**sigcc)$ec(i)
    +
    (cd0(i) * (p_ec(t) / ((1+ts(i))*p_a(t,i) / pc0(i)))**sigec
    * (p_cc(t) / p_ec(t))**sigcc)$ele(i);

*       Consumption demand for non-energy goods
e_a_c(t,i)$(nene(i) and cd0(i)) ..

    a_c(t,i) =e=
    ( (cd0(i) * (p_c(t) / ((1+ts(i))*p_a(t,i) / pc0(i)))**sigc
    * (p_cc(t) / p_c(t))**sigcc) )$nen(i)
    +
    ( (cd0(i) * (p_c(t) / ((1+ts(i))*p_e(t,i) / pc0(i)))**sigc
    * (p_cc(t) / p_c(t))**sigcc) )$cl(i);

*       Demand for aggregate consumption.
```

```
e_a_cc(t) ..

    a_cc(t)
    =e=
    c0 * (c_w(t) / ((1+tc*(1-m_tau$tins("ctx"))) * p_cc(t) / pc00))**sigcl;

*       Demand for leisure.
e_a_lei(t) ..

    a_lei(t) =e= lei0 * (c_w(t) / p_le(t))**sigcl;

*       Demand for period utility.
e_a_u(t) ..

    a_u(t) =e= wref(t) * (c_u / (p_w(t) / pref(t)))**sigu;

*       Demand for domestic good from Armington activity.
e_a_ad(t,i)$d0(i) ..

    a_ad(t,i) =e= d0(i) * (c_a(t,i) / p_d(t,i))**siga(i);

*       Demand for imported good from Armington activity.
e_a_am(t,i)$m0(i) ..

    a_am(t,i) =e= m0(i) * (c_a(t,i) / p_m(t,i))**siga(i);

*       Labor supply.
e_q_sl(t) ..

    li0 * q_sl(t) =e= endl * scale_l * qref(t) - a_lei(t) * q_w(t);

*       ---------------------------------------------------------------
*       Market clearing conditions.

*       Market for aggregate consumption.
e_p_cc(t) ..

    c0 * q_cc(t) =g= a_cc(t) * q_w(t);

*       Market for domestic goods.
e_p_d(t,i)$d0(i) ..

    a_d(t,i) * q_q(t,i) =g= a_ad(t,i) * q_a(t,i);

*       Market for export goods
e_p_x(t,i)$x0(i) ..

    a_x(t,i) * q_q(t,i) =g= x0(i) * q_x(t,i);

*       Market for import goods.
e_p_m(t,i)$m0(i) ..

    m0(i) * q_m(t,i) =g= a_am(t,i) * q_a(t,i);

*       Exchange rate (intertemporal).
e_p_fxv$capflo ..

    sum(t, sum(i$x0(i), pref(t) * x0(i) * q_x(t,i)))
    =g= sum(t, sum(i$m0(i), pref(t) * m0(i) * q_m(t,i)));
```

```
*       Exchange rate (non-intertemporal).
e_p_fx(t)$bopcon ..

    sum(i$x0(i), x0(i) * q_x(t,i)) =g= sum(i$m0(i), m0(i) * q_m(t,i));

*       Market for labor.
e_p_l(t) ..

    li0 * q_sl(t) =g= sum(i$ld0(i), a_l(t,i) * q_q(t,i));

*       Market for period utility.
e_p_w(t) ..

    wg0 * q_w(t) =g= a_u(t) * q_u;

*       Market for lifetime utility.
e_p_u ..

    v_inc_h =g= ubar * q_u * p_u;

*       Market for Armington goods.
e_p_a(t,i) ..

    a0(i) * q_a(t,i) =g=
    ( (a_c(t,i) * q_cc(t))$cd0(i)
    + sum(ii$int0(i,ii), int0(i,ii) * q_q(t,ii)) )$nen(i)

    + ( a_ec(t,i) * q_cc(t)
    + sum(ii$veic(i,ii), a_e(t,i,ii) * q_q(t,ii)) )$ele(i)

    + ( vect(i) * q_cnc(t,i)
    + sum(ii$veinc(i,ii), veinc(i,ii) * q_q(t,ii)) )$cl(i)

    + ( vect(i) * q_cnc(t,i)
    + sum(ii$veinc(i,ii), veinc(i,ii) * q_q(t,ii)) )$ec(i)

    + sh_inv(i) * j0 * q_j(t)

    + (sh_inv(i) * phi * ((q_j(t)*j0)**2 /(2*q_k(t)*ks0)))$phi

    + gd0(i) * q_g(t);

*       Government expenditure (period).
e_p_g(t) ..

