Energy Technology Systems Analysis Programme

http://www.etsap.org/tools.htm

Documentation for the MARKAL Family of Models

October 2004

Authors:

Richard Loulou
Gary Goldstein
Ken Noble

Part III: System for Analysis of Global Energy markets (SAGE)

5. Concepts and Theory 420

5.1 Introduction 420

5.2 The Time-stepped computation of the SAGE Equilibrium 420

5.3 The Market Sharing Mechanism 423
5.3.1 The 'knife-edge' effect 423
5.3.2 The SAGE Market sharing algorithm 424

5.4 Modeling and Interpretation Issues 426
5.4.1 Cumulative constraints 426
5.4.2 Interpretation of SAGE shadow prices 427
5.4.3 Technological Learning in SAGE 427

6. Reference Guide for SAGE 429

6.1 Sets, Switches and Parameters 429

6.2 Variables 434

6.3 Equations 434
6.3.1 MR_MTSOBJ(r,t) 434
6.3.2 MR_OBJ 435
6.3.3 Inter-period Dynamic Market Share and Endogenous Technology Learning Facilities 436
6.3.3.1 Market Share 436
6.3.3.2 Input parameters needed 436
6.3.3.3 Market Share Algorithm 439
6.3.4 Endogenous Technology Learning 441
6.3.4.1 Input parameters needed 441
6.3.4.2 Endogenous Technology Learning Algorithm 443

List of tables

Table 6-1. Definition of SAGE user input data 430
Table 6-2. SAGE variables 434

5. Concepts and Theory

5.1 *Introduction*

The SAGE variant of the MARKAL family was conceived to introduce new behavioral elements into the model's partial equilibrium paradigm. In Standard MARKAL, each agent maximizes its long-term discounted surplus, while being subject to marginal value pricing of all commodities, thus simulating far-sighted, competitive energy markets. The Standard MARKAL paradigm is adapted to many uses, including the assessment of technologies and of and public policies from a social viewpoint. The SAGE variant[102] takes as its main objective a simulation of markets where certain assumptions of perfect competition do not hold. It is also based on the computation of a partial equilibrium on energy markets, but the latter differs from the MARKAL equilibrium in two major ways:

- First, SAGE assumes that economic agents have a limited knowledge of the future, and therefore attribute much more weight to short term than to long term surpluses. This leads to optimizing the surplus over the current period rather that over the entire horizon, as explained in section 5.2.

- The second distinctive assumption made by SAGE is that each agent in the model is not willing or able to exactly maximize its surplus, either because it faces uncertain costs and prices, or because its behavior is not that of a pure profit maximizer (e.g. there exist non monetary costs that influence its investment decisions). This second characteristic of imperfect markets has been represented in several different ways in various simulation models. In SAGE, it is modeled via a *market sharing algorithm* that attributes non zero market share to technologies that are not optimal but are in some way judged to be competitive and thereby close enough to the optimal choice(s) to deserve a share of the market. The closeness to competitiveness is determined using an original approach based on dual information. This mehanism is discussed in section 5.3.

In the rest of this chapter, we describe how these two major differences between SAGE and Standard MARKAL alter the computation of the equilibrium.

5.2 *The Time-stepped computation of the SAGE Equilibrium*

The SAGE assumption is that all agents have a planning horizon (and an information set) limited to the current period. The direct implication is that each agent optimizes its

[102] SAGE was developed by the US Dept of Energy's Energy Information Administration. The full stand-alone documentation and a description of its database may be found from: http://www.eia.doe.gov
Andy Kydes (DOE EIA) was at the origin of the two distinctive features that differentiate SAGE from MARKAL.

surplus over that period only. Beyond the current period, it is assumed that agents have no knowledge of future parameters.

In mathematical terms, this means that the SAGE objective function is equal to Σ_r***ANNCOST(r,t)*** (i.e. the sum of regional annualized costs, see below), to be minimized for each *t* sequentially as follows: in period 1, the objective is to minimize Σ_r***ANNCOST(r,1)***, knowing only the parameters of period 1 and the initial state of the system at beginning of period 1. This first period optimization produces a set of decisions for that period. At period 2, the decisions found for period 1 are frozen, and this determines the initial conditions for period 2 (i.e. capacities resulting from past investment decisions, residual lives, reserves, etc). Then the model optimizes Σ_r***ANNCOST(r,2)*** subject to these initial conditions. The same process is repeated *T* times, until the last period of the horizon (*T*) is reached.

Such a sequential, myopic optimization process is said to be *time-stepped*, as opposed to the global, inter-temporal optimization used in Standard MARKAL. As a consequence, the SAGE objective function does not require any discounting. However, once the model has been solved for all periods, it is possible to compute a total discounted cost over the whole horizon, if desired (for instance for the purpose of comparing the cost to that of a standard MARKAL run).

We reproduce below for completeness the ***ANNCOST(r,t,)*** expression already shown in section 1.4 of PART I.

$$
\begin{aligned}
ANNCOST(r,t) = \Sigma_k \Big\{ & Annualized\_Invcost(r,t,k) * INV(r,t,k) \\
& + Annualized\_Invcost(r,t,k) * Resid(r,t,k) \\
& + Fixom(r,t,k) * CAP(r,t,k) \\
& + Varom(r,t,k) * \Sigma_{s,s} ACT(r,t,k,s) \\
& + \Sigma_c [Delivcost(r,t,k,c) * Input(r,t,k,c) * \Sigma_s ACT(r,t,k,s)] \Big\} \\
+ \Sigma_{c,s} \Big\{ & Miningcost(r,t,c,l) * Mining(r,t,c,t) \\
& + Tradecost(r,t,c) * TRADE(r,t,c,s,i/e) \\
& + Importprice(r,t,c,l) * Import(r,t,c,l) \\
& - Exportprice(r,t,c,l) * Export(r,t,c,l) \Big\} \\
+ \Sigma_c \ \{ & Tax(r,t,p) * ENV(r,t,p) \} \\
+ \Sigma_d \ \{ & DemandLoss(r,t,d) \}
\end{aligned}
$$

where:

