Revue d'Economie et de Statistique Appliquée
Volume 19 number 2, July 2022
ISSN: 1112-234X EISSN: 2600-6642
Pages:6-20

THE ALGEBRAIC FORMULATION OF THE OPEN-SOURCE ENERGY MODELING SYSTEM (OSeMOSYS)

[*]Souhila BENBRAHIM
Ecole Nationale Supérieure de Statistique et d'Economie Appliquée ENSSEA
benbrahim.souhila@enssea.net

Saloua CHAOUCHE-QOUIDRAT
Ecole Nationale Supérieure de Statistique et d'Economie Appliquée ENSSEA
chaouche.saloua@hotmail.com

Rachid TOUMACHE
Ecole Nationale Supérieure de Statistique et d'Economie Appliquée ENSSEA Email :
rtoumache@gmail.com

Reçu le: 2021/10/22 **Accepté le :** 2022/03/12 **Publication en ligne le**: 2022/06/15

ABSTRACT: This paper discusses the mathematical analogy of the Open-Source Energy Modeling System (OSeMOSYS). It describes the algebraic formulation of the model by providing a plain description of sets, parameters and variables used in each block. As statisticians and economists, we are interested in mathematical development more than code implementation, so the added value of this paper is that it treats the detailed algebraic formulation which is developed from its latest full source code., we provide each block with a plain description of sets, parameters and variables used in each block of the OSeMOSYS with the detailed equations by making the passage from conditions used in the source code to simple solutions in order to have simple equations.
Keywords : OSeMOSYS; Energy modeling system; Bottom-up; Algebraic formulation; Open source

JEL Classification: C020; C610; Q40

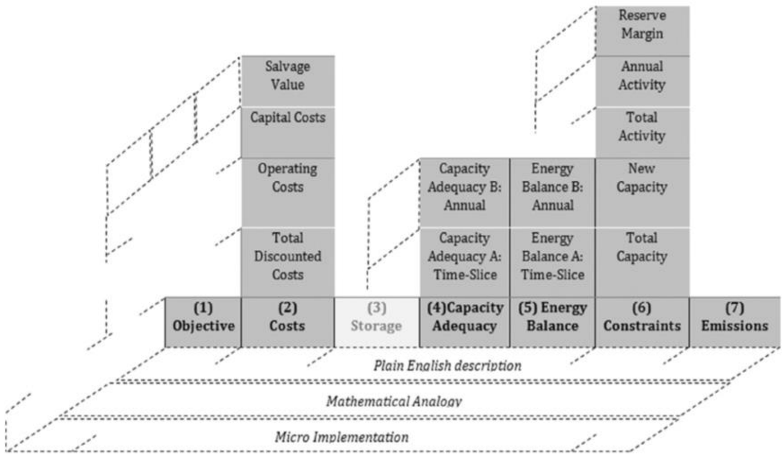
1. INTRODUCTION:

OSeMOSYS (open-source modelling system) is a long-run integrated assessment and energy planning. It has been used to create models of energy systems. It can focus on detailed power representations, or multi-resource (material, financial, all energy) systems. "OSeMOSYS is designed to be easily updated and modified to suit the needs of a particular analysis. To provide this capability, the model is being developed in a series of component 'blocks' of functionality"(Howells et al., 2011). It is a bottom-up model based on a linear programming with an objective function (block1) at the contrary to the rest of blocks: Costs, Storage, Capacity adequacy, Energy balance, Constraints and Emissions. Each block is also subdivided into several degrees of abstraction, as follows:

[*] Corresponding Author

- Plain English description
- Mathematical Analogy
- Micro Implementation

Figure N°1: Current OSeMOSYS 'blocks' and levels of abstraction.



Source : (Howells et al., 2011)

In our paper we will treat the detailed algebraic formulation which is developed from its latest full source code. The paper begins with an introduction and a literature review of modeling energy systems and presenting the OSeMOSYS as a bottom-up modeling energy system. Also, we provide each block with a plain description of sets, parameters and variables used in each block of the OSeMOSYS with the detailed equations by making the passage from conditions used in the source code to simple solutions in order to have simple equations for each block. Using the latest documentation of OSeMOSYS (*OSeMOSYS Documentation Release 0.0.1 KTH-DESA*, 2021) and from the code source published inside we have developed the algebraic formulation of the equations of each block to help statisticians and economists to get a better comprehension of the OSeMOSYS model.

