```
$EOLCOM //
$TITLE Iraq Water Rights Priority Model
$OFFSYMXREF OFFSYMLIST OffLISTING OFFUPPER

OPTION LIMROW=000, LIMCOL = 0;

$ONTEXT

EOLCOM // tells GAMS to ignore anything in the line's text after the symbol //
OFFLISTING deletes all program lines and just includes GAMS listing
Set LIMROW = 0 to eliminate all equations in the GAMS listing
Set LIMROW = 100 or more to show all equations in listing. Helps trap  errors

* ---------------------------------------------------------------------------------------
  Frank A. Ward:
  Dept of Agr Economics/Agr Business
  New Mexico State University, Las Cruces, NM USA
  e-mail: fward@nmsu.edu

  Dina A. Salman:
  Dept of Agr Economics/Agr Business
  New Mexico State University, Las Cruces, New Mexico USA
  email: dinasalm@nmsu.edu

  Saud A.Amer:
  US Geological Survey - International Water Resources Division
  Reston VA  USA
  email:  samer@usgs.gov

  May 13, 2013

* ---------------------------------------------------------------------------------------
*  Model 5 Simple Farm management
*  Exports results to excel spreadsheet
* ---------------------------------------------------------------------------------------

$OFFTEXT

*************************************   Section 1 ***********************************
*                                       Sets                                          *
*******************************************************************************
```

```
SETS

i    province

/
  1-Mousil
  2-Basra
/

k    crop

/  1-wheat
   2-cotton
/


s   hydrologic water supply scenario

/
normal
dry/

j  water right priority:  water right priority: same number # of elements as # of provinces - 1 priority per province

/      j1*j2
/

r water allocation rule allows for many water shortage sharing rules to be considered

/
us_priority         upstream   priority                  //rule 1
prop_sharing        proportional sharing of shortages    //rule 2
/

set rji(r, j, i) mapping set: assigns priorities to province - separate assignments for each proposed rule

/ us_priority  .  (j1.1-Mousil,  j2.2-Basra)
  prop_sharing .  (j1.1-Mousil,  j1.2-Basra)
/

**********************************  Section 2
************************************************************************
```

```
*                                    Data
*
*****************************************************************************************************************
****

table  Bc(i,k)    crop water demand (ET) (1000s cubic meters per Ha = 10ths of meters depth)

* Data: Allen 1998, FAO 56 applied to year 2012 weather information in Iraq
* Crop Water Requirements: Food and Agricultural Organization Report 56.

                1-wheat          2-cotton
 1-mousil         11.9             18.0
 2-Basra          13.5             21.4

*Bc(k) =  0.5 * Bc(k)

table    Yield_p(i,k)   Crop Yield tons per Ha (proportional to ET)

               1-wheat         2-Cotton
1-Mousil        1.39             2.32
2-Basra         1.57             2.75

* economic data


parameter   Price_p(k)        Crop Prices ($ US per ton)

/1-Wheat        225
 2-Cotton       906
/

table Cost_p(i, k)    Crop Production Costs excluding water ($ US per Ha)

* FAO Ag Stats, adapted from Iran and Egypt

               1-wheat   2-cotton
1-Mousil         200      1286
2-Basra          207      1345

;

* data from Dr. Saud Amer 2012
```

```
Table land_p(i,k)   observed land in prodn for year 2006 (1000 Ha)

                 1-wheat        2-cotton
1-mousil           47.4           0.45
2-basra            16.9           0.00
;

Parameter Net_revenue_p(i,k)  calculated net revenue per ha ($ US per HA)
;

Net_revenue_p(i,k) = Price_p(k) * Yield_p(i,k) - Cost_p(i,k);

parameter Wat_supply_p(s)      total water available (million cubic meters per year) calculated total ag water use (not
meas from gauges)
;
Wat_supply_p('normal') = sum((i,k), land_p(i,k) * Bc(i,k));  // calculates observed water use
Wat_supply_p('dry'   ) = 0.50 * wat_supply_p('normal');      // dry: half of observed value

parameter right_p(i)    total water assigned by province in full supply conditions (paper water)
;

right_p(i)  =  sum(k, Bc(i,k) * land_p(i,k));  // sums over crops: ith province's historical (2006) water use under
full supply

******************  code defines water sharing methods ***************************************

* code below loops over each priority (j) then maps each priority to corresponding province i
* code is thanks to Pete Stacy at GAMS Development Corporation Feb 1 2012

parameter   tot_assigned(r,s)   cumulative water assigned to province (i) including higher priorities;
            tot_assigned(r,s)  = 0;  // starts at 0 - prepares to loop

parameter
  tot_pap_right_by_prior(  r,j)  cumulative paper right at jth priority for all provinces including ties
  remain_supply_by_prior(r,s,j)  residual supply    at jth priority after supplying higher priorities
