ภาคผนวก

ภาคผนวก ก

การกำหนดค่าลงในโปรแกรม GAMS

```
option limrow=20;
option ITERLIM = 1000000;
option OPTCR = 0.10;
option reslim = 86400;


file thesismodel /C:\Thesisdata\thesismodel.txt/;
scalar bigM;
bigM = 999999999;


set
$onempty
dum 'dummy site' / i0 /
s 'source counties' /i1*i76/
i 'potential nodes of sites' /i1*i110/
is(i) 'potential source sites' /i1*i76/
ic(i) 'potential collection sites' /i77*i94/
ip(i) 'potential processing sites' /i95*i104/
il(i) 'potential landfill sites' /i105*i110/
il1(i) 'landfill site type 1' /i105/
il2(i) 'landfill site type 2' /i106*i109/
il3(i) /i110/



c 'customers' /c1/
k 'material types' /k1/
```

```
kcol(k) 'materials collected at collection sites' /k1/
*kcust(k) 'materials shipped to landfills' /k2/
*Pm 'main process types' /m1*m2/
*Pmc(Pm) 'main process at permanent collection sites' /m1/
*Pmp(Pm) 'main process at processing sites' /m2/
*p 'sub-process types' /p1*p2/
m 'transportation type' /1/
t 'time periods' /t1*t3/;

alias(i,j),(k,q);

*read data from PlasticData.xls File
$onecho > allt.txt
i=%system.fp%PlasticDataBig1.xls
r1=pop!a2:pop!b77
o1=pop.inc
r2=popn!a2:popn!b77
o2=popn.inc
r3=rho1a!a1:rho1a!d5
o3=rho1.inc
r4=rho2a!a1:rho2a!d5
o4=rho2.inc
r5=dist!a1:dist!dg111
o5=dist.inc

$offecho

$call =xls2gms @allt.txt

*Distance Matrix
table
d(i,j)
$include dist.inc
```

```
;

parameter dist(i,j);

loop(i,
        loop(j,
                dist(i,j) = d(i,j);
        );
);

*Define Recycle Percentage
parameter b(t);
b('t1') = 0.50;
b('t2') = 0.50;
b('t3') = 0.50;
display b;

*Define amount of plastic in each year (1,2,3)
parameter plastic(t);
plastic('t1') = 1094637;
plastic('t2') = 1149368;
plastic('t3') = 1206837;

*Fixed openning cost for ic(collection), ip(processing), il(landfill) sites
parameter Fjopen(i,t) 'fixed cost to openning sites';
Fjopen(ic,t)= 100000000;
Fjopen(ip,t)= 1950000000;
Fjopen(il,t)= 12000000;

display Fjopen;

*Fixed operating cost for ic(collection), ip(processing), il(landfill) sites
```

```
parameter Fop(i,t) 'fixed cost to operate sites =office/working space leasing,forklift and driver,Fop(i,t)';
*assume = 1% from openning cost*
Fop(ic,t)= 1000000;
Fop(ip,t)= 19500000;
Fop(il,t)= 120000;

display Fop;
*Fixed operating cost for ic(collection), ip(processing), il(landfill) sites
parameter Fjopen(i,t);
*assume = 1% from openning cost*
Fjopen(ic,t)= 1000000;
Fjopen(ip,t)= 19500000;
Fjopen(il,t)= 120000;

*Fixed closing cost for ic(collection), ip(processing), il(landfill) sites
*assume to be zero
parameter Fjclose(i,t);
Fjclose(ic,t)= 0;
Fjclose(ip,t)= 0;
Fjclose(il,t)= 0;

parameter Fco(ic,t) 'fixed cost per worker = hire a warehouse worker,Fco(icp,t)';
*data from minimum laber benefit =  206/day
Fco(ic,t)=75190;
display Fco;

parameter Ftrs(i,j,m,t) 'fixed cost per vehicle = buy a truck&trailer + hire a driver,Ftrs(i,j,m,t)';
*Truck 3,500,000 + hire driver 75190
Ftrs(i,j,m,t)=0;
Ftrs(is,il,m,t)= 3575190;
Ftrs(is,ic,m,t)= 3575190;
Ftrs(ic,ip,m,t)= 3575190;
Ftrs(ip,il,m,t)= 3575190;

display Ftrs;