2. LITERATURE REVIEW:

Energy system models can be considered as a subcategory of partial equilibrium models used to assess costs of reducing emissions; no need to represent impact by focusing on emissions rather than climate change (Doukas et al., 2019). OSeMOSYS as one of Energy system models based on the linear programming to minimize the total discounted cost was developed for the first time by Howells in its reference article (Howells et al., 2011) discussing its ethos, structure and development. OSeMOSYS expands in the second paper of Howells with Welsch by describing how 'blocks of functionality' may be added to represent variability in electricity generation, a prioritisation of demand types, shifting demand, and storage options. The paper demonstrates the flexibility and ease-of-use of OSeMOSYS with regard to modifications of its code (Welsch et al., 2012).

Many of studies are done using OSeMOSYS which are published in its official website containing also the data used in each case. We also have the github of the OSeMOSYS. In the appendix of (Howells et al., 2011) we have the first algebraic formulation but it is not refreshed so in this paper we will do a reverse method to obtain the latest algebraic reformulation from equations of the code source published in the github and in (*OSeMOSYS Documentation Release 0.0.1 KTH-DESA*, 2021).

3. Mathematical analogy of OSeMOSYS :

3.1. Objectif function

This particular equation represents the global objective of the model. The main objective of OSeMOSYS is to minimize the total system cost over the entire model period. This can be achieved through using Sets and Variables, for instance, in Sets we have Year (y) and Region (r). And in Variables we have positive costing variables which is represented in TotalDiscountedCost[r,y].

3.1.1. $\textbf{MinimizeCost} = \sum_{r,y} \text{TotalDiscountedCost}[r, y]$

3.2. Rate of demand

The equation provided below is utilized to derive the term RateOfDemand. From the user-provided data for SpecifiedAnnualDemand and SpecifiedDemandProfile. For each combination of commodity, TimeSlice and Year, the RateOfDemand is defined. Through the use of Sets, Parameters and Variables. In Sets we have Year (y), Region (r), TimeSlice (I) and Fuel (f). Moreover, in Parameters we have Global parameters in which YearSplit[l,y] must be strictly positive, and within parameters we have Demands as well as SpecifiedAnnualDemand[r,f,y] and SpecifiedDemandProfile[r,f,l,y].

3.2.1. $\textbf{RateofDemand}[\mathbf{r}, \mathbf{l}, \mathbf{f}, \mathbf{y}] = \text{SpecifiedAnnuualDemand}[r, f, y] \times \text{SpecifedDemandProfil}[r, f, l, y] \,/\, \text{YearSplit}[l, y]$

3.3. Capacity Adequacy A (CAa)

Used to compute total capacity of each technology for each year based on residual capacity from before the model period (ResidualCapacity), AccumulatedNewCapacity during the modelling period, and NewCapacity built in each year. The Capacity is next checked to see if it meets the RateOfTotalActivity in each TimeSlice and Year. There is also a constraint dependent on the size, or capacity, of each Technology unit (CapacityOfOneTechnologyUnit).

This sets down that the capacity of a particular Technology can only be a multiple of the user-defined CapacityOfOneTechnologyUnit. It is only provided through using **Sets** such as Year (y), Region (r), TimeSlice (I), Technology (t) and Mode_Of_Operation (m). Also through using **Parameters**, which include **Performance** and **Capacity constraints**, first, we have **Performance** such as OperationalLife[r,t] ResidualCapacity[r,t,y], CapacityToActivityUnit[r,t] and CapacityFactor[r,t,l,y], second, in **Capacity contsraints** we use CapacityOfOneTechnologyUnit[r,t,y] which must not be equal to zero. Furthermore, in **Variables** we have **Acitivity variables** and **Capacity variables** which are positive. **Activity variables** which include RateOfActivity[r,l,t,m,y] and RateOfTotalActivity[r,t,l,y]. And in **Capacity variables** we have NewCapacity[r,t,y],TotalCapacityAnnual[r,t,y],and NumberOfNewTechnologyUnits[r,t,y].

