

Homework 5 Solution: Static Spatial Price Equilibrium

Computational Economic Equilibrium Analysis
Spring Semester, 2008

Thomas F. Rutherford
Department of Management, Technology and Economics
ETH Zürich
trutherford@ethz.ch

April 25, 2008

```
1 $title Aggregation program for homework 5.
2
3 set      s      Regions in aggregated model/
4         CHE    Switzerland
5         DEU    Germany
6         FRA    France
7         JPN    Japan
8         ITA    Italy
9         GBR    United Kingdom
10        BEL    Belgium
11        ESP    Spain
12        USA    United States of America
13        NLD    Netherlands
14        SWE    Sweden
15
16        ROW    Rest of world /;
17
18
19 $onecho >gdxxrw.rsp
20 trace=3 log=marketdata.log
21 set=r   rng="Regions!a1" rdim=1 cdim=0 values=string
22 par=x0  rng="Trade!a1"  rdim=1 cdim=1
23 par=c0  rng="Margins!a1" rdim=1 cdim=1
24 par=t0  rng="Taxes!a1"  rdim=1 cdim=1
25 $offecho
26 $if not exist marketdata.gdx $call 'gdxxrw i=marketdata.xls o=marketdata.gdx @gdxxrw.rsp'
27
28 set          r(*)      Regions in the full dataset;
29 $gdxin 'marketdata.gdx'
30 $load r
31
32 *          Verify that all regions in S except "row" are in R:
33
```

```

34 set      missing(*)      Regions in S which are not recognized;
35 missing(s) = yes;
36 missing(r) = no;
37 missing("row") = no;
38 abort$card(missing) "Region identifier is not recognized:",missing;
39
40 parameter      x0(r,r) Trade flows
41                c0(r,r) Margins
42                t0(r,r) Taxes;
43 $load x0 c0 t0
44
45 set map(r,s)    Mapping from r to s;
46 map(r,s) = yes$sameas(r,s);
47 map(r,"row")$(not sum(map(r,s),1)) = yes;
48
49 alias (s,ss), (r,rr);
50
51 set      rmap(r,s,rr,ss) Mapping of pairs;
52 rmap(r,s,rr,ss) = yes$(map(r,s) and map(rr,ss));
53
54 *      Calculate the gross value of import demand in each region:
55
56 parameter      t0_(s,ss)      Taxes
57                c0_(s,ss)      Margins
58                x0_(s,ss)      Trade flows;
59
60 x0_(s,ss) = sum(rmap(r,s,rr,ss),x0(r,rr));
61 t0_(s,ss) = sum(rmap(r,s,rr,ss),t0(r,rr));
62 c0_(s,ss) = sum(rmap(r,s,rr,ss),c0(r,rr));
63
64 *      Compute transport and tax rates:
65
66 execute_unload 'aggr.gdx', s=r, x0_=x0, c0_=c0, t0_=t0;

```

The log file provided by the GDXXRW command which appears on line 20 appears as follows:

```

GDXXRW      Dec 24, 2007 WIN.FR.NA 22.6 003.000.000.vis Delphi
Excel version 11.0
Input file : f:\CompEcon\hw5\marketdata.xls
Output file: f:\CompEcon\hw5\marketdata.gdx
Type Symbol      Dim      Sheet      Data      RowHeader  ColHeader
Set  r            1 str Regions      B1:B87     A1:A87     --:--
Par  x0           2      Trade      B2:CK81    A2:A81     B1:CK1
Par  c0           2      Margins     B2:CK81    A2:A81     B1:CK1
Par  t0           2      Taxes       B2:CK81    A2:A81     B1:CK1
Total time = 3406 Ms

```

This confirms that all the data have been processed.

The same steps can be processed when the data are provided in CSV files. These must be prepared with use of the dummy field to indicate the data layout. For example, `trade.csv` might appear as follows:

```

dummy, AUS, NZL, XOC, CHN, HKG, JPN, KOR, TWN, XEA, IDN, MYS, PHL, SGP, ...
AUS, , 242.72, 42.40, 7.21, 5.72, 67.56, 166.29, 8.72, 0.64, 64.16, ...
NZL, 22.63, , 4.40, 0.43, 0.56, 1.67, 0.46, 0.06, 0.02, 0.54, 0.22, ...

```

The top-end of the aggregation program would then appear as:

```
1 $title Aggregation program for homework 5.
2
3 set      s          Regions in aggregated model/
4         CHE        Switzerland
5         DEU        Germany
6         FRA        France
7         JPN        Japan
8         ITA        Italy
9         GBR        United Kingdom
10        BEL        Belgium
11        ESP        Spain
12        USA        United States of America
13        NLD        Netherlands
14        SWE        Sweden
15
16        ROW        Rest of world /;
17
18 set      r          Regions in the full dataset /
19 $include r.gms
20 /;
21
22 table x0(*,*) Bilateral trade flows
23 $ondelim
24 $include trade.csv
25 $offdelim
26 ;
27 table c0(*,*) Margins
28 $ondelim
29 $include margins.csv
30 $offdelim
31 ;
32 table t0(*,*) Taxes
33 $ondelim
34 $include taxes.csv
35 $offdelim
36 ;
37 ...
```

