

MARKAL¹ Version 5.5 Enhancements and Updates

Table of Contents

1.0 Background	1
2.0 Overview of New Capabilities and Updates	1
3.0 New Capabilities	2
3.1 The New MARKAL Benefit/Cost Report	2
3.2 Commodity Flow Level User Constraint Parameter (ADRATIO)	4
4.0 Updates and Corrections	4
4.1 Ignore QC In/Out checks for DUM processes	5
4.2 Set low-temperature heat seasonal transmission to default = 1, to avoid /0 execution error	5
4.3 Interpolate more parameters (e.g., ENV_ENT, GROWTH, DECAY)	5
4.4 Adjust GAMS control files for move to <case>.G00 save files	5
4.5 Some adjustments and minor “bugs” fixes	5
4.6 Outstanding problems	5
Appendix A: New Benefit/Cost Routine	6

1.0 Background

ETSAP continues to provide support for and enhancements to the MARKAL model generator and report writer. This short Information Note describes these latest facilities.

2.0 Overview of New Capabilities and Updates

The main enhancements to MARKAL embodied in Version 5.5 consist of:

1. A new (corrected) Benefit/Cost report employing a new GAMS UnitB/C utility.
2. Addition of VRAT_FLO to enable VEDA-FE to associate commodity flows in/out of technologies in ADRATIO constraints.

These are discussed in details in Section 3.

Other minor enhancements and corrections have also been made, as mentioned in Section 4.

¹ See *Documentation for the MARKAL Family of Models*, Volume I - III, International Energy Agency - Energy Technology Systems Analysis Programme, October 2004.

3.0 New Capabilities

3.1 The New MARKAL Benefit/Cost Report²

Background

The MARKAL community has been using ‘Value Flows’ or ‘Benefit/Cost’ ratios in their analysis (see Value Flows in Linear Economic Models) to produce a Cost/Benefit report (COSTBEN). Some of the calculations require the values of matrix coefficients in concert with the row marginals with which each variable is involved to be examined, and the Benefits/Costs to be accumulated accordingly. This makes the calculations of these indicators very cumbersome in GAMS, and difficult to maintain.

GAMS has responded by developing a new mechanism using the CONVERT facility to produce a GDX file (UnitBC.GDX) with the desired information. This GDX can then be efficiently processed with a compact piece of GAMS code to produce the Cost/Benefit report.

The GDX file contains parameters corresponding to each of the variables of the model having the same name as the variable, with a postfix of _B for benefit and _C for costs appended. Thus for each variable appearing in a model solution (note, not model declaration) two parameters will appear in the UnitBC.GDX file (e.g., for variable CAP CAP_B and CAP_C will exist).

Approach

The nature and algorithm employed for deriving the UnitBenefit (_B) and UnitCost (_C) parameters are briefly outlined here.

j = activity column index(es); i = resource, row, constraint index(es)

$a(i,j)$ is the matrix coefficient

$C(j)$ is the coefficient in the objective function

Level (j) is the `gams.l(j)` value

ReducedCost (j) is the `gams.m(j)` variable marginal value

Shadow price (i) is the `gams.m(i)` row marginal value

At the appropriate indexing level of each variable these parameters are stored in the UnitBC.GDX file:

$$\begin{aligned} \langle \text{var} \rangle_B = \text{UnitBenefit}(r,j,t,ts,\dots) &= \text{sum}(i\$([\text{gams.m}(i)*a(i,j)] > 0), \text{gams.m}(i)*a(i,j)) \\ \langle \text{var} \rangle_C = \text{UnitCost}(r,j,t,ts,\dots) &= C(j) - \text{sum}(i\$([\text{gams.m}(i)*a(i,j)] < 0), \\ &\quad \text{gams.m}(i)*a(i,j)) \end{aligned}$$

² Requires GAMSv22.1 or higher, or obtain a special GMSCV_NX.EXE for ggoldstein@irglt.com. MARKAL must be run in CMD mode as well.

The new Cost/Benefit routine then constructs Benefit/Cost ratios at the appropriate level by summing the variables and indexes as necessary. For MARKAL this means that each variable associated with a technology simply needs to have its <var>_B/C entries summed and reported such that

$$\begin{aligned} \text{BenefitCostRatioT}(tch, t) &= (\text{sum}(\text{indexes}, \text{UnitBenefit}(j,t))/(\text{sum} \text{UnitCost}(j,t)) \\ \text{BenefitCostRatio}(tch) &= (\text{sum}(\text{indexes}, \text{UnitBenefit}(j))/(\text{sum} \text{UnitCost}(j)) \end{aligned}$$

Using the New Benefit/Cost Mechanism

To use the new approach the user must:

1. have GAMS22.0 or move the new GAMS Convert executable (GMSCV_NX.EXE, contact ggoldstein@irglt.com if needed) into your GAMS system folder [**saving your current version 1st**];
2. have a CONVERT.OPT file in the GAMS_WrkXXX folder with a single entry UnitBC (move it from the GAMS_SrcPRDv55 ZIP file), and
3. provide \$SET CBNEW 'YES' in the <case>.GEN file for single region runs, and same in both the <case>.SLV and each <case_reg>.RPT file.