  wet_wat_use           (r,s,i)  wet (not paper) water use assigned to ith province - limited by total basin supply
;

Loop(r,                                                                    // loop over sharing rule (r)
 Loop(j,                                                                   // loop over priority (j)
```

```
   tot_pap_right_by_prior(r,j) = sum[i$rji(r, j, i), right_p(i)] ;           // total paper rights by (j) priority by
provinces includes ties
   remain_supply_by_prior(r,s,j) =
       min[(wat_supply_p(s) - tot_assigned(r,s)), tot_pap_right_by_prior(r,j)] + eps;    // remaining supply by (j)
priority after protecting higher priorities
   Loop(i$rji(r, j, i),
       wet_wat_use(r,s,i) = (right_p(i)/tot_pap_right_by_prior(r,j))
       * remain_supply_by_prior(r,s,j) + eps;                                  // wet water assigned to (i) province
    tot_assigned(r,s) = tot_assigned(r,s) + wet_wat_use(r,s,i) ;               // cumulative water (check) assigned to
last province getting water - should match total supply

   );                                                                          // end province loop
  );                                                                           // end priority loop
);                                                                             // end rule loop

parameter tot_wat_use(r,s)  total water use defined;
tot_wat_use  (r,s) = sum(i, wet_wat_use(r,s,i)) + eps;   // checks total use assigned against supply

*********** end of water allocation system ***************************************************************

**** begin looping over water sharing rule and water supply scenario below
* rr, ss, are subset of orginal sets... allows fast scenario analysis below


set rr(r);   // water sharing rule
set ss(s);   // water supply scenario

rr(r)=no;    // switches subset off for now
ss(s)=no;    // switches subset off for now

*****************************  SECTION 3 ********************************************
*                              Variables                                           *
*********************************************************************************

positive variables

hectares_v        (r,s,i,k)      land in production by rule-province-crop-scen    (1000 Ha - marginal is $US per Ha)
T_hectares_v      (r,s,i  )      total land in prodn by rule-province-scen        (1000 Ha - marginal is $US per Ha)

uses_crop_v       (r,s,i,k)      total water use by rule-province-crop-scen       (million m^3 - marginal is $US per
1000 m^3)
uses_v            (r,s,i  )      total water use by rule-province-scen            (million m^3 - marginal is $US per
```

```
1000 m^3)

variables

ag_ben_k_v        (r,s,i,k)      total farm income by rule-province-crop-scen     ($US 1000s - no marginals shown)
ag_ben_v          (r,    s)      total farm income by rule-scen                   ($US 1000s - no marginals shown)

Tot_b_v          total farm income re-calc for each rule-scen (objective) ($US 1000s - no marginals shown)

*****************************   Section 4  ****************************************
*                               Equations                                          *
*********************************************************************************


Equations

// Equations DECLARED

T_hectares_e       (r,s,i  )    total land in production by province-scen  (1000 Ha - marginal is $US per Ha)

Uses_crop_e        (r,s,i,k)    total water use by crop-province-scean     (million m^3 - marginal is $US per 1000 m^3)
uses_e             (r,s,i  )    total water use by scen                    (million m^3 - marginal is $US per 1000 m^3)

ag_ben_k_e         (r,s,i,k)    total farm income by crop-province-scen    ($US 1000s - no marginals shown)
ag_ben_e           (r,    s)    total farm income by scen                  ($US 1000s - no marginals shown)

Tot_b_e                         total farm income by element of loop       ($US 1000s - no marginals shown)
;


// Equations defined (using above names with algebraic formulas)

// Equations below defined over rule (rr) and scenario (ss) using subset rr and ss

T_hectares_e (rr,ss,i  )..  T_hectares_v(rr,ss,i  )  =e=  sum(k,   hectares_v  (rr,ss,i,k));

Uses_crop_e  (rr,ss,i,k)..  uses_crop_v (rr,ss,i,k)  =e=  Bc(i,k) *  hectares_v(rr,ss,i,k );
uses_e       (rr,ss,i  )..  uses_v      (rr,ss,i  )  =e=  sum(k,    uses_crop_v(rr,ss,i,k));

ag_ben_k_e   (rr,ss,i,k)..  ag_ben_k_v  (rr,ss,i,k)  =e=  net_revenue_p(i,k) * hectares_v(rr,ss,i,k);
ag_ben_e     (rr,ss    )..  ag_ben_v    (rr,ss)      =e=  sum((i,k), ag_ben_k_v(rr,ss,i,k));
```

```
Tot_b_e                 ..  Tot_b_v                 =e=  sum((rr,ss), ag_ben_v(rr,ss   ));   // sums total benefits
over indices for sensitivity analysis