Annualized_Invcost(r,t,k) [103] is the annual equivalent of the lump sum unit investment cost, obtained by replacing this lump sum by a stream of equal annual payments over the life of the equipment, in such a way that the present value of the stream is exactly equal to the lump sum unit investment cost, for technology ***k***, in period ***t***. Note carefully that by stopping the summation over ***t*** at the end of the horizon, the objective function automatically accounts for the salvage value of all assets stranded at the end of the horizon. Note also that investments made prior to the first period are also annualized and added to the objective function, in order to better reflect the impact of these sunk investments on prices.

Fixom(k,t,r), Varom(r,t,k), are unit costs of fixed and operational maintenance of technology ***k***, in region ***r*** and period ***t***;

Delivcost(r,t,k,c) is the delivery cost per unit of commodity ***c*** to technology ***k,*** in region ***r*** and period ***t***;

Input(r,t,k,c) is the amount of commodity ***c*** required to operate one unit of technology ***k,*** in region ***r*** and period ***t***;

Miningcost(r,t,c,l) is the cost of mining commodity ***c*** at price level ***l,*** in region ***r*** and period ***t***;

Tradecost(r,t,c) is the unit transport or transaction cost for commodity ***c*** exported or imported by region ***r*** in period ***t***;

Importprice(r,t,c,l) is the (exogenous) import price of commodity ***c,*** in region ***r*** and period ***t***; this price is used only for exogenous trade, see below;

Exportprice(r,t,c,l) is the (exogenous) export price of commodity ***c,*** in region ***r*** and period ***t***; this price is used only for exogenous trade, see below;

Tax(r,t,p) is the tax on emission ***p,*** in region ***r*** and period ***t***; and

DemandLoss(r,t,d) represents the welfare loss (in non reference scenarios) incurred by consumers when a service demand ***d,*** in region ***r*** and period ***t,*** is less than its value in the reference case. This quantity was derived in subsection 1.6.2.2 of PART I

---

[103] The annualized unit investment cost is obtained from the lumpsum unit investment cost via the following formula:

$$\boldsymbol{ANNUALIZED\_INVCOST} = \boldsymbol{INVCOST} / \sum_{j=1}^{\boldsymbol{LIFE}} (1+\boldsymbol{h})^{-j}$$

where:
INVCOST is the lumpsum unit investment cost of a technology
ANNUALIZED_INVCOST is the annualized equivalent of INVCOST
LIFE is the physical life of the technology
h is the discount rate used for that technology, also called hurdle rate. If the technology specific discount rate is not defined, the general discount rate d is used instead.

The hurdle rate **h** may be technology, sector and/or region specific, so as to reflect the financial characteristics the analyst deems appropriate for each investment decision. For instance, if the initial capital cost of a car is $20,000, and if its technical life is 10 years, the annualized value of the capital cost assuming an 8% discount rate must be such that a stream of 10 such annualized payments adds up, after discounting, to exactly $20,000. The equivalent annualized value is $3,255, as computed using the expression above.

5.3 The Market Sharing Mechanism

The market sharing mechanism is being reviewed and will perhaps be altered by the EIA team in charge of the SAGE model. The approach described below may therefore be replaced by a new or altered one in the next few weeks or months

5.3.1 The 'knife-edge' effect

SAGE possesses a mechanism that allows the user to bound the market share that any single technology or group of technologies is allowed to capture in its own market, in a given time period. A typical example of an end-use market is the set of technologies providing the same demand segment in a given region at a specific time period (e.g., all residential space heating technologies in USA at period 5).

Without such a mechanism, it would be possible for the model to select a single technology to invest in, in a given time period (e.g. the standard gas furnace might capture all new residential space heating demand in period 5, in the US region). This might happen even if the cost of the winner technology were only slightly lower than the cost of competing technologies. This *winner-takes-all* phenomenon, also called *penny-switching* or *knife-edge* effect, results from the linearity of the surplus expression in MARKAL and other LP based models.

In real markets, it is more often observed that end-users' investment choices in a given end-use market are spread over several technologies, mainly for two reasons. The first is that individuals make investment decisions based on preferences other than pure financial costs (e.g. cleanliness, convenience, safety, etc.). The second reason is that there is more variety within a region than is captured by the model, and this variety (due to local conditions) introduces a cost differentiation that is ignored by the model. The latter reason is all the more pertinent if the region being modeled is large and includes several countries or provinces. These two causes, and perhaps others, make it unlikely that all end-users would select the same technology.

One remedy to this phenomenon is to introduce lower or upper bounds on the market shares of technologies in the same market (for instance, upper bounds on gas fired heaters, oil fired heaters, and electric heaters in the residential space heating market at each time period, or lower bounds on solar heaters, or a combination of these). Such bounds may be of the simple, purely exogenous variety, or may be based on a more sophisticated analysis of the relative costs of the competing technologies.

Whereas the first approach (exogenous bounds) may be implemented at will by any MARKAL modeler via regular bounds, it may appear to be too rigid (some flexibility may be introduced by the use of user-constraints that prescribe limits for "baskets" of similar technologies, but hard bounds may still be needed for the "winners" if others in the group are to gain a share of the market). The second approach is a distinctive feature of the SAGE model variant which is used primarily for the end-use technologies, but may also be applied to the supply technologies (e.g., a set of competing renewable technologies). We now describe specific SAGE market sharing mechanism. The complete specification of parameters, variables, and equations is provided in chapter 6.