3.3.1. CAa1_Total NewCapacity

$$\text{AccumulatedNewCapacity}[r, t, y] = \sum_{yy > y - \text{OperationalLifeStorage}[r,s]}^{y} New\ Capacity[r, t, yy]$$

3.3.2. CAa2_Total AnnualCapacity

$$\text{TotalCapacityAnnual}[r, t, y] = \text{AccumulatedNewCapacity}[r, t, y] + \text{ResidualCapacity}[r, t, y]$$

3.3.3. CAa3_Total ActivityofEachTechnology

$$\text{RateOfTotalActivity}\,[\mathrm{r,t,l,y}] = \sum_{m} \mathit{RateOfActivity}\,[\mathrm{r,l,t,m,y}]$$

3.3.4. CAa4_Constraint-Capacity

$$\text{RateOfTotalActivity}\,[\mathrm{r,t,l,y}] \leq \mathit{TotalCapacityAnnual}\,[\mathrm{r,t,y}] \times \mathit{CapacityFactor}\,[\mathrm{r,t,l,y}] \times \mathit{CapacityToActivityUnit}\,[\mathrm{r,t}]$$

3.3.5. CAa5_TotalNewCapacity

$$\text{NewCapacity}\,[\mathrm{r,t,y}] = \text{CapacityOfOneTechnologyUnit}\,[\mathrm{r,t,y}] \times \text{NumberOfNewTechnology}\,[\mathrm{r,t,y}]$$

3.4. Capacity Adequacy B (CAb)

Guarantees that appropriate capacity of technologies exist to meet at least the average yearly demand. Through using **Sets** as well as **Parameters**. In **Sets** we use Year (y), Region (r), TimeSlice (I) and Technology (t). In **Parameters** such as **Global parameters** which include YearSplit[l,y]. And in **Performance** we use CapacityToActivityUnit[r,t], CapacityFactor[r,t,l,y] and AvailabilityFactor[r,t,y]. In **Variables** we have **Activity variables** and **Capacity variables** which are positive. **Activity variables** such as RateOfTotalActivity[r,t,l,y], and **Capacity variables** like TotalCapacityAnnual[r,t,y].

3.4.1. Cab1_PlannedMaintenance

$$\sum_{l} (\mathit{RateOfTotalActivity}\,[\mathrm{r,t,l,y}] \times \mathit{YearSplit}\,[l,y]) \leq \sum_{l} (\mathit{TotalCapacityAnnual}\,[\mathrm{r,t,y}] \times \mathit{CapacityFactor}\,[\mathrm{r,t,l,y}] \times \mathit{YearSplit}\,[l,y]) \times \mathit{AvailabilityFactor}\,[\mathrm{r,t,y}] \times \mathit{CapacityToActivityUnit}\,[\mathrm{r,t}]$$

3.5. Energy Balance A (EBa)

Guarantees that each commodity's demand is met in each TimeSlice. This can be achieved by using **Sets** such as Year (y), Region (r), TimeSlice (l), Fuel (f), Technology (t) and Mode_Of_Operation (m). Also by the use of **Parameters** which include **Global Parameters** and **Performance.** In Global **Parameters** we use YearSplit[l,y] , TradeRoute[r,rr,f,y] And in **Performance** in which must not be equal to a zero, we have OutputActivityRatio[r,t,f,m,y] and InputActivityRatio[r,t,f,m,y]. Moreover, we use **Variables** which are positive including **Activity Variables** like RateOfProductionByTechnologyByMode[r,l,t,m,f,y], RateOfActivity[r,l,t,m,y], RateOfProductionByTechnology[r,l,t,f,y], RateOfProduction[r,l,f,y], RateOfUseByTechnologyByMode[r,l,t,m,f,y], RateOfUseByTechnology[r,l,t,f,y], RateOfUse[r,l,f,y], Use[r,l,f,y], Trade[r,rr,l,f,y] and Production[r,l,f,y]. Also, we have **Demands** like RateOfDemand[r,l,f,y] and Demand[r,l,f,y].