*************** Section 5 ************************************************************************
*  Labels and defines models                                                                     *
*  Each model has one objective function                                                         *
*********************************************************************************************

model TE_05 /all/ ;

*********************************  Section 6   ***************************************
*                                 BOUNDS                                               *
*  Bounding shows a positive shadow price for each limiting resource
*********************************************************************************************

uses_v.up    (r,s,i  ) = wet_wat_use(r,s,i  );  // upper bound limits available water to observed water use
hectares_v.up(r,s,i,k) =      land_p(    i,k);  // can produce no more crops than observed under full water supply
conditions

*hectares_v.lo('1-mousil','1-wheat') =  10;     // food security requirement
*uses_crop_v.lo('wheat')  = 1;                  // another way of enforcing food security


*************** Section 7 ************************************************************************
*               SOLVE THE MODEL                                                               *
*********************************************************************************************
parameter shad_price_p(r,s,i)   names shadow price of water to send to spreadsheet
;

  loop(r,
   loop(s,

     ss(s) = yes;
     rr(r) = yes;

     Solve TE_05 using nlp maximizing Tot_b_v;

      shad_price_p(r,s,i) =       uses_v.m(r,s,i) + eps;    // export shad prices for all water allocation rules except
unrestricted trading

* closes loops below over r,s (rule for sharing shortages, water supply scenario)
```

```
    rr(r) = no;
    ss(s) = no;

   );
  );
```

```
*******************************************************************************************
*   Section 8: DISPLAY RESULTS TO SPREADSHEET                                             *
*******************************************************************************************
```

```
parameter

uses_p                      (r,s,i  )   total water use by province summed over crops
land_v_p                    (r,s,i,k)   total land in production
ben_by_crop_province_p      (r,s,i,k)   total benefits by crop and province
ben_by_province_p           (r,s,i  )   total economic benefits by province
tot_ben_p                   (r,s    )   total economic benefits

;

* land in prodn (1000 hectares)

land_v_p                    (r,s,i,k) = hectares_v.l(r,s,i,k) + eps;

* water use (million cubic meters)

uses_p                      (r,s,i  ) =    Uses_v.l(r,s,i) + eps;

* economic benefits ($1000 US)

ben_by_crop_province_p      (r,s,i,k)      =        ag_ben_k_v.l(r,s,i,k ) + eps;
ben_by_province_p           (r,s,i  )      = sum(k, ag_ben_k_v.l(r,s,i,k)) + eps;
tot_ben_p                   (r,s    )      =         ag_ben_v.l (r,s     ) + eps;

* GAMS GDX facility writes to excel spreadsheet

execute_unload "farm_mgmt_05.gdx"

tot_pap_right_by_prior  remain_supply_by_prior wet_wat_use
tot_assigned            land_p                 land_v_p
uses_p                  ben_by_crop_province_p ben_by_province_p
```

```
tot_ben_p                net_revenue_p          shad_price_p
Price_p                  yield_p                cost_p
Net_revenue_p            Bc

;

$onecho > gdxxrwout.txt

i=farm_mgmt_05.gdx
o=farm_mgmt_05.xls

epsout = 0

par = tot_pap_right_by_prior  rng =  total_paper_right_by_priority!c4   Cdim = 0
par = remain_supply_by_prior  rng =  remain_supply_by_priority!c4       cdim = 0
par = wet_wat_use             rng =  wet_water_use_by_province!c4       Cdim = 0
par = tot_assigned            rng =  basin_cum_water_use!c4             cdim = 0

par = land_v_p                rng =  land_in_prodn!c4                   cdim = 0
par = uses_p                  rng =  total_water_use_by_province!c4     cdim = 0

par = net_revenue_p           rng =  net_rev_per_ha!c4                  cdim = 0
par = ben_by_crop_province_p  rng =  benefits_by_crop_province!c4       cdim = 0
par = ben_by_province_p       rng =  benefits_by_province!c4            cdim = 0
par = tot_ben_p               rng =  total_benefits!c4                  cdim = 0

par = shad_price_p            rng =  shadow_price!c4                    cdim = 0
par = Price_p                 rng =  price!c4                           cdim = 0
par = cost_p                  rng =  cost!c4                            cdim = 0
par = yield_p                 rng =  yield!c4                           cdim = 0
par = Net_revenue_p           rng =  net_rev!c4                         cdim = 0
par = Bc                      rng =  wu_ha!c4                           cdim = 0

$offecho
execute 'gdxxrw.exe @gdxxrwout.txt trace=2';

// THE END
```