5.3.2 The SAGE Market sharing algorithm

The objective is to consider those technologies that are at their lower bounds in the SAGE initial solution, and to select among them those that will be receiving some market share. First note that technologies at their lower bounds have a positive reduced cost (see PART I, section 1.7).

The mechanism comprises the following steps:

1. The modeler must first identify the various groups of competing technologies, which we call the *markets*, denoted ***M***. For instance, the automobile market contains all passenger vehicles, which compete for satisfying the auto travel demand segment.
2. The user next defines a *closeness* value *E*, such that if a technology has a reduced cost exceeding *E*, it will not capture any market share (see below). This value may be expressed in absolute or in relative terms (as a percentage of the cost coefficient of the basic technology). Default value = 0.2.

Now, for each period *t=1, 2, .., T,* the following steps are implemented:

3. Initial model run: The model is solved for period *t*, and several quantities are recorded: first the total size of each market ***M***, denoted $\boldsymbol{TOTMKTSIZ_M}$. Next, the reduced costs of the technologies in each market (denoted by B_k for technology *k*). In each market, some technologies are at their upper bound or are basic in the initial solution. These technologies have a negative reduced cost (if at the upper bound) or a zero reduced cost (if basic), and they already have a non-zero market share. All other technologies have a positive reduced cost.[104] Technologies that have a positive reduced cost which is smaller than the threshold *E* are now *candidates* for receiving some market share. Let ***G*** be the generic name of a group of candidates.
4. Market shares: For each market M, we now compute the market share for each candidate technology ($\boldsymbol{MKTSHR_k}$), in two stages: first, the total market available for reallocation to the candidates ($\boldsymbol{MKTSIZ_M}$) is defined as a user defined fraction ***C*** of the total market size ($\boldsymbol{TOTMKTSIZE_M}$). Then the market share of each candidate technology is defined.

$$\boldsymbol{MKTSIZ = C \bullet TOTMKTSIZ}$$

and:

[104] Unless there are multiple optimal solutions, *in which case some technologies which are at their lower bound have a zero reduced cost.* The occurrence of multiple solutions is unlikely and is not treated in the market share mechanism.

$$MKTSHR_k = \frac{a_k \cdot B_k^{-d}}{\sum_{i \in G} a_i \cdot B_i^{-d}}$$

where:

- ***C*** is a user-defined fraction (between 0 and 1) that defines the fraction of the total market ***M*** will be re-allocated to the candidate technologies. The smaller ***C*** is, the closer SAGE is to a purely optimizing model. Default value: 0.2
- a_i is user defined preference for technology *i.* This parameter serves to represent a non monetary preference, assigned by the user, that may be attached to technology *i*. Default value: 1.
- B_i is the reduced cost of technology *i*, obtained from the initial SAGE run for the current period (step 3).
- ***d (> 0)*** is a user-defined parameter which determines the degree to which reduced costs alone determine the re-allocation. The larger ***d*** is, the more predominant are the reduced costs in determining the re-allocation. For very small values of ***d***, the reduced costs play no role, and the market shares are determined only by the preferences a_k. For residential, commercial and transportation markets, it is recommended that ***d*** be less than 0.3, since these are sectors with many small agents, where hidden costs are important. For industrial segments, ***d*** should be larger. Values of 2 or more are recommended for sectors with large firms such as electric utilities and oil refineries. Default value: 0.2.
- ***G*** is the group of candidate technologies to which technology *k* belongs (defined at step 3).

5. Final solution: each candidate technology is assigned an lower bound:

$$LB_k = MKTSHR_k \bullet MKTSIZ_M$$

where ***M*** is the market containing technology ***k***, and $MKTSIZ_M$ was obtained at step 3. Then the model is re-solved for period *t* to provide the final solution for that period.

Integrity Test: If any lower bound derived above exceeds the level obtained by any basic or upper bounded variable in the same market, then a 2nd pass at assigning market shares is done as follows:

- all such "winners" (i.e. technologies that are basic or at their upper-bound) having lower values than the non-winners' lower bounds are added to the candidates pool; the candidates' market size is increased by their combined values;
- each of the previous "winners" is then assigned a fictitious reduced cost slightly smaller than the smallest reduced cost of the "non-winners" (non basic technologies at their lower bound), and
- the market shares are re-computed as described above.

This insures that the "winners" will get a higher market share than any of the "non-winners".

6. If ***t=T*** , STOP;
otherwise: go to step 3 with: $t \leftarrow t+1$

Remark: there are three potential variants of the market share mechanism, depending on what reduced cost is selected, as follows:

- INV variant uses the reduced costs of investment variables;
- INVPCT variant uses the ratio of investment reduced cost over current investment cost, and
- ACT variant may only be used for demand devices (dmd) and uses the reduced costs of the capacity variable (which also serves as the device activity variable.

5.4 Modeling and Interpretation Issues

5.4.1 Cumulative constraints

In SAGE, the agents have no knowledge of the future beyond the current period. Therefore, when SAGE minimizes the current period's annual cost, it cannot directly take into account any constraint that extends over future periods. To illustrate this, consider the following example:

> Assume a SAGE model with a horizon of 30 years composed of 3 periods of 10 years each, and assume a cumulative emission limit of 8 Gt over the whole horizon. At period 1, the model is only concerned with minimizing the cost for that period. Therefore, it will very likely take no heed of the emission constraint, since the minimum cost for period 1 is reached when emissions in that period are unrestricted. Suppose that the optimization for period 1 results in annual emissions of 500 Mt in period 1 (i.e. 5 Gt for the whole period). Therefore, by the beginning of period 2, the remaining cumulative emission bound for periods 2 and 3 is 3Gt. The model then optimizes period 2 subject to an emission limit of 3 Gt, which happens to be less than the unrestricted emissions in period 2. Therefore, emissions in period 2 are exactly equal to 3 Gt. At this point, the cumulative limit has been reached, and period 3 has a zero emission limit, which is clearly unacceptable, and probably infeasible as well! Therefore, some problem instances that might solve with the clairvoyant standard MARKAL, may prove to be unsolvable via SAGE.