3.5.1. EBa1_RateOfFuelProduction1

$$\text{RateOfProductionByTechnologyByMode}\,[\mathrm{r,l,t,m,f,y}] = \text{RateOfActivity}\,[\mathrm{r,l,t,m,f,y}] \times \text{OutputActivityRatio}\,[\mathrm{r,l,t,m,f,y}]$$

3.5.2. EBa2_RateOfFuelProduction2

$$\text{RateOfProductionByTechnology}\,[\mathrm{r,l,t,f,y}] = \sum_{\mathrm{m}} \text{RateOfProductionByTechnologyByMode}\,[\mathrm{r,l,t,m,f,y}]$$

3.5.3. EBa3_RateOfFuelProduction3

$$\text{RateOfProduction}\,[\mathrm{r,f,l,y}] = \sum_{\mathrm{t}} \text{RateOfProductionByTechnology}\,[\mathrm{r,l,t,f,y}]$$

3.5.4. EBa4_RateOfFuelUse1

$$\text{RateOfUseByTechnologyByMode}[r,l,t,m,f,y] = \text{RateOfActivity}[r,l,t,m,y] \times \text{InputActivityRatio}[r,t,m,f,y]$$

3.5.5. EBa5_RateOfFuelUse2

$$\text{RateOfUseByTechnology}[r,l,t,f,y] = \sum_{m} \text{RateOfUseByTechnologyByMode}[r,l,t,m,f,y]$$

3.5.6. EBa6_RateOfFuelUse3

$$\text{RateOfUse}[r,l,f,y] = \sum_{t} \text{RateOfUseByTechnology}[r,l,t,f,y]$$

3.5.7. EBa7_EnergyBalanceEachTS1

$$\text{Production}[r,l,f,y] = \text{RateOfProduction}[r,l,f,y] \times \text{YearSplit}[l,y]$$

3.5.8. EBa8_EnergyBalanceEachTS2

$$\text{Use}[r,l,f,y] = \text{RateOfUse}[r,l,f,y] \times \text{YearSplit}[l,y]$$

3.5.9. EBa9_EnergyBalanceEachTS3

$$\text{Demand}[r,l,f,y] = \text{RateOfDemand}[r,l,f,y] \times \text{YearSplit}[l,y]$$

3.5.10. EBa10_EnergyBalanceEachTS4

$$\text{Trade}[r,rr,l,f,y] = -\text{Trade}[rr,r,l,f,y]$$

3.5.11. EBa11_EnergyBalanceEachTS5

$$\text{Production}[r,l,f,y] \geq \text{Demand}[r,l,f,y] + \text{Use}[r,l,f,y] + \sum_{rr} \text{Trade}[r,rr,l,f,y] \times \text{TradeRate}[r,rr,l,f,y]$$

3.6. Energy Balance B (EBb)

Guarantees that each commodity's demand is met each year. This can be achieved through using **Sets**, **Parameters** and **Variables**. In **Sets** we have Year (y), Region (r), TimeSlice (I) and Fuel (f). Moving on, in **Parameters** we have **Global Parameters** such as TradeRoute[r,rr,f,y] and in **Demands** we use AccumulatedAnnualDemand[r,f,y]. Finally in **Variables** we have **Activity variables** which are positive, such as, Use[r,l,f,y], Trade[r,rr,l,f,y], Production[r,l,f,y], TradeAnnual[r,rrf,y], ProductionAnnual[r,f,y] and UseAnnual[r,f,y].

3.6.1. EBb1_EnergyBalanceEachYear1

$$\text{ProductionAnnual}[r,f,y] = \sum_{l} \text{Production}[r,l,f,y]$$

3.6.2. EBb2_EnergyBalanceEachYear2

$$\text{UseAnnual}[r,f,y] = \sum_{l} \text{Use}[r,l,f,y]$$

3.6.3. EBb3_EnergyBalanceEachYear3

$$\text{TradeAnnual}[r,rr,f,y] = \sum_{l} \text{Trade}[r,rr,l,f,y]$$

3.6.4. EBb4_EnergyBalanceEachYear4

$$\text{ProductionAnnual}[r,f,y] \geq \text{UseAnnual}[r,f,y] + \sum_{rr} (\text{TradeAnnual}[r,l,f,y] \times \text{TradeRoute}[r,rr,l,f,y]) + \text{AccumulatedAnnualDemand}[r,f,y]$$

3.7. Accounting Technology Production Use (Acc)

ProductionByTechnology, UseBytechnology, TotalAnnualTechnologyActivityByMode, and ModelPeriodCostByRegion are accounting equations used to generate specific intermediate variables. This can only be achieved through using **Sets** such as YEAR (y), REGION (r), TIMESLICE (I), FUEL(f), TECHNOLOGY (t) and MODE_OF_OPERATION (m). Also through the use of **Parameters** we have **Global**

parameters such as YearSplit[l,y]. Finally Through the use of **Variables** in which we have **Activity variables** that are positive including RateOfActivity[r,l,t,m,y], RateOfProductionByTechnology[r,l,t,f,y], RateOfUseByTechnology[r,l,t,f,y], ProductionByTechnology[r,l,t,f,y], UseByTechnology[r,l,t,f,y] and TotalAnnualTechnologyActivityByMode[r,t,m,y]. As well as using **Costing variables** such as ModelPeriodCostByRegion[r] and TotalDiscountCosts[r,y].

