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

OPTION LIMROW=000, LIMCOL = 0;

$ONTEXT

* ------------------------------------------------------------------------------------------
  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 6 Simple Farm Management
*  Expands scope and scale of Model 5
*  2 to 13 provinces
*  2 to 3 water supply scenarios
*  2 to 4 water shortage sharing methods
*  2 to 9 crops
* ------------------------------------------------------------------------------------------

* model 6, adds 13 provinces and 9 crops and 4 water sharing arrangements


$OFFTEXT

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

```
*************************************************************************************

SETS

**************************************************************************************
i     province
**************************************************************************************

/      1-Mousil
       2-Kurkuk
       3-Salaheldeen
       4-Deyala
       5-Anbar
       6-Baghdad
       7-Babylon
       8-Karbala
       9-Qadeseeya
      10-Muthana
      11-Thieqar
      12-Meesan
      13-Basra
/


k    crop

/ 1-Rice
  2-Wheat
  3-Cotton
  4-Sunflower
  5-Maize
  6-Barley
  7-Tomato
  8-Lettuce
  9-Onion
/


s   hydrologic water supply scenario

/
1-normal
```

```
2-dry
3-drought /


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


/      j1*j13
/

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

/1-ds_priority   downstream priority                                                   // rule 1
 2-us_priority   upstream   priority                                                   // rule 2
 3-prop_sharing  proportional sharing of shortages                                     // rule 3
 4-free_market   free market - proportional sharing then water moves by market         // rule 4
/

*subsets of r
set rwa(r) / 1-ds_priority, 2-us_priority, 3-prop_sharing/   // subset of rules = everything except market trading
set rfm(r) / 4-free_market/                                  // subset of rules = only market trading of water in drought


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

// best (senior) right assigned to canal with j1 priority - worst (junior) priority assigned to canal with j4
// Two canals with equal priority (e.g. two both with j1) share shortages proportionally

/1-ds_priority  . (j1.13-Basra,  j2. 12-Meesan,  j3.11-Thieqar,    j4.10-Muthana, j5.9-Qadeseeya, j6.8-Karbala,  j7.7-Babylon,  j8.6-
Baghdad,  j9.5-Anbar,     j10.4-Deyala,   j11.3-Salaheldeen, j12.2-Kurkuk,  j13.1-Mousil)
 2-us_priority  . (j1.1-Mousil,  j2.2-Kurkuk,    j3.3-Salaheldeen, j4.4-Deyala,   j5.5-Anbar,     j6.6-Baghdad,  j7.7-Babylon,  j8.8-
Karbala,  j9.9-Qadeseeya, j10.10-Muthana, j11.11-Thieqar,    j12.12-Meesan, j13.13-Basra )
 3-prop_sharing . j1. (1-Mousil, 2-Kurkuk, 3-Salaheldeen, 4-Deyala, 5-Anbar, 6-Baghdad, 7-Babylon, 8-Karbala, 9-Qadeseeya, 10-
Muthana, 11-Thieqar, 12-Meesan, 13-Basra )
 4-free_market  . j1. (1-Mousil, 2-Kurkuk, 3-Salaheldeen, 4-Deyala, 5-Anbar, 6-Baghdad, 7-Babylon, 8-Karbala, 9-Qadeseeya, 10-
Muthana, 11-Thieqar, 12-Meesan, 13-Basra )
/

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

```
TABLE Bc(i, k)    ET - Per hectare crop water demand 10ths of meters depth  = 1000s cubic meters per ha.

* Data:  Allen, R. G., L.S. Pereira, D. Raes, and M. Smith, 1998. Crop Evapotranspiration Guidelines for Computing
* Crop Water Requirements: Food and Agricultural Organization Report 56.