SAGE is by no means the only model with such a limitation. All models that treat the horizon in a time-stepped manner (and this includes all simulation models, as well as all static equilibrium models), have the same difficulty with cumulative constraints (indeed with any constraint requiring the model to balance present and future actions).

For these models, there are two alternatives to a cumulative emission constraint. The first alternative is for the modeler to impose a series of annual limits (summing up to the cumulative limit), but this eliminates the flexibility of optimally spreading the emissions over the horizon. The second alternative is to impose an emission tax at each period. However, this is only a partial remedy, as it does not allow the precise representation of a cap-and-trade policy, except via a lengthy trial-and-error search for tax values at each period.

Similarly, SAGE will not properly respond to a constraint representing the total reserves of a resource such as gas or crude oil. It will use what it wants/needs in each period until the resource is depleted without any planning ahead for this eventuality, until the period when the reserve is exhausted.

5.4.2 Interpretation of SAGE shadow prices

In Standard MARKAL, the objective function is the total discounted cost of the system. Therefore, the shadow prices have the dimension of a discounted price, which are later un-discounted by a back-end program. In SAGE, the objective function at each time period is the annual cost in that period. Therefore the shadow prices are not discounted, and may be reported as is.

5.4.3 Technological Learning in SAGE

The SAGE endogenous technology learning algorithm employs the same representation that was used in Standard MARKAL, whereby the unit investment cost of a technology is assumed to be reduced by a certain amount for every doubling of the capacity. The progress ratio ***pr*** is the key parameter that controls this relationship.

However, the time-stepped nature of SAGE does not permit to truly endogenize the learning process, since the model has no anticipatory capability. However, it is still possible to adjust the unit investment costs of learning technologies as a function of their cumulative investment at each period ***t***. Indeed, after each period's optimization and after all market share adjustments have been addressed, the solution for the current period is examined and the investment cost associated with each learning technology is calculated and injected into the model for use in the next period's optimization.

In SAGE, international and cross-technology spillover is also modeled, and controlled by user selected parameters. The unit investment cost of each learning technology is thus related to the cumulative investment in that technology plus the weighted sum of cumulative investments in related technologies in the current region and in other regions. The weights used to calculate this weighted sum are fully under user control. The basic formula is as follows:

In this form, we accommodate international and cross-technology learning by means of a "spill" parameter (LSPILL_R(r',t,r,j,year)), which is the fraction of the capacity of technology (t) in one region (r') contributes to learning in the current region (r) of the learning technology (j). Note that the code sets LSPILL_R(r,j,r,j) = 1 (that is the learning technology learns from itself).

$$\boldsymbol{INVCOST}_{k,r,t} = \boldsymbol{a} \cdot \left[\boldsymbol{C}_{k,r,t} + \sum_{j \in J,\, r' \in R} \boldsymbol{spill}_{j,r',k,r,t} \cdot \boldsymbol{C}_{j,r',t} \right]^{-b} \quad , \quad \text{for each } \boldsymbol{t} \in \boldsymbol{T}, \boldsymbol{j} \in \boldsymbol{J}, \boldsymbol{r} \in \boldsymbol{R}$$

where:

$\boldsymbol{spill}_{j,r',k,r,t}$ ($0 \le \boldsymbol{spill}_{j,r',k,r,t} \le 1$) is the user chosen index of spillover of technology ***j*** in region ***r'*** onto technology ***k*** in region ***r***. It is assumed that $\boldsymbol{spill}_{k,r,k,r,t} = 1$, i.e. that technology ***k*** has a maximum spillover onto itself in its own region

$\boldsymbol{C}_{j,r,t}$ is the cumulative investment in technology ***j*** in region ***r***, up to period ***t***. Note that the spillover index may vary over time.

6. Reference Guide for SAGE

As discussed in Part III, Chapter 5, SAGE, the System for Analysis of Global Energy markets, is a specialized version of MARKAL adapted for use by the US Department of Energy's Energy Information Administration as the analytic foundation for the annual publication of the *International Energy Outlook*. The main two differences between SAGE and Standard MARKAL are first that SAGE is run in a time-stepped manner, that is myopically solving for each model period in sequence, whereas MARKAL employs perfect foresight as it clairvoyantly solves for the entire modeling horizon at once, and second that SAGE employs a market sharing algorithm that modifies the least cost criterion used by MARKAL. SAGE also has available a technology learning component similar to Standard MARKAL, but with one important difference due to the near-sighted nature of the model. SAGE adjusts the investment cost between each period based upon the cumulative capacity thus far installed for technologies with learning. Indeed, without the look-ahead capability of MARKAL, SAGE cannot anticipate the need to invest early in such technologies so the learning mechanism is only of a truly endogenous nature.

In the next section the switch and parameters that activate and control SAGE are presented, followed by the variables and equations. The two main SAGE features are discussed in the last two sections.

6.1 Sets, Switches and Parameters

Like all other aspects of MARKAL the user describes SAGE by means of Switches and with the Sets and Parameters described in this section. Table 6-1 below describes the User Input data, and the Matrix Coefficient and Internal Model Sets and Parameters, respectively, that are associated with the SAGE option.

