A Synthetic Gas Network

October 18, 2004

1 Geography

We consider a gas network in which supply nodes are randomly distributed on a circle of radius one. Markets in which gas is sold are then randomly distributed within the square area enscribed by this circle. There is a subtle challenge involved in producing an "interesting" prototypical pipleine network. The network needs to be constructed in such a way to assure feasibility of supply and a reasonable level of interconnectedness between markets. Linear programming provides a convenient way to construct such a network.

2 Calibration

The network structure is calibrated to assure feasible flows using "constructed transport costs" which generate geographically distributed network flows within the demand region and relatively few long-distance flows. Transport costs are constructed as:

$$c_{ij} = D_{ij}^2 \left(1 - \frac{\delta_{ij}}{2}\right)$$

where δ_{ij} takes on values of 1 or 0 with $\delta_{ij} = 1$ representing a randomly generated set of arcs between points separated by a distance of less than 0.5.

The synthetic transport costs are used to solve for a feasible network flows in the following linear program:

$$\max \sum_{ij} c_{ij} x_{ij} + \sum_{jk} c_{jk} y_{jk}$$

subject to:

$$\sum_i x_{ij} + \sum_k y_{kj} = \bar{d}_j + \sum_k y_{jk}$$

$$\bar{s}_i \geq \sum_j x_{ij}$$

3 Equilibrium

We use the solution to the constructed linear program to define the supply network links, $\delta_{ij} = \{x^*_{ij} > 0\}$ and $\delta_{jk} = \{y^*_{jk} > 0\}$. Having determined the network structure we then compute a spatial price equilibrium in which transport costs t_{ij} are defined with respect to "crow-flies distance" rather than the constructed transport costs, c_{ij}:

1. Market clearance at supply nodes and markets:

$$s_i = \sum_{j|\delta_{ij}} x_{ij}$$

$$\sum_{i|\delta_{ij}} x_{ij} + \sum_{k|\delta_{kj}} y_{kj} = \bar{d}_j + \sum_{k|\delta_{jk}} y_{jk}$$

2. Calibrated demand and supply:

$$d_j = \bar{d}_j p_j^{\epsilon_j}$$

$$s_i = \bar{s}_j w_i^{\eta_i}$$

3. Transport arbitrage:

$$w_i + t_{ij} \geq p_j$$

$$p_j + t_{jk} \geq p_k$$

Figure 1: Network Flows in the Benchmark Linear Program

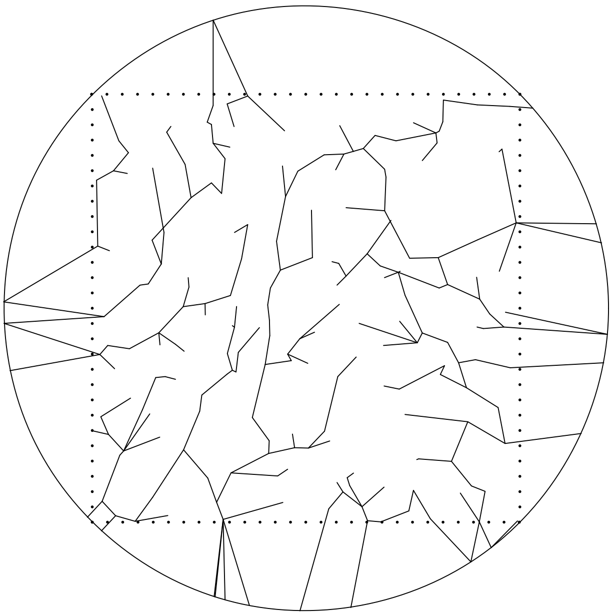
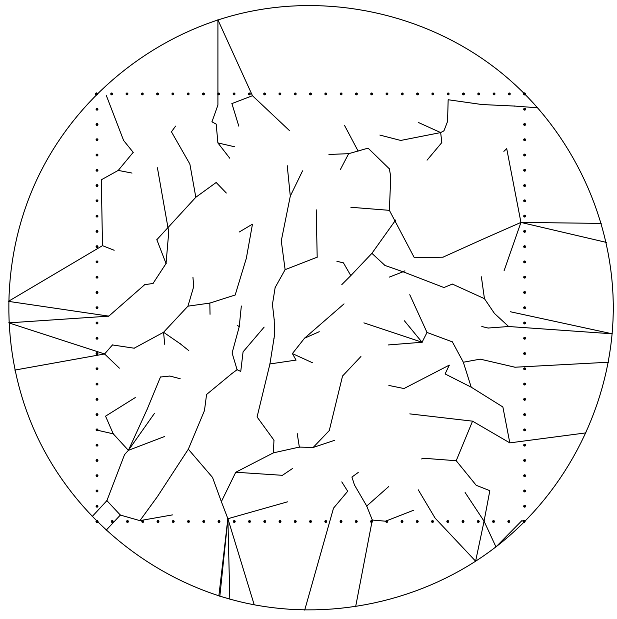