3.7.1. Acc1_FuelProductionByTechnology

$$\text{ProductionByTechnology}[r,l,t,f,y] = \text{RateOfProductionByTechnology}[r,l,t,f,y] \times \text{YearSplit}[l,y]$$

3.7.2. Acc2_FuelUseByTechnology

$$\text{UseByTechnology}[r,l,t,f,y] = \text{RateOfUseByTechnology}[r,l,t,f,y] \times \text{YearSplit}[l,y]$$

3.7.3. Acc3_AverageAnnualRateOfActivity

$$\text{TotalAnnualTechnologyActivityByMode}[r,t,m,y] = \sum_{l} (\text{RateOfActivity}[r,l,t,m,y] \times \text{YearSplit}[l,y])$$

3.7.4. Acc4_ModelPeriodCostByRegion

$$\text{ModelPeriodCostByRegion}[r] = \sum_{y} \text{TotalDiscountCosts}[r,y]$$

3.8. Storage Equations (S)

This can only be achieved through using **Sets** Year (y), Region (r), TimeSlice (I), Mode_Of_Operation (m), Season (ls), DayType (ld), DailyTimeBracket (lh) and Storage (s). Morevover, we use **Parameters**, first we have **Global parameters** which includes YearSplit[l,y], DaySplit[lh,y], Conversionls[l,ls], Conversionld[ld,l], Conversionlh[lh,l] and DaysInDayType[ls,ld,y]. Second, we have **Storage** in which they are strictly positive, we use TechnologyToStorage[r,t,s,m], TechnologyFromStorage[r,t,s,m], and StorageLevelStart[r,s]. Finally in **Variables** we use **Activity variables** which are positive like RateOfActivity[r,l,t,m,y] and in **Storage Variables** we have RateOfStorageCharge[r,s,ls,ld,lh,y], RateOfStorageDischarge[r,s,ls,ld,lh,y], NetChargeWithinYear[r,s,ls,ld,lh,y] and NetChargeWithinDay[r,s,ls,ld,lh,y]. And other storage variables which are positive like StorageLevelYearStart[r,s,y], StorageLevelSeasonStart[r,s,ls,y], StorageLevelDayTypeStart[r,s,ls,ld,y], StorageLevelYearFinish[r,s,y] and StorageLevelDayTypeFinish[r,s,ls,ld,y].

3.8.1. S1_RateOfStorageCharge

$$\text{RateOfStorageCharge}[r,s,ls,ld,lh,y] = \sum_{t,m,l} \text{RateOfActivity}[r,l,t,m,y] \times \text{TechnologyToStorage}[r,t,s,m] \times \text{Conversionls}[l,ls] \times \text{Conversionld}[l,ld] \times \text{Conversionlh}[l,lh]$$

3.8.2. S2_RateOfStorageDischarge

$$\text{RateOfStorageDischarge}[r,s,ls,ld,lh,y] = \sum_{t,m,l} (\text{RateOfActivity}[r,l,t,m,y] \times \text{TechnologyFromStorage}[r,t,s,m] \times \text{Conversionls}[l,ls] \times \text{Conversionld}[l,ld] \times \text{Conversionlh}[l,lh])$$

3.8.3. S3_NetChargeWithinYear

$$NetChargeWithinYear[r,s,ls,ld,lh,y] = \sum_{l} (RateOfStorageCharge[\mathrm{r,s,ls,ld,lh,y}] - RateOfStorageDischarge[\mathrm{r,s,ls,ld,lh,y}]) \times YearSplit[l,y] \times Conversionls[l,ls] \times Conversionld[l,ld] \times Conversionlh[l,lh]$$

3.8.4. S4_NetChargeWithinDay

$$NetChargeWithinDay[r,s,ls,ld,lh,y] = (RateOfStorageCharge[\mathrm{r,s,ls,ld,lh,y}] - RateOfStorageDischarge[\mathrm{r,s,ls,ld,lh,y}]) \times DaySplit[\mathrm{lh,y}]$$