Table 6-1. Definition of SAGE user input data

Input Data (Indexes)	**Alias/Internal Name**	**Related Parameters**	**Units/Range & Defaults**	**Instance (Required/Omit/ Special Conditions)**	**Description**
AMSWTCH (ADRATIO, AMS) (a/m)		*ADRATIO* *MKT_ID*	• Indicator. • ['A','B','X']; default = 'B'.	• Provided for each ADRATIO for which there is also a market share MKT_ID.	Indication of how to treat individual user-defined constraints with respect to the market share groups, according to which of the three values for the ADRATIO/MKTSHR switch is activated as described here. • AMS = 'A', always generate the user-defined constraint (MR_ADRATn). • AMS = ' B', generate as a user-defined constraint (MR_ADRATn) or handle via the market share algorithm depending upon whether MKTSHR is active. [Default, if MKTSHR is active suppress generation of the MKT_ID=ADRATIO MR_ADRATn equations.] • AMS = 'X', ignore the user-defined constraint (MR_ADRATn) if MKTSHR is active, regardless of whether or not a corresponding MKT_ID exists.
LSPILL[105] (p,p,t) [L]			• Fraction. [0.01 - 1], none.	• Provided if SETL is activated (see section 6.3.4.1) for learning technologies that have learning components.	The amount of learning that "spills" from one technology to another to contribute to the latter's learning rate. There is also a regional form of LSPILL_R that handles "spill" between regions and technologies. • Results in a portion of the cumulative investment of a learning component being taken into consideration when determining the current costs for a "dependent" learning technology.
MKT_GRP *(m,p)*			• Set. • [market groups and tch];	• Provided to define the members of a market group.	The mapping of the individual technologies subject to the market share algorithm (when MKTSHR is activated, see section 6.3.3.2) indicating the group to which the

[105] The user is referred to section 2.8 where the rest of the technology learning related parameters are presented.

Input Data (Indexes)	Alias/Internal Name	Related Parameters	Units/Range & Defaults	Instance (Required/Omit/ Special Conditions)	Description
			default none.		technology is assigned.
MKT_ID			• Set. • [market groups]; no default.	• Provided to define each market group.	The set of user provided names for the groups of technologies to be involved in the market share algorithm (when MKTSHR is activated, see section 6.3.3.2). MKT_IDs often correspond directly to the user-defined constraint (adratio) that they substitute for when the market share algorithm is activated.
MKT_CE (m)			• Either fraction (if MKTSHR = 'INVPCT') or monetary units (if 'INV') or activity units (if 'ACT'). • [depends on MKTSHR type], default = 0.2.	• Required if MKTSHR is activated (see section 6.3.3.2.) for each MKT_ID group, unless default will do. • Must be provided for each group if MKTSHR is not 'INVPCT'.	The "close enough" criteria qualifying a technology to get a share of a market based on how close to being competitive the technology is when it does not make it into the preliminary solution for a period. • Involved in each market share group evaluation to determine the candidates to be included in the re-assignment of the shares for the competing technologies.
MKT_GAMA (m)			• Scalar. • [0.1-5]; default = 2.	• Provided if MKTSHR is activated (see section 6.3.3.2) for each MKT_ID group, unless default will do.	The degree of optimization to be applied when determining the share for qualifying candidates in a market share group. • Involved in each market share group evaluation to determine how much of a share each of the candidates is to receive.
MKT_LO (m)			• Either fraction (if MKTSHR = 'INVPCT') or monetary units (if 'INV') or activity units (if 'ACT').	• Provided if MKTSHR is activated (see section 6.3.3.2) for each MKT_ID group, unless default will do.	Tiny lower bound applied to the investment variable (R_INV) if MKTSHR is 'INV' or 'INVPCT', or capacity (R_CAP) if MKTSHR is 'ACT'. • Applied to each MKT_GRP technology to force a marginal value on each candidate.

Input Data (Indexes)	Alias/Internal Name	Related Parameters	Units/Range & Defaults	Instance (Required/Omit/ Special Conditions)	Description
			• [depends on MKTSHR type], default = 0.00001.		
MKT_PREF (m,p)			• Fraction. • [0.001 - 5]; default = 1.	• Provided if MKTSHR is activated (see section 6.3.3.2) for each MKT_ID group, if a weighted preference is desired unless default will do.	A preference or weighting factor applied to an individual technology when determining its share as part of the market share algorithm. • Involved in each market share group when determining how to split the reallocation group and how much of a share the candidate is to receive.
MKT_REAL (m)			• Fraction. • [0.001 - 1]; default = 0.2.	• Provide if MKTSHR is activated (see section 6.3.3.2) for each MKT_ID group, unless default will do.	The size of the market to subject to the reallocation algorithm. • Involved in each market share group when determining how big the market to be reallocated is to be.
MKT_RARC (m)			• Fraction. • [0.000001 - 1]; default = 0.1.	• Provided if MKTSHR is activated (see section 6.3.3.2) for each MKT_ID group, unless default will do.	The size of the market to subject to the second reallocation pass if one of the reallocated shares exceeds the smallest basic technology that had penetrated during the initial solve. • Involved in each market share group second reallocation algorithm.
MKTSHR			• Switch. • [‘NO’, ‘ACTPCT’]; default = ‘NO’.	• Provided to activate the Market Share algorithm.	The switch that activates the market share algorithm, and indicates which algorithm is to be applied (see section 6.3.3.2). • When activated MKTSHR results in two model SOLVEs being done for each period, where the second solve has the market share rules imposed and

Input Data (Indexes)	Alias/Internal Name	Related Parameters	Units/Range & Defaults	Instance (Required/Omit/ Special Conditions)	Description
					thereby technologies bounded to their “appropriate” level.
TEG (p)			• Set. • [tch]; no default.	• Provided to identify the technologies subject to technology learning.	Those technologies subject to the endogenous technology learning algorithm when SETL active (see section 6.3.3.2).

6.2 Variables

The only two additional variables needed for SAGE are both related to the objective function. The variables are presented in Table 6-2 below.