Figure 2: Equilibrium Flows on the Constructed Network



```
$title  A Random Gas Network Model

*       A network is defined by nodes and connecting arcs:

set     i       Gas supply nodes /i1*i10/,
        j       Gas markets /j1*j200/,
        ij(i,j) Supply-demand links,
        jk(j,j) Demand-demand links;

alias (j,k);

*       The input data are obtained in the form of reference
*       price-quantity pairs for all supply and demand functions:

parameter       s0(i)           Reference supply,
                d0(j)           Reference demand,
                epsilon(j)      Elasticity of demand
                eta(i)          Elasticity of supply
                utc(*,*)        Transport costs,
                theta(i)        Location for supplier,
                lon(*)          Longitude,
                lat(*)          Latitude;

*       Generate some random data:

eta(i) = uniform(0,2);
epsilon(j) = uniform(-2,-0.2);

*       Randomly distribute supply points on a circle of radius one
*       centered at (1,1):

theta(i) = uniform(0,2) * 3.141592;
lon(i) = 1 + cos(theta(i));
lat(i) = 1 + sin(theta(i));

*       Place demand nodes within the enscribed square:

lon(j) = 1-sqrt(1/2) + uniform(0,2*sqrt(1/2));
lat(j) = 1-sqrt(1/2) + uniform(0,2*sqrt(1/2));

*       Define the "crow-flies distances":

utc(i,j) = sqrt( sqr(lon(i)-lon(j)) + sqr(lat(i)-lat(j)) );
utc(k,j) = sqrt( sqr(lon(k)-lon(j)) + sqr(lat(k)-lat(j)) );

*       Generate benchmark supplies and demands, leaving a 5%
*       margin to assure sufficient supply to match demand:

s0(i) = uniform(1,5);
d0(j) = uniform(1,5);
d0(j) = d0(j) * sum(i,s0(i))/sum(k,d0(k)) * 0.95;

*       Define a random "target network" of flows, randomly
*       selecting arcs from pairs of points which are separated
*       by a distance of 0.25 or less:

ij(i,j)$(utc(i,j) < 0.25) = yes$(uniform(0,1) > 0.25);
```

```
jk(j,k)$(utc(j,k) < 0.25) = yes$(uniform(0,1) > 0.25);
jk(j,j) = no;

*       If utc() is used in the calibrate LP we produce a network
*       with too many long arcs.  Using square of distance assures
*       that the resulting network has shorter distances on each
*       link.  The randomly ij() and jk() networks are used to
*       target a randomly generated network:

parameter       c(*,j)  Synthetic transport cost;

c(i,j) = utc(i,j)**2 * (1 - 0.5$ij(i,j));
c(j,k) = utc(j,k)**2 * (1 - 0.5$jk(j,k));

*       Generate flows in a LP which is always going to be feasible:

variables       x(i,j)          Supply from nodes i to market j
                y(j,k)          Transhipment from market j to market k
                tcost           Transport cost;

positive variables x,y;

equations       cost            Transport cost definition,
                demand(j)       Demand at node j,
                supply(i)       Supply at node i;

cost..          tcost =e= sum((i,j), c(i,j) * x(i,j)) + sum((j,k), c(j,k) * y(j,k));

demand(j)..     sum(i, x(i,j)) + sum(k, y(k,j)) =e= d0(j) + sum(k, y(j,k));

supply(i)..     s0(i) =g= sum(j, x(i,j));

*       Avoid degeneracy by preventing flows from a node to itself:

y.fx(j,j) = 0;
model gtmcalib /all/;
solve gtmcalib using nlp minimizing tcost;

*       Generate a LaTeX file with the network geometry:

$onecho >network.tex
\documentclass[epic,eepic]{article}
\usepackage{epic,eepic}
\begin{document}
\begin{figure}[p]
\caption{Network Flows in the Benchmark Linear Program}
\begin{center}
\setlength{\unitlength}{4cm}
\begin{picture}(2.5,2.5)