parameter Ftrc(i,c,m,t) 'fixed cost per vehicle = buy a truck&trailer + hire a
driver,Ftrc(i,c,m,t)';
Ftrc(i,c,m,t)=0;
display Ftrc;

parameter Fpr(i,t) 'fixed cost per main-process(inspector/repair & disassembly
worker),Fpr(i,Pm,t)';
*collect assume 10 workers
Fpr(ic,t) = 741600;
*process  assume 30 workers
Fpr(ip,t) = 2224800;
*landfill  assume 1 workers
Fpr(il,t) = 74160;

display Fpr;

parameter Vco(ic,k,t) 'collection cost per ton of material k per time period,Vco(icp,k,t)';
***** assume =0 *****
Vco(ic,k,t)=0;
********************
display Vco;

parameter Vtrs(i,j,m,t) 'transportation cost per ton of materials per vehicle of shipping
mode m,Vtrs(i,j,m,t)';
Vtrs(is,j,m,t)= 0.34*dist(is,j);
Vtrs(ic,j,m,t)= 0.34*dist(ic,j);
Vtrs(ip,j,m,t)= 0.34*dist(ip,j);
Vtrs(i,is,m,t)= 0;
```

```
display Vtrs;

parameter Vtrc(i,c,m,t) 'transportation cost per ton of materials per vehicle of shipping
mode m,Vtrc(i,c,m,t)';
Vtrc(i,c,m,t)=0;
display Vtrc;

parameter Vpr(i, t) 'processing cost per ton of materials in sub-process p,Vpr(i,p,t)';
*assume*
Vpr(ic, t) = 200;
Vpr(ip, t) = 500;
Vpr(il, t) = 100;

display Vpr;

parameter Rco(ic,k,t) 'collection fee per ton of material k,Rco(ic,k,t)';
Rco(ic,k,t)=0;
*display Rco;

parameter
Pop1(i) 'population in county s at the first time period,Pop1(s,t)'/
$include pop.inc
/;

scalar gamma 'percent of population growth per time period'/0.03/;

parameter Pop(i,t) 'population in county s at time period t';
loop(t,
        Pop(i,t)=power(1+gamma,ord(t)-1)*Pop1(i);
);
display Pop;
```

```
parameter Popn1(i) 'population at location n'/
$include pop.inc
/;
*display popn1;
parameter Popnt(i,t)'population at location n, at time period t';
loop(t,
          Popnt(i,t)= power(1+gamma,ord(t)-1)*Popn1(i);
);
*display Popnt;
scalar dummy;
parameter sumpop(t);
parameter NSupply(i,t) 'total supply of all products at time period t';

loop (t,
        sumpop(t) = 0;
        loop(i,
                sumpop(t) = sumpop(t) + Popnt(i,t);
        );
        dummy = sumpop(t);
        NSupply(i,t)= Popnt(i,t)*plastic(t)/dummy ;
);

*display Nsupply;

parameter Sup(i,k,t)'Supply of material k, at time t, at location n';
Sup(i,'k1',t)=NSupply(i,t);

*display sup;

parameter Dem(k,c,t) 'demand of material k at customer k at time period t (in lbs per
period),Dem(k,c,t)';
Dem('k1',c,t)=10000000;
```

```
parameter Lc(i,t) 'amount of material (tons) that a collection site can receive,Lc(ic,t)';
*From capacity 500 tons/day
Lc(ic,t)=182500;

parameter Lp(i,t) 'amount of material (tons) that a processing site can receive,Lp(ip,t)';
* capacity = 2190 tons/period/machine,so use 30 machines
Lp(ip,t)=65700;



parameter Cco(i,k) 'amount of material k (tons) that a worker can handle per period
(tons per period),Cco(ic,k)';
*assume from
Cco(i,'k1')=65700;

*table
*Cpr(p,Pm) 'amount of material (lbs) that a main process, Pm, can perform process
p,Cpr(p,Pm)'
*          m1        m2
*p1         100000
*p2                   100000;
*display Cpr;

parameter Ctrs(i,k,j,m) 'amount of material k (tons) that a vehicle can ship per time