The main program then appears as follows:

```
$title Computational Economics Homework 5 : Spatial Price Equilibrium

*      Read the aggregate data:

set      r(*)      Regions in the model;
$gdxin 'aggr.gdx'
$load r

alias (r,rr);

parameter      x0(r,rr)      Benchmark trade flows
               c0(r,rr)      Trade cost
```

```

                t0(r,rr)      Tax cost;

$load x0 c0 t0

parameter      s0(r)      Base year supply
               d0(r)      Base year demand
               c(r,rr)    Unit transport cost
               t(r,rr)    Unit tax wedge;

*      Omit intra-regional trade flows:

x0(r,r) = 0;

*      Convert tax rates:

t(r,rr)$x0(r,rr) = t0(r,rr)/x0(r,rr);
c(r,rr)$x0(r,rr) = c0(r,rr)/x0(r,rr);
display t, c;

*      Benchmark supply and demand:

s0(r) = sum(rr, x0(r,rr));
d0(r) = sum(rr, x0(rr,r));

variables      X(r,rr)    Bilateral trade flows,
               S(r)      Supply
               D(r)      Demand
               COST      Total cost of shipments;

positive variable X;

equations      costdef      Defines total cost,
               supply(r)    Regional supply,
               demand(r)    Regional demand;

costdef..      COST =e= sum((r,rr), X(r,rr)*(c(r,rr)+t(r,rr)));

supply(r)..    S(r) =e= sum(rr, X(r,rr));

demand(r)..    sum(rr, X(rr,r)) =e= D(r);

X.FX(r,r) = 0;
S.FX(r) = s0(r);
D.FX(r) = d0(r);

option lp=bdmlp;

model calib /all/;
solve calib using lp minimizing cost;

display x0, X.L;

*      Use the LP solution to impute the base year price.
*      Calibrate the demand price in CHE to unity:

```

```

parameter      ps0(r)      Reference supply price
                pd0(r)      Reference demand price
                xref(r,r)    Reference trade flow;

ps0(r) = 1 + supply.m(r) - supply.m("usa");
pd0(r) = 1 + demand.m(r) - supply.m("usa");
xref(r,rr) = X.L(r,rr);
display ps0, pd0;

parameter      eta(r)      Price elasticity of export supply
                epsilon(r)  Price elasticity of import demand;

eta(r) = 4;
epsilon(r) = 2;

variables      PS(r)      Supply price
                PD(r)      Demand price;

PS.L(r) = ps0(r);
PD.L(r) = pd0(r);

equations      profit(r,rr)  Bilateral profit
                lsdef(r)     Supply function (linear)
                lddef(r)     Demand function (linear)
                isdef(r)     Supply function (iso-elastic)
                iddef(r)     Demand function (iso-elastic);

profit(r,rr).. PS(r) + c(r,rr) + t(r,rr) =g= PD(rr);

lsdef(r)..     S(r) =e= s0(r) * (1 + eta(r) * (PS(r)/ps0(r)-1));

lddef(r)..     D(r) =e= d0(r) * (1 - epsilon(r) * (PD(r)/pd0(r)-1));

isdef(r)..     S(r) =e= s0(r) * (PS(r)/ps0(r))**eta(r);

iddef(r)..     D(r) =e= d0(r) * (PD(r)/pd0(r))**(-epsilon(r));

model lequil /lsdef.S, lddef.D, supply.ps, demand.pd, profit.x/;
model iequil /isdef.S, iddef.D, supply.ps, demand.pd, profit.x/;

S.L0(r) = -inf; S.UP(r) = +inf; S.M(r) = 0;
D.L0(r) = -inf; D.UP(r) = +inf; D.M(r) = 0;

lequil.iterlim = 0;
iequil.iterlim = 0;
solve lequil using mcp;
solve iequil using mcp;

variable      Z          Social surplus;

equation      lsurplus    Defines social surplus (linear)