    (g0 * q_g(t) - gqref(t) * q_gle)$(not gbb)
    + (v_inc_g(t) - g0 * p_g(t) * q_g(t))$gbb =g= 0;

*       Shadow price of capital.
e_p_k(t) ..

    j0 * q_j(t) + (1-delta) * ks0 * q_k(t)
    =g= ks0 * (q_k(t+1)$(not tlast(t)) + q_tcap$tlast(t));

*       Market for renting capital.
e_p_rk(t) ..

    ks0 * q_k(t) =g= sum(i$kd0(i), a_k(t,i) * q_q(t,i));
```

```
*       Price index for government lifetime consumption.
e_p_gle$(not gbb) ..

    v_inc_gl =g= gbar * q_gle * p_gle;

*       Market for emission permit.
e_p_ca(t) ..

    carbl(t) =g= sum(e, carbt0(e) * q_cnc(t,e));

*       Price index of emission sources.
e_p_e(t,i)$e(i) ..

    vect(i) * q_cnc(t,i) =g=
    ((a_c(t,i) * q_cc(t))$cd0(i)
    + sum(ii$veic(i,ii), veic(i,ii) * q_q(t,ii)))$cl(i)
    +
    ((a_ec(t,i) * q_cc(t))$cd0(i)
    + sum(ii$veic(i,ii), a_e(t,i,ii) * q_q(t,ii)))$ec(i);

*       Interest rate.
e_p_intr(t)$(not tfirst(t)) ..

    p_intr(t) =e=
    (p_w(t) / p_w(t-1))$bopcon
*     (1/(1+r - p_w(t)/p_w(t-1)))$bopcon
    + (1/(1+r))$capflo;

*       Discount factor.
e_p_disc(t) ..

    p_disc(t) =e=
    (prod(tt$((not tfirst(tt)) and (ord(tt) le ord(t))),
    p_intr(tt)))$(not tfirst(t))
    + 1$tfirst(t);

*       ---------------------------------------------------------------
*       Income constraints.

*       Household lifetime income.
e_v_inc_h ..

    v_inc_h =e=

    sum(t, p_disc(t) * p_le(t) * ple0 * endl * scale_l * qref(t))

    + sum(t$tfirst(t), (rke0 * p_rke(t) + (1-delta) * pk0 * p_k(t)
    + (pka0 * p_ka(t))$phi)) * ks0 * scale_k

    - sum(tlast, p_disc(tlast) * pk0 * p_k(tlast) *  ks0 * q_tcap)

    + sum(t, p_disc(t) * p_g(t) * vot * scale_t * qref(t))

    + (p_u * m_tau)$tins("lum");

*       Government income (period).
e_v_inc_g(t) ..
```

```
    v_inc_g(t) =e=

    sum(i$ld0(i),
    tl(i) * (1-m_tau$tins("lab")) * p_l(t) * a_l(t,i) * q_q(t,i))

    + sum(i$kd0(i), tk(i) * (1-m_tau$tins("cap"))
    * rk0 * p_rk(t) * a_k(t,i) * q_q(t,i))

    + sum(i$out0(i), tq(i) * p_q(t,i) * out0(i) * q_q(t,i))

    + sum(i$m0(i), tm(i) * (1-m_tau$tins("imp"))
    * (p_fxv$capflo + p_fx(t)$bopcon) * m0(i) * q_m(t,i))

    + tli * (1-m_tau$tins("lin")) * p_l(t) * li0 * q_sl(t)

    + tci * (1-m_tau$tins("cin")) * rk0 * p_rk(t) * ks0 * q_k(t)

    + tc * (1-m_tau$tins("ctx")) * p_cc(t) * a_cc(t) * q_w(t)

    + ( sum(nen$cd0(nen), ts(nen) * p_a(t,nen) * a_c(t,nen))
    + sum(ele$cd0(ele), ts(ele) * p_a(t,ele) * a_ec(t,ele))
    + sum(cl$cd0(cl), ts(cl) * p_a(t,cl) * a_c(t,cl))
    + sum(ec$cd0(ec), ts(ec) * p_a(t,ec) * a_ec(t,ec)) ) * q_cc(t)

    - p_g(t) * vot * scale_t * qref(t)

    + (p_ca(t) * carbl(t))$carbl(t);

*       Government income (lifetime).
e_v_inc_gl$(not gbb) ..

    v_inc_gl =e= sum(t, p_disc(t) * v_inc_g(t))

    - p_u * m_tau$tins("lum");