$offecho
file ktex /network.tex/; put ktex; ktex.nw=0; ktex.nd=3; ktex.ap=1;
put '\put(1,1){\circle{2}}'/;
put '\dottedline[.]{0.05}(',(1-sqrt(1/2)),',',(1-sqrt(1/2)),')(',
                  (1+sqrt(1/2)),',',(1-sqrt(1/2)),')(',
                  (1+sqrt(1/2)),',',(1+sqrt(1/2)),')(',
                  (1-sqrt(1/2)),',',(1+sqrt(1/2)),')(',
                  (1-sqrt(1/2)),',',(1-sqrt(1/2)),')'/;
```

```
loop((i,j)$x.l(i,j),put '\drawline(',lon(i),',',lat(i),')(',lon(j),',',lat(j),')'/;);
loop((j,k)$y.l(j,k),put '\drawline(',lon(j),',',lat(j),')(',lon(k),',',lat(k),')'/;);
put '\end{picture}'/;
put '\end{center}'/;
put '\end{figure}'/;

*       Define the gas network based on the LP solution:

ij(i,j) = yes$x.l(i,j);
jk(j,k) = yes$y.l(j,k);

*       Create a randomly connected network network:

positive variables      d(j)    Market demand,
                        p(j)    Market price,
                        w(i)    Supply price (marginal cost);

equations               db, dd, pmc, xprofit, yprofit;

db(j)..                 sum(i$ij(i,j), x(i,j)) + sum(k$jk(k,j), y(k,j))

                                 =g= d(j) + sum(k$jk(j,k), y(j,k));

dd(j)..                 d(j) =e= d0(j) * p(j)**epsilon(j);

pmc(i)..                s0(i) * w(i)**eta(i) =g= sum(j$ij(i,j), x(i,j));

xprofit(ij(i,j))..      w(i) + utc(i,j) =g= p(j);

yprofit(jk(j,k))..      p(j) + utc(j,k) =g= p(k);

*       Avoid bad function calls:

p.lo(j) = 0.01; w.lo(i) = 0.01;
p.l(j) = 1;     w.l(i) = 1;
y.fx(j,j) = 0;

*       Define the complementarity problem here:

model random /db.p, dd.d, pmc.w, xprofit.x, yprofit.y/;

solve random using mcp;


*       Generate a second network graph based on the resulting
*       equilibrium flows:

put ktex;
put     '\begin{figure}[p]'/
        '\caption{Equilibrium Flows on the Constructed Network}'/
        '\begin{center}'/
        '\setlength{\unitlength}{4cm} '/
        '\begin{picture}(2.5,2.5)'/;
put '\put(1,1){\circle{2}}'/;
put '\dottedline[.]{0.05}(',(1-sqrt(1/2)),',',(1-sqrt(1/2)),')(',
                   (1+sqrt(1/2)),',',(1-sqrt(1/2)),')(',
                   (1+sqrt(1/2)),',',(1+sqrt(1/2)),')(',
```

```
                  (1-sqrt(1/2)),',',(1+sqrt(1/2)),')(',
                  (1-sqrt(1/2)),',',(1-sqrt(1/2)),')'/;
loop((i,j)$x.l(i,j), put '\drawline(',lon(i),',',lat(i),')(',lon(j),',',lat(j),')'/;);
loop((j,k)$y.l(j,k), put '\drawline(',lon(j),',',lat(j),')(',lon(k),',',lat(k),')'/;);
put '\end{picture}'/;
put '\end{center}'/;
put '\end{figure}'/;

y.up(j,k) = uniform(0.8,1) * y.l(j,k);
solve random using mcp;

put     '\begin{figure}[p]'/
        '\caption{Equilibrium Flows on the Constrained Network}'/
        '\begin{center}'/
        '\setlength{\unitlength}{4cm} '/
        '\begin{picture}(2.5,2.5)'/;
put '\put(1,1){\circle{2}}'/;
put '\dottedline[.]{0.05}(',(1-sqrt(1/2)),',',(1-sqrt(1/2)),')(',
                  (1+sqrt(1/2)),',',(1-sqrt(1/2)),')(',
                  (1+sqrt(1/2)),',',(1+sqrt(1/2)),')(',
                  (1-sqrt(1/2)),',',(1+sqrt(1/2)),')(',
                  (1-sqrt(1/2)),',',(1-sqrt(1/2)),')'/;
loop((i,j)$x.l(i,j), put '\drawline(',lon(i),',',lat(i),')(',lon(j),',',lat(j),')'/;);
loop((j,k)$y.l(j,k), put '\drawline(',lon(j),',',lat(j),')(',lon(k),',',lat(k),')'/;);
put '\end{picture}'/;
put '\end{center}'/;
put '\end{figure}'/;


put '\end{document}'/;
putclose;
```

```
$title An International Gas Trade Model (GTM,SEQ=53) -- MCP Formulation

$Ontext

   The Gas Trade Model (GTM) models interrelated gas markets.