Table 6-2. SAGE variables

VAR Ref	Variable (Indexes)	Variable Description
MTS.1	R_MTSOBJ (r,t)	The total producer/consumer surplus, embodying the annualized least-cost configuration of the energy system associated with each region in period t.
MTS.2	RMTSOBJ	The overall model objective function in period t . The period index t is omitted because the problem is solved one period at a time.

6.3 Equations

As SAGE is essentially core MARKAL modified to solve each period in sequence, rather than simultaneously, all of the equations of the model with the exception of the regional and overall objective function are the same as those described in sections 2.4 and 2.5. SAGE employs annual energy costs, and thus the main equation (MR_MTSOBJ) is much the same as the standard MARKAL MR_ANNCOST equation, but with a RHS variable (R_MTSOBJ, accumulating the total for each region) for inclusion in the overall objective function (MR_OBJ).

6.3.1 MR_MTSOBJ(r,t)

Description: Total annualized energy system cost associated with each region in each time period.

Purpose and Occurrence: To accumulate all the energy system related costs for each region and time period.

This equation is generated for each region and time period.

Remarks: This equation is essentially identical to the standard MARKAL MR_ANNCOST equation, but with a RHS variable that accumulates the total for each region for inclusion in the overall objective function.

GAMS Routine: MMEQOBJ.MTS

MATHEMATICAL DESCRIPTION

$$\text{EQ\# MTS.1: MR\_MTSOBJ}_{r,t}$$

All the components of MR_ANNCOST, see Chapter 2, section 2.5.2 for details.

• • •

$$\{=\}$$

Total energy system cost.

$$R\_MTSOBJ_{r,t}$$

6.3.2 MR_OBJ

Description: Total annualized energy system cost, over all regions, in the current period.

Purpose and Occurrence: To accumulate all the energy system related costs in the current period. This equation is established once for each period as the objective function.

GAMS Routine: MMEQOBJ.MTS

MATHEMATICAL DESCRIPTION

$$\text{EQ\# MTS.2 : MR\_OBJ}$$

Sum over each region's total cost for the current period.

$$\sum_{\boldsymbol{r}} \boldsymbol{R\_MTSOBJ_{r,t}}$$

$$\{=\}$$

Total energy system cost for all regions - the objective function variable to be minimized for each period sequentially.

$$\boldsymbol{RMTSOBJ}$$

6.3.3 Inter-period Dynamic Market Share and Endogenous Technology Learning Facilities

The SAGE market share mechanism is under revision by the Energy Information Administration. The description of this section is therefore subject to possible change in the next few months.

SAGE has embedded algorithms, under modeler control, that can be employed to examine the results of the last period solved and augment the solution by introducing new bounds on competing technologies or adjusting the investment cost of a technology based upon the cumulative capacity installed to date. The former capability is referred to as SAGE dynamic Market Share (MKTSHR) and the latter as Endogenous Technology Learning (SETL). The data requirements for both facilities were presented in Table 6.1 earlier in this chapter, and are repeated here, where the algorithm employed is also described.

6.3.3.1 Market Share

The market share facility allows the modeler to identify market segments or "groups of competing technologies" (e.g., all alternative fueled passenger vehicles), then employ an algorithm after each period is solved to redistribute a portion of each such market segment to technologies that are nearly competitive (as measured by a function of their reduced costs). Below each of the input parameters is explained in more detail than was provided earlier in the tables, and then the algorithm is elaborated.

6.3.3.2 Input parameters needed[106]

The input parameters are presented here in "logical" order based upon the order in which the modeler would tend to initiate, control and specify the Market Share algorithm and its components.

"SAGE Activation" To request time-stepped running of the model the SAGE switch must be activated in the <case>.GEN/SLV templates[107]. The market share algorithm is applied between periods during a time-stepped model run only. A fatal error will occur if the algorithm is requested and SAGE is not activated.

[$SET SAGE 'YES'/'NO', default 'NO']

"Market Share Algo." To request a SAGE run that uses the market share algorithm, this switch must be activated in the <case>.GEN/SLV templates[108]. There are currently three versions of the algorithm implemented.

---

[106] The input switch, set or parameter name, and default are mentioned in [] at the end of each component.
[107] This is accomplished at runtime by selecting MARKAL-SAGE from the VEDA-SAGE run form.
[108] Adjusted "semi-permanently" in VEDA-SAGE via the Run/Edit GEN/RegGEN/RegSLV option, or at Run time by requesting Edit of the GEN file before submitting. Make sure to include 'value'!!!

Only one may be selected currently for a SAGE time-stepped run although the option to choose a different one for each market group will be implemented later.
– '**INV'** – uses the reduced cost associated with the investment variable;
– '**INVPCT'** – uses (investment variable reduced cost/current investment cost); and
– '**ACT'** – uses the marginal cost associated with the activity of a technology (for DMDs only at this time!) The user should be aware that the 'ACT' option is currently limited in its use because of the implementation. Since an 'ACT' variable is the sum of all activities from a specific technology which may be available in multiple periods, a lower bound on such variables may not give the expected results since the technology may be attractive in one period but be seriously uneconomic in future periods.

Only one of these options may be selected per SAGE run.

[$SET MKTSHR 'INV'/'INVPCT'/'ACT', no default]

"Market Share Log"	Indication of whether the full or only a partial trace of the market share algorithm actions are to be reported in the MKTSHR.LOG file[109]. It is suggested that the 'ALL' option be used until the modeler has confidence that the case has been defined as intended, and that the modeler then switch to the 'MIN' option. [$SET MKTSHRLOG 'MIN'/'ALL', default 'ALL'].
"Market Group ID"	A list of the market group names corresponding to each of the clustered candidates to be evaluated for a share of each such group. Note that special treatment is given to the user-defined constraints when the market share is activated according to AMSWTCH, below. [MKT_ID(*)]
"Use-defined/ Market Share Control"	When the Market Share algorithm is activated certain groups may be in conflict with some user-defined constraints, and vice versa. vice versa. The handling of such situations is controlled according to the value provided by the modeler for AMSWTCH such that:

[109] Adjusted "semi-permanently" in VEDA-SAGE via the Run/Edit GEN/RegGEN/RegSLV option, or at Run time by requesting Edit of the GEN file before submitting. Make sure to include 'value'!!!