period (tons per truck per period),Ctrs(i,k,j,m)';
*capacity = 5 tons/vehicle:day
ctrs(i,k,j,m) = 1825;

parameter Ctrc(i,k,c,m) 'amount of material k (tons) that a vehicle can ship per time
period,Ctrc(i,k,c,m)';
ctrc(i,k,c,m) = 1825;
```

```
parameter CAP(i) 'Current Cap at Site i';
CAP(il1) = 5912000;
CAP(il2) = 0;
CAP(il3) = 0;

*LCAP(il1) = 5913000;
*LCAP(il2) = 2555000;



parameter muco(ic,t) 'must collect (have to hire workers to collect
materials),muco(icp,t)';
muco(ic,t)=0;
display  muco;

parameter mutrs(i,j,m,t) 'must transport between sites (have to buy/rent
vehicles),mutrs(i,j,m,t)';
mutrs(i,j,m,t)=0;
display  mutrs;

parameter mutrc(i,c,m,t) 'must transport to customers (have to buy/rent
vehicles),mutrc(i,c,m,t)';
mutrc(i,c,m,t)=0;
display  mutrc;

parameter mupr(i,t) 'must process (have to buy/rent machines to operate),mupr(i,Pm,t)';
mupr(i,t)=0;
display  mupr;

parameter aco(ic,t) 'allow to collect (can hire workers to collect materials),aco(icp,t)';
aco(ic,t)=1;
display aco;
```

```
parameter atrs(i,j,m,t) 'allow to transport between sites (can buy/rent
vehicles),atrs(i,j,m,t)';
atrs(i,j,m,t)=1;
atrs(i,i,m,t)=0;
atrs(ip,il,m,t)=0;
atrs(il,ip,m,t)=0;
atrs(is,il,m,t)=0;
atrs(il,is,m,t)=0;
atrs(is,ip,m,t)=0;
atrs(ip,is,m,t)=0;
display atrs;

parameter atrc(i,c,m,t) 'allow to transport to customers (can buy/rent
vehicles),atrc(i,c,m,t)';
atrc(ip,c,m,t)=1;
atrc(ic,c,m,t)=0;
atrc(is,c,m,t)=0;
atrc(il,c,m,t)=0;
display atrc;


parameter apr(i,t) 'allow to process (can buy/rent machines),apr(i,Pm,t)';
apr(ic,t)=1;
apr(ip,t)=1;
display apr;

parameter LCAP(i) 'Landfill Cap at Site i';
LCAP(il1) = 5913000;
LCAP(il2) = 2555000;

**********
* MODEL **
**********
```

```
variable
TCost;

positive variable
xtrs(i,k,j,m,t) 'amount of material shipped between sites(i to j or j to i), xtrs(i,k,j,m,t)'
xtrc(i,k,c,m,t) 'amount of material shipped to customers(i to c), xtrc(i,k,c,m,t)'
xprm(i,t) 'the amount of material processed by main process Pm, sub-process p,
xprm(i,p,Pm,t)'
xprs(i,t) 'the amount of material in sub-process p, xprs(i,p,t)'
ICAP(i,t) 'inventory capacity'

binary variables
yop(i,t) 'open/close a node, yop(i,t) = yoper'
yjopen(i,t) 'just opened site i at the beginning of time period t,yjopen(i,t)'
yjclose(i,t) 'just closed site i at the end of time period t-1,yjclose(i,t)'
yjclose_bar(i,t) 'just closed site i at the end of time period t-1,yjclose(i,t)'

integer variables
ytrs(i,j,m,t) " #vehicles between sites, ytrs(i,j,m,t)"
ypr(i,t) " #machines, ypr(i,Pm,t)"
yco(i,t) " #workers at collection sites, yco(icp,t)"
ytrc(i,c,m,t) " #vehicles to customers, ytrc(i,c,m,t)"
;

free variables
TotalSiteOpenningCost
;


equation
obj objective function
minrecycle 'minimum recycle each year'
```

```
balancecollect  'flow balance for permanent collection sites'
balanceprocess  'flow balance for processing sites'
balancelandfill 'flow balance for landfill sites'
balance_amt_process 'calculate amt process = amt sent to customer'
balance_amt_collect