* ----- Column Heads are Crops -- Row heads are irrigation canals (water use areas) -----

                   1-Rice        2-wheat     3-cotton  4-sunflower  5-Maize   6-Barley     7-tomato     8-lettuce     9-onion
*-------------------------- use node rows ---------------------------------------------------------------------------------
 1-Mousil            30.8        11.9        18.0        12.9        7.0        2.8        7.1          1.7          9.5
 2-Kurkuk            32.4        12.3        19.1        13.4        7.8        2.6        7.9          1.6         10.0
 3-Salaheldeen       28.4        9.9         16.6        10.9        6.8        2.5        6.9          1.6          9.4
 4-Deyala            29.6        10.8        17.3        11.8        6.6        3.1        6.7          2.4         10.1
 5-Anbar             33.8        12.4        19.7        13.6        8.8        3.1        8.8          1.9         11.7
 6-Baghdad           32.2        11.8        18.8        13.0        8.2        2.8        8.2          1.7         10.8
 7-Babylon           32.5        11.9        18.6        13.0        8.2        3.0        8.2          1.8         10.9
 8-Karbala           32.8        11.9        18.8        13.1        8.6        3.1        8.6          1.9         11.5
 9-Qadeseeya         34.2        12.3        19.6        13.6        9.2        3.4        9.2          2.0         12.3
 10-Muthana          34.4        12.3        19.6        13.5        9.3        3.6        9.3          2.1         12.8
 11-Thieqar          35.6        12.8        20.3        14.1        9.0        3.0        9.1          1.9         11.9
 12-Meesan           34.8        12.5        20.1        13.8        9.2        3.4        9.2          2.1         12.3
 13-Basra            37.2        13.4        21.4        14.9        9.7        3.4        9.8          2.1         12.7
*-------------------------------------------------------------------------------------------------------------------------
;

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

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

                  1-Rice        2-wheat    3-cotton   4-sunflower   5-Maize     6-Barley    7-tomato    8-lettuce       9-onion
*-------------------------- use node rows (+) -------------------------------------------------------------------------------------
  1-Mousil          2.89        1.38        2.42        1.33          2.43        0.90        15.8        20.7          7.40
  2-Kurkuk          2.89        1.38        2.42        1.33          2.43        0.90        15.8        20.7          7.40
  3-Salaheldeen     2.89        1.38        2.42        1.33          2.43        0.90        15.8        20.7          7.40
  4-Deyala          2.89        1.38        2.42        1.33          2.43        0.90        15.8        20.7          7.40
  5-Anbar           2.89        1.38        2.42        1.33          2.43        0.90        15.8        20.7          7.40
  6-Baghdad         2.89        1.38        2.42        1.33          2.43        0.90        15.8        20.7          7.40
  7-Babylon         2.89        1.38        2.42        1.33          2.43        0.90        15.8        20.7          7.40
  8-Karbala         2.89        1.38        2.42        1.33          2.43        0.90        15.8        20.7          7.40
  9-Qadeseeya       2.89        1.38        2.42        1.33          2.43        0.90        15.8        20.7          7.40
  10-Muthana        2.89        1.38        2.42        1.33          2.43        0.90        15.8        20.7          7.40
```

```
  11-Thieqar        2.89        1.38        2.42        1.33          2.43        0.90       15.8         20.7            7.40
  12-Meesan         2.89        1.38        2.42        1.33          2.43        0.90       15.8         20.7            7.40
  13-Basra          2.89        1.38        2.42        1.33          2.43        0.90       15.8         20.7            7.40
*----------------------------------------------------------------------------------------------------------------------------------
;

* use formula below to reduce yields with cooler climates and increase them with warmer .

Yield_p(i,k) = yield_p ('6-baghdad',k) *  Bc(i, k)/ Bc('6-baghdad', k) ;


* economic data

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

/1-Rice                      985
 2-wheat                     225
 3-cotton                    906
 4-sunflower                 415
 5-Maize                     180
 6-Barley                    175
 7-tomato                    115
 8-lettuce                   120
 9-onion                     142
/
;


Table  Cost_p(i,k)      Crop Production Costs Excluding water ($ US per Ha)

                  1-Rice      2-wheat    3-cotton  4-sunflower    5-Maize    6-Barley  7-tomato    8-lettuce      9-onion
*-------------------------- use node rows (+) --------------------------------------------------------------------------------

 1-Mousil            180        200         1300          220         70         100       60          80              70
 2-Kurkuk            180        200         1300          220         70         100       60          80              70
 3-Salaheldeen       180        200         1300          220         70         100       60          80              70
 4-Deyala            180        200         1300          220         70         100       60          80              70
 5-Anbar             180        200         1300          220         70         100       60          80              70
 6-Baghdad           180        200         1300          220         70         100       60          80              70
 7-Babylon           180        200         1300          220         70         100       60          80              70
 8-Karbala           180        200         1300          220         70         100       60          80              70
 9-Qadeseeya         180        200         1300          220         70         100       60          80              70
```