- 'A' - always generate the user-defined constraint, suppressing an associated MRK_ID group with the same name;
- 'B' – generate as a user-defined constraint when MKTSHR is not active or if no matching MKT_ID, but suppress if there is the same MKT_ID and the algorithm is activated; and
- 'X' - ignore the user-defined constraint if MKTSHR is active, regardless of MKT_ID match.

[AMSWTCH(adratio, 'A'/'B'/'X'), default 'B'.]

"Market Groups" The two-tuples of market group ID and technology that are to make up each market share group.

A critical test is performed to ensure that no technology appears in more than one market group. If this requirement is violated the model run is halted and a message presented on the screen and in the MKTSHR.LOG file. The reason the run is halted is that a different market share or lower bound would be developed from each group for the variable otherwise and it is unclear which market share should be honored.

[MKT_GRP(mkt_id,tch)]

"Close Enough" The definition of "close enough" to get a share of market is based on how close to being competitive a technology is. The "close enough" parameter is intended to partially correct for regional and market aggregation errors. For example, if we knew how much relevant prices varied in a region we could set "close enough" to be close to the variation in a region in $/MMBtu or $/PJ. Alternatively, we could use the rule that anything within 20% of the competitive price gets a share. Let B be the duals (reduced cost) of the technologies of interest; more accurately, let B be the measure of closeness used which could be the dual or reduced cost or a function of the reduced cost (reduced cost/investment cost) used to measure closeness.

A technology in a market group is allocated a share only if $abs(B_i) \leq CE_i$ (CE = close enough value). We make sure that no B equals zero! If any is equal to zero, a warning message is written out for the modeler that the variable has been ignored in the market share reallocation, and the share allocation process continues.

[MKT_CE(mkt_id), default = 0.2.]

"Tech. Preference" It may be desirable to allow a weighted preference for a technology, ∀. This value will be applied to each technology as part of determining the market share to be credited to each technology in a group.

Why is a market preference weight needed? For certain markets, consumers make their decisions based on more than just least cost – some of which are cultural/regional in nature. For example, in the U.S., vehicle efficiency and cost are often subordinate to performance. Such regional biases can be reflected by choosing appropriate weights.

[MKT_PREF(mkt_id,tch), default = 1.]

"Gamma" An indication of the degree of optimization to be applied when determining the shares for qualifying candidates in a market share group, γ. The larger the gamma (γ), the more the shares are based on cost only. For end-use markets, the gamma should be less than 0.3. For industrial and electricity generation choices, gamma should reflect consumers that are more optimizing. A gamma of about 2 is more appropriate for refineries and electric utility choices.

[MKT_GAMA(mkt_id), default = 2.]

"Reallocation %" Once the size of the entire market for market share group has been determined, the reallocation algorithm apportions this fraction of the total market for re-distribution to the competing "non-winners" who were "close enough" in the initial solve in each period.

[MKT_REAL(mkt_id), default = 0.2.]

6.3.3.3 Market Share Algorithm[110]

In this section the sequential actions taken by the code, and the implications, are elaborated to provide the modeler with an understanding of the way the algorithm is applied.

"Market Candidates" Out of the list of market candidates originally provided by the modeler (MKT_GRP), a check is done to ensure that a technology has not been explicitly bounded to 0; if it is bounded to 0, it is unavailable to the model in the current period. Those not bounded to zero constitute the viable market candidates.

[110] Each of the steps outlined in the algorithm are reflected in the MKTSHR.LOG file, in accordance with the level of details requested by the $MKTSHRLOG switch.

"Initial Solve" With a small lower bound set for each of the market candidates, the model is solved so as to obtain a reduced cost for every non-basic candidate.

"Market Qualifiers" A check is done to ensure that the "close enough" criteria are met, that is that the reduced cost of the candidates is below the criteria (and are not basic or upper bounded). These candidates then qualify for the reallocation algorithm. Those whose reduced costs are higher than the "close enough" criteria are deemed uncompetitive and thus not included in the group of qualified candidates.

"Size of Reallocation Group" Based upon all the candidates in a market group, the market size is determined according to the level of the variables associated with each technology in the group for the period. A proportion of this market, according to the reallocation % (MKT_REAL), is then calculated for redistribution to the qualifying market candidates. This pool then corresponds to the market size, MKT_SZ.

"Market Share" Let B be the duals (reduced cost) of the technologies of interest. Let K be all candidates in the group, J be the set of technologies that are "close enough" to get a share of the market set aside. Let η be the modeler provided technology preference weight. Then

$$\text{Market Share}_j = \eta * \text{abs}(B_j)^{-\gamma} / \{ \Sigma(\forall * \text{abs}(B_k)^{-\gamma}) \}$$
$$\text{for } k \in K, j \in J, \gamma \subset (.1, 5)$$

where the larger the η, the more optimizing the choices. This is tracked as MKT_SHR(mkt_id,tch).

"Reallocation Value" Each qualifying market candidate, former "non-winner" meeting the "close enough" criteria, is then assigned a potential market share value:

Market penetration level for technology = MKT_SZ(mkt_id) * MKT_SHR(mkt_id,tch). This target level for all qualifying candidates is named MKT_VAL(tch).