balance_amt_landfill

close1
close2
close3
close4
close5
close6
close7

supply 'supply constraint at physical site location'
demand 'demand constraint'
justopen 'just opened a site at the beginning of current time period'
justclose 'just closed a site at the end of previous time period'
firstopen 'open a site at the first time period'

collectCap collection Cap
processCap processing Cap
allowcol allow to collect
allowpro allow to process
allowlf  allow to open landfill
allow_ytrs
allow_ypr
allow_ypr1
allow_ypr2

;
```

```
***************************** Objective Function ******************************            ok
obj.. TCost =e=   sum((ic,t)$Fop(ic,t),Fop(ic,t)*yop(ic,t))
               +sum((ic,t)$Fjopen(ic,t),Fjopen(ic,t)*yjopen(ic,t))
               +sum((ic,t)$Fjclose(ic,t),Fjclose(ic,t)*yjclose(ic,t))

               +sum((ip,t)$Fop(ip,t),Fop(ip,t)*yop(ip,t))
               +sum((ip,t)$Fjopen(ip,t),Fjopen(ip,t)*yjopen(ip,t))
               +sum((ip,t)$Fjclose(ip,t),Fjclose(ip,t)*yjclose(ip,t))

               +sum((il,t)$Fop(il,t),Fop(il,t)*yop(il,t))
               +sum((il,t)$Fjopen(il,t),Fjopen(il,t)*yjopen(il,t))
               +sum((il,t)$Fjclose(il,t),Fjclose(il,t)*yjclose(il,t))

               +sum((i,j,m,t),Ftrs(i,j,m,t)*ytrs(i,j,m,t))
               +sum((i,k,j,m,t),Vtrs(i,j,m,t)*xtrs(i,k,j,m,t))

               +sum((ip,k,c,m,t),Vtrc(ip,c,m,t)*xtrc(ip,k,c,m,t))

               +sum((i,t)$Fpr(i,t),Fpr(i,t)*ypr(i,t))
               +sum((i,t)$Vpr(i,t),Vpr(i,t)*xprs(i,t));

******************************* Min Recycle Percentage ********************************** ok
minrecycle(t)..
sum((ip,k,c,m),xtrc(ip,k,c,m,t)) =g= b(t)* plastic(t);

******************************* Flow Balance **********************************   ok
balancecollect(i(ic),k,t)..
sum((j(is),m),xtrs(j,k,i,m,t)) =e=
sum((j(ip),m),xtrs(i,k,j,m,t)) + sum((j(il),m),xtrs(i,k,j,m,t));

balanceprocess(i(ip),k,t)..
sum((j(ic),m),xtrs(j,k,i,m,t)) =e=
sum((c,m),xtrc(i,k,c,m,t)) + sum((j(il),m),xtrs(i,k,j,m,t));

balancelandfill(i(il),k,t)..
sum((j(is),m),xtrs(j,k,i,m,t)) + sum((j(ic),m),xtrs(j,k,i,m,t)) + sum((j(ip),m),xtrs(j,k,i,m,t))
=e=
sum((c,m),xtrc(i,k,c,m,t));

balance_amt_process(i(ip),t)..
sum((c,k,m),xtrc(i,k,c,m,t)) =e= xprs(i,t);

balance_amt_collect(i(ic),t)..
sum((j(is),k,m),xtrs(j,k,i,m,t)) =e= xprs(i,t);

balance_amt_landfill(i(il),t)..
sum((c,k,m),xtrc(i,k,c,m,t)) =e= xprs(i,t);

******************************* Supply/Demand **********************************        ok
supply(i(is),k,t).. sum((j(ic),m),xtrs(i,k,j,m,t)) =e= Sup(i,k,t);
demand(k,c,t).. sum((i(ip),m),xtrc(i,k,c,m,t)) + sum((i(il),m),xtrc(i,k,c,m,t)) =l= Dem(k,c,t);

*****site capacity*****          ok
collectCap(i(ic),t).. sum((j(is),k,m),xtrs(j,k,i,m,t)) =l= Lc(i,t);
processCap(i(ip),t).. sum((j(ic),k,m),xtrs(j,k,i,m,t)) =l= Lp(i,t);

***************************** Logical Open/Close *******************************
close1(i(il),t)$(ord(t)= 1 ).. ICAP(i,t) =e= CAP(i);
close2(i(il),t)$(ord(t)>1 ).. ICAP(i,t) =e=  ICAP(i,t-1) + sum((j,k,m),xtrs(j,k,i,m,t));