*       ---------------------------------------------------------------
*       Terminal condition.
e_q_tcap ..

    sum(t$tlast(t), q_j(t) * q_w(t-1) - q_j(t-1) * q_w(t)) =e= 0;

*       ---------------------------------------------------------------
*       Constraints

e_m_tau$((not gbb)
    and (tins("lab") or tins("cap") or tins("lin")
    or tins("cin") or tins("ctx")
    or tins("imp") or tins("lum"))) ..

    q_gle =e= scale_g;

*       ---------------------------------------------------------------
*       Model declaration.
model dynamic_mcp Dynamic model in MCP format /

      e_a_x.a_x, e_a_d.a_d, e_a_l.a_l, e_a_k.a_k, e_a_e.a_e, e_a_ec.a_ec,
      e_a_c.a_c, e_a_cc.a_cc, e_a_lei.a_lei, e_a_u.a_u, e_a_ad.a_ad,
      e_a_am.a_am,
```

```
      e_q_q.q_q, e_q_a.q_a, e_q_x.q_x, e_q_m.q_m, e_q_cc.q_cc, e_q_w.q_w,
      e_q_u.q_u, e_q_g.q_g, e_q_j.q_j, e_q_sl.q_sl, e_q_k.q_k,
      e_q_gle.q_gle, e_q_cnc.q_cnc,

      e_c_cc.c_cc, e_c_a.c_a, e_c_w.c_w, e_c_u.c_u, e_c_q.c_q, e_c_e.c_e,

      e_p_q.p_q, e_p_f.p_f, e_p_ee.p_ee, e_p_fe.p_fe, e_p_c.p_c, e_p_ec.p_ec,
      e_p_e.p_e, e_p_k.p_k, e_p_rke.p_rke, e_p_rk.p_rk, e_p_gle.p_gle,
      e_p_i.p_i, e_p_ka.p_ka, e_p_intr.p_intr, e_p_disc.p_disc,

      e_p_cc.p_cc, e_p_d.p_d, e_p_x.p_x, e_p_m.p_m, e_p_fxv.p_fxv,
      e_p_fx.p_fx, e_p_l.p_l, e_p_le.p_le, e_p_w.p_w, e_p_u.p_u, e_p_a.p_a,
      e_p_g.p_g, e_p_ca.p_ca,

      e_v_inc_h.v_inc_h, e_v_inc_g.v_inc_g, e_v_inc_gl.v_inc_gl,
      e_q_tcap.q_tcap, e_m_tau.m_tau /;

*       ---------------------------------------------------------------
*       Lower bounds for variables.

*       Unit demand.
a_x.lo(t,i) = 0;
a_d.lo(t,i) = 0;
a_l.lo(t,i) = 0;
a_k.lo(t,i) = 0;
a_e.lo(t,ii,i) = 0;
a_ec.lo(t,i) = 0;
a_c.lo(t,i) = 0;
a_cc.lo(t) = 0;
a_lei.lo(t) = 0;
a_u.lo(t) = 0;
a_ad.lo(t,i) = 0;
a_am.lo(t,i) = 0;

*       Activity.
q_q.lo(t,i) = 0;
q_a.lo(t,i) = 0;
q_x.lo(t,i) = 0;
q_m.lo(t,i) = 0;
q_cc.lo(t) = 0;
q_w.lo(t) = 0;
q_g.lo(t) = 0;
q_j.lo(t) = 0;
q_sl.lo(t) = 0;
q_u.lo = 0;
q_k.lo(t) = 0;
q_gle.lo = 0;
q_cnc.lo(t,e) = 0;
q_tcap.lo = 0;

*       Unit cost.
c_cc.lo(t) = 1e-6;
c_a.lo(t,i) = 1e-6;
c_w.lo(t) = 1e-6;
c_u.lo = 1e-6;
c_q.lo(t,i) = 1e-6;

*       Price.
p_q.lo(t,i) = 1e-6;
```

```
p_f.lo(t,i) = 1e-6;
p_ee.lo(t,i) = 1e-6;
p_fe.lo(t,i) = 1e-6;
p_c.lo(t) = 1e-6;
p_ec.lo(t) = 1e-6;
p_e.lo(t,e) = 1e-6;
p_cc.lo(t) = 1e-6;
p_d.lo(t,i) = 1e-6;
p_x.lo(t,i) = 1e-6;
p_m.lo(t,i) = 1e-6;
p_fxv.lo = 1e-6;
p_fx.lo(t) = 1e-6;
p_l.lo(t) = 1e-6;
p_le.lo(t) = 1e-6;
p_w.lo(t) = 1e-6;
p_u.lo = 1e-6;
p_a.lo(t,i) = 1e-6;
p_g.lo(t) = 1e-6;
p_k.lo(t) = 1e-6;
p_rk.lo(t) = 1e-6;
p_rke.lo(t) = 1e-6;
p_gle.lo = 1e-6;
p_ca.lo(t) = 0;
p_ka.lo(t) = 1e-6;
p_i.lo(t) = 1e-6;