   Prices may be free to move as to equilibriate supplies and
   demand. Disequilibria can be introduced with controls over
   prices and/or quantities traded.


Manne, A S, and Beltramo, M A, GTM: An International Gas Trade Model ,
International Energy Program Report. Stanford University, 1984.

$Offtext

Sets
        i       Supply Regions
                / mexico, alberta-bc, atlantic, appalacia, us-gulf,
                mid-cont, permian-b, rockies, pacific, alaska  /
        j       Demand regions
                / mexico, west-can, ont-quebec, atlantic, new-engl,
                ny-nj, mid-atl, south-atl, midwest,
                south-west, central, n-central, west, n-west /

        fx(j)   Regions with fixed demand
                / mexico, west-can, ont-quebec, atlantic /

        ij(i,j) Feasible trade links;


Table  sdat(i,*)  supply data

             ref-p1   ref-q1   ref-p2   ref-q2   limit
*           ($/mcf)   (tcf)   ($/mcf)   (tcf)    (tcf)
 mexico                2.0                .5     2.5
 alberta-bc            3.0               1.6     3.75
 atlantic               .25               .03     .3
 appalacia    3.5       .58      7.0      .65     .72
 us-gulf      3.5      7.88      7.0     8.82    9.75
 mid-cont     3.5      2.07      7.0     2.31    2.55
 permian-b    3.5      1.39      7.0     1.55    1.72
 rockies      3.5      1.16      7.0     1.30    1.44
 pacific      3.5       .42      7.0      .47     .52
 alaska       2.0       .80      2.0      .1      inf


Table ddat(j,*)  demand data

             ref-p   ref-q   elas   tax     ex-dem
*           ($/mcf)  (tcf)          ($/mcf)  (tcf)
 mexico      1.0     2.2    -.5
 west-can    3.0     1.47   -.5
 ont-quebec  3.5     1.38   -.5
 atlantic    3.5      .20   -.5
 new-engl    9.37     .76   -.60
 ny-nj       8.33    1.18   -.66
 mid-atl     8.26     .89   -.65
```

```
 south-atl   8.07    1.62   -.89
 midwest     8.01    2.96   -.65
 south-west  7.29    6.04   -.84
 central     7.79    1.17   -.67
 n-central   8.06    1.51   -.54
 west        8.18    2.10   -.43
 n-west      9.39     .36   -.57

parameters      d0(j)           Reference demand,
                p0(j)           Reference price,
                epsilon(j)      Elasticity of demand;

d0(j) = ddat(j,"ref-q");
p0(j) = ddat(j,"ref-p");
epsilon(j) = ddat(j,"elas");

 Parameters  supa(i)  supply constant a
             supb(i)  supply constant b
             supc(i)  supply capacity ;

  supc(i)  =  sdat(i,"limit");
  supb(i)  =  ((sdat(i,"ref-p1")-sdat(i,"ref-p2"))
                /(1/(supc(i)-sdat(i,"ref-q1"))-1/(supc(i)-sdat(i,"ref-q2"))))
             $(supc(i) ne inf);
  supa(i)  =  sdat(i,"ref-p1") - supb(i)/(supc(i)-sdat(i,"ref-q1"));

* We rely on supa(i) evaluating to exactly zero in some cases

  supa(i)  =  round(supa(i),4);
  supc(i)$(supc(i) eq inf) = 100;
  sdat(i,"sup-a") = supa(i); sdat(i,"sup-b") = supb(i); display sdat;

 Table  utc(i,j)  unit transport cost  ($ per mcf)

                mexico    west-can  ont-quebec    new-engl       ny-nj
mexico            0.25                                            2.29
atlantic                                              1.50
alberta-bc                    0.40        0.90        1.15        1.10
us-gulf                                               2.12        1.08

         +     mid-atl   south-atl     midwest  south-west     central

mexico            2.22        2.03        1.96        1.25