close3(i(il),t)$(ord(t)< 3).. yjclose(i,t+1) - 1 =l= bigM*yjclose_bar(i,t+1);
close4(i(il),t)$(ord(t)>1 ).. LCAP(i) - ICAP(i,t) =l= bigM*(1-yjclose_bar(i,t+1));
close5(i,t)$(ord(t)>1 ).. yjclose(i,t-1) =l= yjclose(i,t);
close6(i,t)$(ord(t)>1 ).. yjopen(i,t+1) + yjclose(i,t) =l= 1;
close7(i).. yjopen(i,'t1') + yjopen(i,'t2') + yjopen(i,'t3') =l= 1;
```

```
justopen(i,t)$(ord(t)>1 ).. yop(i,t)-yop(i,t-1) =l= yjopen(i,t);
justclose(i,t)$(ord(t)>1 ).. yop(i,t-1)-yop(i,t) =l= yjclose(i,t);
firstopen(i,t).. yop(i,'t1') =l= yjopen(i,'t1');

**************************** Logical Constraints *******************************
allowcol(ic,t).. sum((j(is),k,m),xtrs(j,k,ic,m,t)) =l= bigM*yop(ic,t);
allowpro(ip,t).. sum((j(ic),k,m),xtrs(j,k,ip,m,t)) =l= bigM*yop(ip,t);
allowlf(il,t).. sum((j(is),k,m),xtrs(j,k,il,m,t)) + sum((j(ic),k,m),xtrs(j,k,il,m,t))+
sum((j(ip),k,m),xtrs(j,k,il,m,t)) =l= bigM*yop(il,t);

allow_ytrs (i,k,j,m,t)..   xtrs(i,k,j,m,t) =l= bigM*ytrs(i,j,m,t);
allow_ypr(ip,t).. xprs(ip,t) =l= bigM*ypr(ip,t);
allow_ypr1(ic,t).. xprs(ic,t) =l= bigM*ypr(ic,t);
allow_ypr2(il,t).. xprs(il,t) =l= bigM*ypr(il,t);

yjopen.lo(il1,'t1') = 1;
*yjopen.lo(il2,'t1') = 1;
yop.lo(il1,'t1') = 1;
*yop.lo(il2,'t1') = 1;
*yjopen.lo('i77','t1') = 1 ;


model finalmodel /all/;
finalmodel.PriorOpt = 1;
finalmodel.optcr = 0.05;

yop.up(is,t) = 0;
yjopen.up(is,t) = 0;
xtrc.up(is,k,c,m,t) = 0;



solve finalmodel using mip min TCost;
```

```
display xtrs.l;
display xtrc.l;
display xprs.l;
display yop.l;
display yjopen.l;
display yjclose.l;
display TCost.l;

parameter CostCollectOpening;
CostCollectOpening =sum((ic,t),Fop(ic,t)*yop.l(ic,t))
                         + sum((ic,t),Fjopen(ic,t)*yjopen.l(ic,t))
                         + sum((ic,t),Fjclose(ic,t)*yjclose.l(ic,t));
parameter CostProcessOpening;CostProcessOpening =sum((ip,t),Fop(ip,t)*yop.l(ip,t))
                         + sum((ip,t),Fjopen(ip,t)*yjopen.l(ip,t))
                         + sum((ip,t),Fjclose(ip,t)*yjclose.l(ip,t));

parameter CostLandFill;CostLandFill =sum((il,t),Fop(il,t)*yop.l(il,t))
                         + sum((il,t),Fjopen(il,t)*yjopen.l(il,t))
                         + sum((il,t),Fjclose(il,t)*yjclose.l(il,t));

parameter Transportation;
Transportation      =sum((i,j,m,t),Ftrs(i,j,m,t)*ytrs.l(i,j,m,t))
                 +sum((i,k,j,m,t),Vtrs(i,j,m,t)*xtrs.l(i,k,j,m,t));

*parameter TransportationCust;
*TransportationCust =sum((i,k,c,m,t),Vtrc(i,c,m,t)*xtrc.l(i,k,c,m,t));

parameter CostProcessC;
CostProcessC       =sum((ic,t),Fpr(ic,t)*ypr.l(ic,t))
                 +sum((ic,t),Vpr(ic,t)*xprs.l(ic,t));

parameter CostProcessP;
```

```
CostProcessP       =sum((ip,t),Fpr(ip,t)*ypr.l(ip,t))
                   +sum((ip,t),Vpr(ip,t)*xprs.l(ip,t));

parameter CostProcessL;
CostProcessL       =sum((il,t),Fpr(il,t)*ypr.l(il,t))
                   +sum((il,t),Vpr(il,t)*xprs.l(il,t));



display CostCollectOpening, CostProcessOpening,
CostLandFill,Transportation,CostProcessC, CostProcessP, CostProcessL;
```