3.8.5. S5_and_S6_StorageLevelYearStart

$$\text{If } \mathrm{y} = \min(\mathrm{yy}) : \mathrm{StorageLevelStart}[\mathrm{r,s}]$$
$$\text{else}: \mathrm{StorageLevelYearStart}[\mathrm{r,s,y}-1] + \sum \mathrm{NetChargeWithinYear}[\mathrm{r,s,ls,ld,lh,y}-1]$$

3.8.6. S7_and_S8_StorageLevelYearFinish

$$\text{If } \mathrm{y} < \max(\mathrm{yy}) \ \mathrm{StorageLevelYearStart}[\mathrm{r,s,y}+1]$$
$$\text{else}: \mathrm{StorageLevelYearStart}[\mathrm{r,s,y}] + \sum \mathrm{NetChargeWithinyear}[\mathrm{r,s,ls,ld,lh,y}]$$

3.8.7. S9_and_S10_StorageLevelSeasonStart

$$\text{If } \mathrm{Ls} = \min(\mathrm{LsLs}) : \mathrm{StorageLevelYearStart}[\mathrm{r,s,y}]$$
$$\text{else}: \mathrm{StorageLevelSeasonStart}[\mathrm{r,s,ls}-1,\mathrm{y}] + \sum \mathrm{NetChargeWithinYear}[\mathrm{r,s,ls}-1,\mathrm{ld,lh,y}]$$

3.8.8. S11_and_S12_StorageLevelDayTypeStart

$$\text{If } \mathrm{Ld} = \min(\mathrm{LdLd}) : \mathrm{StorageLevelSeasonStart}[\mathrm{r,s,ls,y}]$$
$$\text{else}: \mathrm{StorageLevelDayTypeStart}[\mathrm{r,s,ls,ld}-1,\mathrm{y}] + \sum \mathrm{NetChargeWithinDay}[\mathrm{r,s,ls,ld}-1,\mathrm{lh,y}] \times \mathrm{DaysInDayType}[\mathrm{ls,ld}-1,\mathrm{y}]$$

3.8.9. S13_and_S14_ and_S15_StorageLevelDayTypeFinish

$$\text{If } \mathrm{Ls} = \max(\mathrm{LsLs}) \ \&\& \ \mathrm{Ld} = \max(\mathrm{LdLd}): \mathrm{StorageLevelYearFinish}[\mathrm{r,s,y}]$$
$$\text{else If } \mathrm{Ld} = \max(\mathrm{LdLd}): \mathrm{StorageLevelSeasonStart}[\mathrm{r,s,ls}+1,\mathrm{y}]$$
$$\text{else}: \mathrm{StorageLevelDayTypeFinish}[\mathrm{r,s,ls,ld}+1,\mathrm{y}] - \sum \mathrm{NetChargeWithinDay}[\mathrm{r,s,ls,ld}+1,\mathrm{lh,y}] \times \mathrm{DaysInDayType}[\mathrm{ls,ld}+1,\mathrm{y}]$$

3.9. Storage Constraints (SC)

This can only be achieved through using **Sets** such as Year (y), Region (r), TimeSlice (I), Season (ls), DayType (ld), DailyTimeBracket (lh) and Storage (s). As well as using **Parameters** which include **Storage** such as StorageMaxChargeRate[r,s] and StorageMaxDischargeRate[r,s]. Finally by the use of **Variables** which are positive including **Storage variables** like NetChargeWithinDay[r,s,ls,ld,lh,y], RateOfStorageCharge[r,s,ls,ld,lh,y], RateOfStorageDischarge[r,s,ls,ld,lh,y], StorageLevelDayTypeStart[r,s,ls,ld,y], StorageLevelDayTypeFinish[r,s,ls,ld,y], StorageLowerLimit[r,s,y] and StorageUpperLimit[r,s,y].