```

```

                isurplus          Defines social surplus (isoelastic);

lsurplus..      Z =e= - (sum(r,
                pd0(r)*D(r) * (1 - (1/epsilon(r)) * (D(r)/(2*d0(r))-1))
                - ps0(r)*S(r) * (1 + (1/eta(r))      * (S(r)/(2*s0(r))-1)))
                - sum((r,rr),X(r,rr)*(c(r,rr)+t(r,rr))));

supply.m(r) = ps0(r);
demand.m(r) = pd0(r);
lsurplus.m = 1;

Z.L = - (sum(r, pd0(r)*D.L(r) * (1 - (1/epsilon(r)) * (D.L(r)/(2*d0(r))-1))
        - ps0(r)*S.L(r) * (1 + (1/eta(r))      * (S.L(r)/(2*s0(r))-1)))
        - sum((r,rr),X.L(r,rr)*(c(r,rr)+t(r,rr))));

model qpequil /lsurplus, supply, demand /;

option nlp = examiner;
solve qpequil using nlp minimizing Z;

isurplus..      Z =e= - (sum(r,
                pd0(r)*D(r) * (D(r)/d0(r))**(-1/epsilon(r)) / (1-1/epsilon(r))
                - ps0(r)*S(r) * (S(r)/s0(r))** (1/eta(r)) / (1 + 1/eta(r)))
                - sum((r,rr),X(r,rr)*(c(r,rr)+t(r,rr))));

Z.L = - (sum(r,
                pd0(r)*D.L(r) * (D.L(r)/d0(r))**(-1/epsilon(r)) / (1-1/epsilon(r))
                - ps0(r)*S.L(r) * (S.L(r)/s0(r))** (1/eta(r)) / (1 + 1/eta(r)))
                - sum((r,rr),X.L(r,rr)*(c(r,rr)+t(r,rr))));
isurplus.m = 1;

model nlpequil /isurplus, supply, demand /;
solve nlpequil using nlp minimizing Z;

option nlp = conopt;

parameter      bmdata Benchmark data;
bmdata(r,"tx") = sum(rr, X.L(r,rr)*t(r,rr))/s0(r);
bmdata(r,"tm") = sum(rr, X.L(rr,r)*t(rr,r))/d0(r);

*      Free trade equilibrium:

t(r,rr) = 0;

lequil.iterlim = 10000;
solve lequil using mcp;

parameter      impact Equilibrium impact;
impact("Linear",r,"PD") = 100 * (PD.L(r)/pd0(r)-1);
impact("Linear",r,"PS") = 100 * (PS.L(r)/ps0(r)-1);

solve qpequil using nlp minimizing Z;
impact("Linear",r,"PD_QP") = 100 * (demand.m(r)/pd0(r)-1);
impact("Linear",r,"PS_QP") = 100 * (supply.m(r)/ps0(r)-1);

```

```

iequil.iterlim = 10000;
solve iequil using mcp;
impact("Isoelastic",r,"PD") = 100 * (PD.L(r)/pd0(r)-1);
impact("Isoelastic",r,"PS") = 100 * (PS.L(r)/ps0(r)-1);

solve nlpequil using nlp minimizing Z;
impact("Isoelastic",r,"PD_NLP") = 100 * (demand.m(r)/pd0(r)-1);
impact("Isoelastic",r,"PS_NLP") = 100 * (supply.m(r)/ps0(r)-1);
option impact:5:1:1;
display impact;

bmdata(r,"%s") = 100 * (S.L(r)/s0(r)-1);
bmdata(r,"%d") = 100 * (D.L(r)/d0(r)-1);
display bmdata;

```

The key program output are as follows:

```

----      185 PARAMETER impact  Equilibrium impact

INDEX 1 = Linear

          PD          PS          PD_QP          PS_QP
CHE      4.06221      -0.31815      4.06221      -0.31815
DEU      0.43424       0.41339      0.43424      0.41339
FRA      0.40726       0.42517      0.40726      0.42517
JPN      0.92940       5.46514      0.92940      5.46514
ITA      0.40817      -1.64741      0.40817     -1.64741
GBR      0.40816       0.44020      0.40816      0.44020
BEL      0.40766     -1.66050      0.40766     -1.66050
ESP      0.40777     -2.77699      0.40777     -2.77699
USA      2.51973     -6.70170      2.51973     -6.70170
NLD      0.42045       0.41480      0.42045       0.41480
SWE      0.40830       0.41305      0.40830       0.41305
ROW     -8.57803       2.94846     -8.57803       2.94846

INDEX 1 = Isoelastic

          PD          PS          PD_NLP          PS_NLP
CHE      4.06226     -0.31810      4.06226     -0.31810
DEU      0.43430       0.41345      0.43430       0.41345
FRA      0.40731       0.42522      0.40731       0.42522
JPN      0.92945       5.46519      0.92945       5.46519
ITA      0.40822     -1.64735      0.40822     -1.64735
GBR      0.40822       0.44027      0.40822       0.44027
BEL      0.40772     -1.66043      0.40772     -1.66043
ESP      0.40782     -2.77694      0.40782     -2.77694
USA      2.51978     -6.70165      2.51978     -6.70165
NLD      0.42050       0.41486      0.42050       0.41486
SWE      0.40835       0.41311      0.40835       0.41311
ROW     -8.57799       2.94851     -8.57799       2.94851

```

```

---- 189 PARAMETER bmdata Benchmark data

      tx      tm      %s      %d
CHE   -0.012   0.011   -1.266   -7.655
DEU           0.003    1.664   -0.863
FRA    0.056           1.712   -0.810
JPN    0.024   -0.002   23.719   -1.833
ITA    0.071           -6.428   -0.811
GBR           1.773   -0.811
BEL    0.073           -6.478   -0.810
ESP    0.063           -10.654  -0.811
USA    0.022   0.014  -24.230  -4.855
NLD           1.670   -0.836
SWE           1.663   -0.812
ROW    0.003   0.053   12.326   19.646

```