Note that when CBNEW is activated the RUNRPT.INC COSTBEN flag is automatically set in the code to request the Cost/Benefit report be run.

Also, there is an additional option (\$SET CBALL 'YES') that can be provided (in the single region GEN or multi-region RPT files) to request that the report be generated for all TCH if desired. This eliminates the need for the COSTBEN.DD file with the list of TCH (by type) to be handled, but results in a large COSTBEN table for big models.

Open Issues

In addition, there is some discussion underway regarding how to reflect any Cost associated with RESIDuals. To this end GianCarlo has recommended that the sense of the EQ_CPT for all TCH (not just DMD) be =L=, rather than =E= for CON/PRC. There is the \$SET CPTL 'L/E' option available to control this, and with the next full release the default will be changed to 'E' (from 'L'). At the moment there is an associated option for the New Benefit/Cost report \$SET CBCPT 'YES' to have the RESID * EQ_CPTi.M added to the cost if said product is < 0. Once a consensus is reached on both of these points the CPTL will be changed (as noted) and the need for CBCPT dropped if always to be included in the ratio calculation.

GAMS Requirements

The new GAMS UnitBC facility, necessary for the new Benefit/Cost ratio report (COSTBNEW), requires that the user have a current (June 2004 or later) GAMS license, and GAMS21.8 or higher installed.

----- International Resources Group -----

Implementation³

In order to introduce the new mechanism into MARKAL (or TIMES, or any other model) the programming procedure requires:

1. that OPTION LP=CONVERT be activated followed by a SOLVE for the model for which the UnitBC.GDX file is to be produced;
2. each <var>_B and <var>_C parameter must be declared;
3. an EXECUTE_LOAD 'UnitBC' command be provided requesting that each <var>_B/C parameter be loaded⁴;
4. the appropriate combination of the relevant <var>_B/C parameters must be summed, and
5. the Benefit/Cost ratio calculated and reported.

The MARKAL code for steps 3-5 are embodied in the COSTBNEW.INC file reproduced in Appendix A.

3.2 Commodity Flow Level User Constraint Parameter (ADRATIO)

The TIMES model conveniently allows user constraints to include the flow variables associated with commodities and processes. In MARKAL the “variables” tracking commodity flows through processes vary according to the nature of the technology (resource, process (regular/limit), power plants (electric-only, coupled heat and power, heat-only), and thus the actual flows need to be determined by apply the associated balance equation coefficient for each variable type. The commodity-based emissions accounting (via the ENV_ENT parameter) mechanism employs the same approach.

User data entry is via a single parameter **VRAT_FLO(adratio,tch,ent/dm,year)** that contains the multiplier to be applied to the appropriate variable/balance equation coefficient as determined (by the code) according to the nature of the technology.

4.0 Updates and Corrections

Various minor adjustments and corrections were made to the code.

³ Note that the current GAMS approach is an R&D (beta) implementation, and will be refined. This may necessitate adjustments to the implementation.

⁴ Note that if a particular variable did not have any instances in the model solved the associated <var>_B/C parameter will not appear in the UnitBC.GDX file and an execution error will occur. However, the MARKAL example shows how this is handled by checking for some instance of each variable before requesting the LOAD.

4.1 Ignore QC In/Out checks for DUM processes

4.2 Set low-temperature heat seasonal transmission to default = 1, to avoid /0 execution error

4.3 Interpolate more parameters (e.g., ENV_ENT, GROWTH, DECAY)

4.4 Adjust GAMS control files for move to _<case>.G00 save files

4.5 Some adjustments and minor “bugs” fixes

- Test explicitly for ‘INF’ on trade bounds
- Correct peaking equation entries for vintaged demand devices
- Correct Export consumption reported in T08
- QC RAT_ACT for CON/DMDs
- Problem with TCHAFCK on multi-region run if the CLEAR was missing for BAT-mode corrected
- Additional output to VEDA-BE including:
 - Suppress all 0 lines going to the <case>.VD file (reducing file size by 40%)
 - Emission marginals and global emissions information output for SAGE
 - MGA results for US-EPA
 - Longer dump file page size

4.6 Outstanding problems

- Some systems are not pausing/halting when an error is encountered?