```
 10-Muthana            180       200        1300          220        70        100       60        80             70
 11-Thieqar            180       200        1300          220        70        100       60        80             70
 12-Meesan             180       200        1300          220        70        100       60        80             70
 13-Basra              180       200        1300          220        70        100       60        80             70
*------------------------------------------------------------------------------------------------------------------------
;

* 1/4 of all costs relate to yield, and 3/4 are independent of yield

parameter cost_harvest_p(i,k)  production costs per ha related to crop yield
;

cost_harvest_p(i,k) = 0.25 * cost_p(i,k);  // harvest-related costs approx 25% of all prodn costs - data source needed
cost_harvest_p(i,k) =  cost_harvest_p('6-Baghdad', k) * yield_p(i,k) / yield_p('6-baghdad', k);  // costs related to harvests


parameter cost_n_harvest_p(i,k)  costs not related to (independent of) harvest;

cost_n_harvest_p(i,k) = 0.75 * cost_p(i,k)  // higher total prodn costs with higher yields.  Baghdad is reference value

* re assemble cost below.  For any given crop we have (1) rising costs with higher yields and (2)  constant costs over provinces

parameter cost_p(i,k)  total cost per ha;

Cost_p(i,k)= Cost_harvest_p(i,k) + cost_n_harvest_p(i,k);


Table land_p(i,  k)   observed land in prodn (1000 Ha)

                    1-Rice          2-wheat      3-cotton  4-sunflower      5-Maize    6-Barley    7-tomato    8-lettuce    9-onion
*-------------------------- use node rows (+) ----------------------------------------------------------------------------
   1-Mousil           0.0             47.4         0.45        0.4            0.0        77.1        43.6          3.7         0.0
   2-Kurkuk           0.0             26.1         1.60        0.0            0.5         1.5         6.9          2.0         0.0
   3-Salaheldeen      0.0            145.5         3.00        6.4           40.3        13.6        46.3         69.1         7.2
   4-Deyala           0.6            107.1         1.40        0.3           10.3        22.2        17.7         12.9         6.2
   5-Anbar            0.2             55.5         0.15        2.2            7.2         3.4        13.2         13.6        27.6
   6-Baghdad          0.0             65.7         2.60        2.0           25.6         8.2        51.8         27.6        13.7
   7-Babylon          1.6             74.2         1.00        0.6           73.0        44.4        15.6         15.3        47.0
   8-Karbala          0.0              2.9         0.04        0.2            3.1         2.9        11.8          7.8         4.8
   9-Qadeseeya       32.5             91.8         0.40        0.05           3.8        72.3        15.2          9.6        15.3
   10-Muthana         4.5             12.8         0.00        0.0            0.0        29.1         1.6          0.7        10.3
   11-Thieqar         0.1             61.3         0.05        0.03           3.4        81.5        16.7         11.1        11.4
```

```
   12-Meesan           6.8              81.0         0.00         0.005          12.1         65.5        18.6          8.9          20.0
   13-Basra            0.0              16.9         0.00         0.0             8.6          0.0        10.5         24.9           0.1
;

Parameter Net_revenue_p  (i,k)  net revenue per ha observed
*          T_net_revenue_p(i  )  total net revenue by province summed over crops
;

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

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('1-normal')  =  sum((i,k), Bc(i,k) * land_p(i,k));    // actual water use in full supply year
Wat_supply_p('2-dry')     =  0.50 * Wat_supply_p('1-normal');              // 50% of full supply
Wat_supply_p('3-drought') =  0.20 * Wat_supply_p('1-normal');              // 20% of full supply

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))};

*******************  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 assignment at canal i including higher priorities;
            tot_assigned(r,s)  = 0;   // starts at 0 - prepares to loop

parameter
  tot_pap_right_by_prior(  r,j) total paper rights by jth priority for all canals including ties
  remain_supply_by_prior(r,s,j) residual supply    by jth priority after supplying higher priorities
  wet_wat_use           (r,s,i) wet water use assigned to ith 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) after protecting higher priorities
   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 jth priority after
```

```
*supplying 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;
tot_wat_use  (r,s) = sum(i, wet_wat_use(r,s,i)) + eps;

************** 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;   // switch rr subset off -- prepares to turn it on below in equations
ss(s) = no;   // switch ss subset off -- prepares to turn it on below in equations