alberta-bc        1.10        1.55        0.80        1.25        0.80
appalacia         0.72                    0.46
us-gulf           1.01        0.82        0.75        0.04        0.54
mid-cont                                  0.86        0.14        0.64
permian-b                     0.83        0.77        0.05        0.55
rockies                                   0.53                    0.31
alaska                                    6.00

         +   n-central        west      n-west

mexico                        2.13
alberta-bc        0.65        0.70        0.65
permian-b                     0.94
rockies           0.58        0.70        1.91
```

```
pacific                       0.43

Table  pc(i,j)   pipeline capacities (tcf)

             mexico  west-can  ont-quebec  atlantic  new-engl  ny-nj  mid-atl  south-atl  midwest  south-west
 mexico        inf                                              .067   .067     .067       .067
 alberta-bc            inf       inf                   .30      .150   .10                  inf
 atlantic                                    inf        inf
 appalacia                                                             .34                 .35
 us-gulf                                                inf    1.390  1.060    2.0        2.62       3.73
 mid-cont                                                                                  .62       2.30
 permian-b                                                                                 .12       1.45
 rockies                                                                                   .48
 alaska                                                                                    .80

 +           central  n-central  west  n-west
 mexico                          .033
 alberta-bc    inf       inf      inf    inf
 mid-cont    1.03
 permian-b                      1.46
 rockies      .14        inf     .10     inf
 pacific                         .48

SETS    check1(i,j)  Supply links with zero cost and non-zero capacity
        check2(i,j)  Supply links with nonzero cost but zero capacity ;

check1(i,j) = yes$(utc(i,j) eq 0 and pc(i,j) ne 0);
check2(i,j) = yes$(utc(i,j) ne 0 and pc(i,j) eq 0);

DISPLAY check1, check2;

*       Create a tuple defining the feasible links:

ij(i,j) = yes$pc(i,j);

Variables
        x(i,j)  shipment of natural gas (tcf)
        s(i)    regional supply         (tcf)
        d(j)    regional demand         (tcf)
        w(i)    Supply price ($ per mcf)
        p(j)    Demand price ($ per mcf);

Positive Variables x, s, d, p, w;

Equations sb(i)         Supply balance          (tcf)
          db(j)         Demand balance          (tcf)
          dd(j)         Demand function defined
          pmc(i)        Price-marginal cost
          profit(i,j)   Transportation profit;

sb(i)..   s(i) =g= sum(j$ij(i,j), x(i,j));

db(j)..   sum(i$ij(i,j), x(i,j)) =g= d(j);

dd(j)..   d(j) =e= d0(j) * (p(j)/p0(j))**epsilon(j);

pmc(i)..  supa(i) + supb(i)/(supc(i)-s(i)) =g= w(i);
```

```
profit(ij(i,j))..       w(i) + utc(i,j) =g= p(j);

p.l(j) = p0(j);
d.l(j) = d0(j);
d.fx(j)$fx(j) = d0(j);
w.l(i) = 1;
s.l(i) = sdat(i,"ref-q1");

p.lo(j) = 0.01;
s.up(i) = supc(i) - 0.0001;

model gtmmcp Gas trade model / sb.w, db.p, dd.d, pmc.s, profit.x /;
solve gtmmcp using mcp;


parameter echoprint     Report of values;

echoprint(i,"s") = s.l(i);
echoprint(i,"supa") = supb(i);
echoprint(i,"supb") = supb(i);
echoprint(i,"supc") = supc(i);
display echoprint;
```