For more information on the ETSAP model generators, MARKAL and TIMES, contact the ETSAP Primary Systems Coordinator (Gary Goldstein, DecisionWare, Inc, ggoldstein@irglt.com).

Appendix A: New Benefit/Cost Routine

```

*=====*
*  COSTBNEW new UnitB/C *
*  %1 - "" or _R *
*  %2 - "" or REG, *
*v5.4h use GAMS UnitBC.GDX mechanism to get the variables *
*  UnitBenefit/Costs *
*=====*
$IF NOT '%MODVER%' == 'CMD' $GOTO NOCBN
* check that a value for each variable, else no GDX parameter is
*   created and an EXECUTE error is encountered
* Activity
if(sum(tpprc(tp,prc),ACT.l(tp,prc)),
  execute_load 'unitbc' %1ACT_B, %1ACT_C );
* Capacity
if(sum(tptch(tp,tch),CAP.l(tp,tch)),
  execute_load 'unitbc' %1CAP_B, %1CAP_C );
* Investment
if(sum(tptch(tp,tch),INV.l(tp,tch)),
  execute_load 'unitbc' %1INV_B, %1INV_C );
* Limit process output
if(sum((tpprc(tp,prc),enc),LOUT.l(tp,prc,enc)),
  execute_load 'unitbc' %1LOUT_B, %1LOUT_C );
* Conversion plant maintenance
if(sum((tpcon(tp,con),z),M.l(tp,con,z)),
  execute_load 'unitbc' %1M_B, %1M_C );
* Coupled heat and power pass-out turbines
if(sum((tpcpd(tp,cpd),z,y),TCZYH.l(tp,cpd,z,y)),
  execute_load 'unitbc' %1TCZYH_B, %1TCZYH_C );
* Electricity from power plants
if(sum((tpela(tp,ela),z,y),TEZY.l(tp,ela,z,y)),
  execute_load 'unitbc' %1TEZY_B, %1TEZY_C );
* Heat from heating plants
if(sum((tphpl(tp,hpl),z),THZ.l(tp,hpl,z)),
  execute_load 'unitbc' %1THZ_B, %1THZ_C );
;
$IF %RUN_MREG% == 'YES' LOOP(REG,
* UnitBenefit of technology
ub(tp,tchx) = %1act_b(%2tp,tchx) + %1cap_b(%2tp,tchx) +
              %1inv_b(%2tp,tchx) + sum(enc,%1lout_b(%2tp,tchx,enc)) +
              sum(z,%1m_b(%2tp,tchx,z)) +
              sum(td,%1tczyh_b(%2tp,tchx,td) + %1tezy_b(%2tp,tchx,td))+
              sum(z,%1thz_b(%2tp,tchx,z));
* UnitCost of technology
uc(tp,tchx) = %1act_c(%2tp,tchx) + %1cap_c(%2tp,tchx) +
              %1inv_c(%2tp,tchx) + sum(enc,%1lout_c(%2tp,tchx,enc)) +
              sum(z,%1m_c(%2tp,tchx,z)) +
              sum(td,%1tczyh_c(%2tp,tchx,td) + %1tezy_c(%2tp,tchx,td))+
              sum(z,%1thz_c(%2tp,tchx,z));
$IF %RUN_MREG% == 'YES' );

$IF NOT %CBCPT% == 'YES' $GOTO NORESID
* handle RESID

```

----- *International Resources Group* -----

```

uc(tpdmd(tp,dmx))$(tch_res(dmx,tp) * eq_cpt1.m(tp,dmx)) < 0) =
uc(tp,dmx) + TCH_RES(DMX,TP) * EQ_CPT1.M(TP,DMX);
uc(tpcon(tp,conx))$(tch_res(conx,tp) * eq_cpt2.m(tp,conx)) < 0) =
uc(tp,conx) + TCH_RES(CONX,TP) * EQ_CPT2.M(TP,CONX);
uc(tpprc(tp,prx))$(tch_res(prx,tp) * eq_cpt3.m(tp,prx)) < 0) =
uc(tp,prx) + TCH_RES(PRX,TP) * EQ_CPT3.M(TP,PRX);
$LABEL NORESID

* calculate the Ratio
ratio(tp,tchx)$(uc(tp,tchx) GT .0000001) = ub(tp,tchx) / uc(tp,tchx);
ratiot(tchx)$(SUM(tp,uc(tp,tchx)) GT .0000001) = sum(tp,ub(tp,tchx)) /
sum(tp,uc(tp,tchx));

display '*** New UnitB/C ***', ub, uc, ratio, ratiot;

$LABEL NOCBN

```