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

Sum_uses_v                (r,s    )    summed wat uses over provinces by rule-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-scenario-province-crop ($US 1000s - no marginals shown)
Ag_Ben_v                  (r,s    )    total farm income by rule-scenario-province      ($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 rule-scen-prov-crop     (million m^3 - marginal is $US per 1000 m^3)
Uses_e                   (r,s,i      )       total water use by rule-scen-prov          (million m^3 - marginal is $US per 1000 m^3)

Sum_uses_e               (r,s        )       total water use by rule-scen               (million cubic meters per year)

ag_ben_k_e               (r,s,i,   k )       total farm income by rule-scen-prov-crop   ($US 1000)
Ag_ben_e                 (r,s        )       total farm income by rule-scen-prov        ($US 1000)

Tot_b_e                                      total farm income by element of loop       ($US 1000)
;


// 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));
sum_uses_e  (rr,ss    )..   sum_uses_v(rr,ss     )  =e=          sum(i, Uses_v(rr,ss,i  ));

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 each rule-scen

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

MODEL TE_06 /all/;

**********************************  Section 6   *****************************************
*                                   BOUNDS                                              *
*  Bounding variables lets you see a non-zero shadow price if a resource is limiting    *
*****************************************************************************************

uses_v.up   (rwa,s, i)  = wet_wat_use(rwa,s,i);  // without water trading water use limited to historic by province
sum_uses_v.up(rfm,s  )  = tot_wat_use(rfm,s  );  // with water trading only total (basin) water is bounded

hectares_v.up(r,s,i,k)  = land_p(i,k);           // can produce no more crops than observed under full water supply

**************** Section 7 **************************************************************
*               MODEL SOLVES                                                            *
*****************************************************************************************

parameter shad_price_p(r,s,i)  shadow prices calculated below after each optimization
;

* prepare for multiple solves inside the loops

  loop(r,
   loop(s,

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

     Solve TE_06 using nlp maximizing Tot_b_v;

     shad_price_p(rwa,s,i    ) =            uses_v.m(rwa,s,i) + eps;   // shad prices for all water allocation rules ex trading
     shad_price_p(rfm,s,i    ) =        sum_uses_v.m(rfm,s  ) + eps;   // shad prices for trading -  should be = across provinces

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

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

   );
  );

************************************************************************************************
*   Section 8: DISPLAYS                                                                        *
************************************************************************************************

*  Parameters are calculated values from the optimal solution, first defined then found

parameter

tot_use_p           (r,s,i,k  )   total water use by rule-scen-prov-crop
uses_p              (r,s,i    )   total water use by rule-scen_prov

land_v_p            (r,s,i,k  )   total land in production

ben_by_crop_canal_p (r,s,i,k  )   total benefits by crop and province
ben_by_canal_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 (1000 hectare meters)

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

* economic benefits ($1000 US)

ben_by_crop_canal_p (r,s,i,k) =    Ag_ben_k_v.l    (r,s,i,k)  + eps;
ben_by_canal_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 external spreadsheet
```

```
execute_unload "farm_mgmt_06.gdx"

tot_pap_right_by_prior,        remain_supply_by_prior,  wet_wat_use,   tot_assigned
land_p,          cost_p,        land_v_p,    tot_use_p,  uses_p,       ben_by_crop_canal_p,
ben_by_canal_p, tot_ben_p,     net_revenue_p, shad_price_p, Price_p
yield_p,         Net_revenue_p, Bc
;

$onecho > gdxxrwout.txt

i=farm_mgmt_06.gdx
o=farm_mgmt_06.xls

epsout = 0

* cdim = 0 puts output into LIST format, one piece of data per line

par = net_revenue_p          rng =  data_net_rev_per_ha!c4             Cdim = 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 = tot_use_p              rng =  water_use_by_crop_canal!c4         cdim = 0
par = uses_p                 rng =  total_water_use_by_canal!c4        cdim = 0
par = ben_by_crop_canal_p    rng =  benefits_by_crop_canal!c4          cdim = 0

par = Bc                     rng =  data_wu_ha!c4                      cdim = 0

par = ben_by_canal_p         rng =  benefits_by_canal!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 =  data_price!c4                      cdim = 0
par = cost_p                 rng =  data_cost_per_ha!c4                cdim = 0
par = yield_p                rng =  data_yield!c4                      cdim = 0

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

// THE END
```