"Integrity Test" If within a market group any proposed market value (MKT_VAL) exceeds the level obtained by any basic or upper bounded variable then a second pass at assigning market shares is done. In this case all such basic or upper bounded variables having lower shares then

	the non-winner share are added to the market reallocation pool, and the market size (MKT_SZ) is increased by their level. Each of these former "winners" is then assigned a "dummy" reduced cost slightly smaller than the smallest reduced cost obtained by the "non-winners." This retains the relative attractiveness of the former "winners" but re-distributes the pool between all these closely competing technologies. The former "winners" are assigned upper bounds according to the share of the market suggested by the algorithm.
"Assignment of Lower Bound"	Each of the original market qualifying candidates are assigned lower bounds at their target market level to ensure that they at least obtain the desired level in the solution.
"Resolving with Market Shares"	With the market share criteria in place, the model is resolved for the current period. The final values obtained are then written to the LOG.

As a result of the Market Share algorithm, competing technologies are assured a share of the market based upon their relative attractiveness, and any preference imposed by the modeler.

6.3.4 Endogenous Technology Learning

The inter-period endogenous technology learning feature of SAGE enables the modeler to identify new technologies for which a determination is to be made as to their future investment cost based upon the total cumulative build up of the technology from the beginning of the modeling horizon up until the current period being solved. Below each of the input parameters is explained in more detail than was provided earlier in the tables, then the algorithm is elaborated.

6.3.4.1 Input parameters needed[111]

The input parameters are presented here in "logical" order based upon the order in which the modeler would tend to initiate, control and specify the Endogenous Technology Learning algorithm and its components.

"SETL Activation"	To request inter-period learning the model must be run time-stepped, activitate by means of the $SET SAGE 'YES' switch, and then the SETL switch activated and the appropriate data provided. [$SET SETL 'YES'/'NO', default 'NO']
"Identification of	The subset of all technologies that are to be subject to the

[111] The input switch, set or parameter name, and defaults are mentioned in [] at the end of each component.

Learning Technologies"	endogenous learning algorithm must be explicitly identified by the modeler. [TEG(tch)]
"Initial Cost"	The initial commercial cost associated with each learning technology. This initial investment cost will be subjected to the learning algorithm, and reduced according to the progress ratio specified by the modeler and the cumulative deployment level of the technology. [SC0(tch)]
"Progress Ratio"	The cost associated with learning technologies is a function of the cumulative deployment of the technology subject to the progress ratio. The progress ratio indicates how much this investment cost is to be lowered based upon each doubling of installed capacity. Note that another expression of this ratio is called the "learning rate", and is defined as 1 – PRAT. [PRAT(tch)]
"Initial Capacity Threshold"	Learning with respect to actually reducing the cost of a new technology is assumed to begin only once the total cumulative capacity exceeds some minimal level as expressed by this parameter. Below this level the "initial cost" (SC0) continues to apply. [CCAP0(tch)]
"Learning Spill"	Between technologies, and regions, learning based upon progress with components associated with one technology may have beneficial "spill" learning effects on another technology. To allow for such shared learning (of components or across technologies or regions) the modeler can interrelate such technologies by means of this parameter. Note that the technology "spilling" does not need to be a learning technology itself, and that the spill rate may change over time. [LSPILL(tch,teg,tp).LSPILL_R(reg1,tch,reg2,teg,tp)]

6.3.4.2 Endogenous Technology Learning Algorithm

The SAGE endogenous technology learning algorithm employs a standard representation of learning progress whereby the cost of a technology is assumed to be reduced by a certain amount for every doubling of the capacity. The progress ratio (PRAT) is the key parameter that controls this relationship.

Between each period, after all market share adjustments have been addressed, the solution for the current period is examined and the investment cost associated with the learning technologies is calculated and prepared for the next period solve as described in the following paragraphs.

Let $j \in J$ be the learning technologies, SC0j be the initial investment cost associated with each learning technology until such time as the cumulative capacity reaches the learning threshold (CCAP0[112]), and CCj be the total cumulative capacity installed to date (including any deployed prior to the modeling horizon (as expressed via RESID). Then the basic estimate of the cost of investing in technologies benefiting from learning is determined by:

$$SC_i = SC0_i * CC_j^{-(\ln(PRATj)/\ln(2))}$$

In this form, we accommodate international and cross-technology learning by means of a “spill” parameter (LSPILL_R(r’,t,r,j,year)), which is the fraction of the capacity of technology (t) in one region (r’) that contributes to learning in the current region (r) of the learning technology (j). Note that the code sets LSPILL_R(r,j,r,j) = 1 (that is the learning technology learns from itself).

$$SC_{r,i} = SC0_{r,i} * [CC_{r,j} + SUM((r',t), lspill\_r_{r',t,r,j} * CC_{r',t})]^{-(\ln(PRATj)/\ln(2))}$$

Note that in both of the examples above the period index has been left off, but LSPILL_R may be expressed by period and thus could change over time.

So SC_i is the specific cost owing to the cumulative capacity buildup and learning rate which then corresponds to the new investment cost (INVCOST) to be used when calculating the actual annualized investment cost (COST_INV) for each learning technology in each region. Note that for conversion technologies (con) the investment transmission and distribution investment costs (ETRANINV and EDISTINV) are added to the total cost.

[112] It is well know that the algorithm employed does not work if CC is < 1, so CCAP0 should always be larger than 1.

Thus although there is a “pot” for all capacity to go into, the learning that takes place in a particular region is a function of the capacity built in that region plus some weighted sum of the learning in other regions.

At the current time the initial values for the INVCOST and the SETL parameters are dumped before the solve loop begins in the list (LST) file. During each iteration, SETL_CC and SETL_SC are output to the LST file. [So searching for SETL_CC is the way to see what is going on SETL wise.]