*       Income
v_inc_h.lo = 0;
v_inc_g.lo(t) = 0;
v_inc_gl.lo = 0;

*       ---------------------------------------------------------------------
*       Initial values for variables.

*       Unit demand.
a_x.l(t,i) = x0(i);
a_d.l(t,i) = d0(i);
a_l.l(t,i) = ld0(i);
a_k.l(t,i) = kd0(i);
a_e.l(t,ene,i) = veic(ene,i);
a_ec.l(t,ene) = cd0(ene);
a_c.l(t,nene) = cd0(nene);
a_cc.l(t) = c0;
a_lei.l(t) = lei0;
a_u.l(t) = wref(t);
a_ad.l(t,i) = d0(i);
a_am.l(t,i) = m0(i);

*       Activity.
q_q.l(t,i) = qref(t);
q_a.l(t,i) = qref(t);
q_x.l(t,i) = qref(t);
q_m.l(t,i) = qref(t);
q_cc.l(t) = qref(t);
q_w.l(t) = qref(t);
q_g.l(t) = qref(t);
q_j.l(t) = qref(t);
q_sl.l(t) = qref(t);
q_u.l = 1;
```

```
q_k.l(t) = qref(t);
q_gle.l = 1;
q_cnc.l(t,e) = qref(t);
q_tcap.l = sum(tlast, qref(tlast)) * (1+gr);

*       Unit cost.
c_cc.l(t) = 1;
c_a.l(t,i) = 1;
c_w.l(t) = 1;
c_u.l = 1;
c_q.l(t,i) = 1;
c_e.l(t,i) = 1;

*       Price.
p_q.l(t,i) = 1;
p_f.l(t,i) = 1;
p_ee.l(t,i) = 1;
p_fe.l(t,i) = 1;
p_c.l(t) = 1;
p_ec.l(t) = 1;
p_e.l(t,e) = 1;
p_cc.l(t) = 1;
p_d.l(t,i) = 1;
p_x.l(t,i) = 1;
p_m.l(t,i) = 1;
p_fxv.l = 1;
p_fx.l(t) = 1;
p_l.l(t) = 1;
p_le.l(t) = 1;
p_u.l = 1;
p_a.l(t,i) = 1;
p_g.l(t) = 1;
p_k.l(t) = 1;
p_rk.l(t) = 1;
p_rke.l(t) = 1;
p_gle.l = 1;
p_ca.l(t) = 0;
p_ka.l(t) = 1;
p_i.l(t) = 1;
p_w.l(t) = pref(t);

*       Income.
v_inc_h.l = ubar;
v_inc_g.l(t) = gqref(t);
v_inc_gl.l = gbar;

*       Fix capital stock in the initial period.
q_k.fx(tfirst) = 1;

*       Tax multiplier.
m_tau.l = 0;
m_tau.lo = -inf;
m_tau.up = +inf;
display m_tau.l;

*       Numeraire (fix household lifetime income)
v_inc_h.fx = v_inc_h.l;
display v_inc_h.l;
```

```
*       Interest rate.
p_intr.lo(t) = 1e-6;
p_intr.l(t) = 1 / (1 + r);
display p_intr.l;

*       Discount factor.
p_disc.lo(t) = 1e-6;
p_disc.l(t) =
    (prod(tt$((not tfirst(tt)) and (ord(tt) le ord(t))),
    p_intr.l(tt)))$(not tfirst(t))
    +
    1$tfirst(t);
display p_disc.l;

$ontext
*       ---------------------------------------------------------------
*       Benchmark replication with MCP
display "com: Benchmark replication with MCP";
dynamic_mcp.workspace = 300;
dynamic_mcp.iterlim = 0;
dynamic_mcp.optfile=1;
solve dynamic_mcp using mcp;

*       ---------------------------------------------------------------
*       Cleanup calculation with MCP
display "com: Cleanup calculation with MCP";
dynamic_mcp.iterlim = 9000;
solve dynamic_mcp using mcp;
$offtext

* --------------------
* Local Variables:
* mode: gams
* fill-column: 78
* End:
```