3.9.1. SC1_LowerLimit_BeginningOfDailyTimeBracketOfFirstInstanceOfDayTypeInFirstWeekConstraint

$$\mathrm{StorageLevelDayTypeStart}[\mathrm{r,s,ls,ld,y}]) + \sum_{\mathrm{LhLh}}^{\mathrm{Lh}} (\mathrm{NetChargeWithinDay}[\mathrm{r,s,ls,ld,lhlh,y}]) - \mathrm{StorageLowerLimit}[\mathrm{r,s,y}] \geq 0$$

3.9.2. SC1_UpperLimit_BeginningOfDailyTimeBracketOfFirstInstanceOfDayTypeInFirstWeekConstraint

$$\mathrm{StorageLevelDayTypeStart}[\mathrm{r,s,ls,ld,y}]) + \sum_{\mathrm{LhLh}}^{\mathrm{Lh}} (\mathrm{NetChargeWithinDay}[\mathrm{r,s,ls,ld,lhlh,y}]) - \mathrm{StorageUpperLimit}[\mathrm{r,s,y}] \leq 0$$

3.9.3. SC2_LowerLimit_EndOfDailyTimeBracketOfLastInstanceOfDayTypeInFirstWeekConstraint

$$\text{If Ld} > \min(\text{LdLd}) : \text{StorageLevelDayTypeStart}[r,s,ls,ld,y] - \sum_{\text{LhLh}}^{\text{Lh}} (\text{NetChargeWithinDay}[r,s,ls,ld-1,lhlh,y]) - \text{StorageLowerLimit}[r,s,y] \geq 0$$

3.9.4. SC2_UpperLimit_EndOfDailyTimeBracketOfLastInstanceOfDayTypeInFirstWeekConstraint

$$\text{If Ld} > \min(\text{LdLd}) : \text{StorageLevelDayTypeStart}[r,s,ls,ld,y] - \sum_{\text{LhLh}}^{\text{Lh}} (\text{NetChargeWithinDay}[r,s,ls,ld-1,lhlh,y]) - \text{StorageUpperLimit}[r,s,y] \leq 0$$

3.9.5. SC3_LowerLimit_EndOfDailyTimeBracketOfLastInstanceOfDayTypeInLastWeekConstraint

$$\text{StorageLevelDayTypeFinish}[r,s,ls,ld,y]) - \sum_{\text{LhLh}}^{\text{Lh}} (\text{NetChargeWithinDay}[r,s,ls,ld,lhlh,y]) - \text{StorageLowerLimit}[r,s,y] \geq 0$$

3.9.6. SC3_UpperLimit_EndOfDailyTimeBracketOfLastInstanceOfDayTypeInLastWeekConstraint

$$\text{StorageLevelDayTypeFinish}[r,s,ls,ld,y]) - \sum_{\text{LhLh}}^{\text{Lh}} (\text{NetChargeWithinDay}[r,s,ls,ld,lhlh,y]) - \text{StorageUpperLimit}[r,s,y] \leq 0$$

3.9.7. SC4_LowerLimit_BeginningOfDailyTimeBracketOfFirstInstanceOfDayTypeInLastWeekConstraint

$$\text{If Ld} > \min(\text{LdLd}) : \text{StorageLevelDayTypeFinish}[r,s,ls,ld-1,y] - \sum_{\text{LhLh}}^{\text{Lh}} (\text{NetChargeWithinDay}[r,s,ls,ld,lhlh,y]) - \text{StorageLowerLimit}[r,s,y] \geq 0$$

3.9.8. SC4_UpperLimit_BeginningOfDailyTimeBracketOfFirstInstanceOfDayTypeInLastWeekConstraint

$$\text{If Ld} > \min(\text{LdLd}) : \text{StorageLevelDayTypeFinish}[r,s,ls,ld-1,y] - \sum_{\text{LhLh}}^{\text{Lh}} (\text{NetChargeWithinDay}[r,s,ls,ld,lhlh,y]) - \text{StorageUpperLimit}[r,s,y] \leq 0$$

3.9.9. SC5_MaxChargeConstraint

$$\text{RateOfStorageCharge}[r,s,ls,ld,lh,y] \leq \text{StorageMaxChargeRate}[r,s]$$

3.9.10. SC6_MaxDischargeConstraint

$$\text{RateOfStorageDischarge}[r,s,ls,ld,lh,y] \leq \text{StorageMaxDichargeRate}[r,s]$$

3.10. Storage Investments (SI)

Calculating the total discounted capital costs spent in each year for each storage technology. This can only be achieved through the use of **Sets** Year (y), Region (r) and Storage (s). As well as using **Parameters** which include **Global parameters** such as DiscountRate[r], DepreciationMethod[r] And in **Storage** we have OperationalLifeStorage[r,s], ResidualStorageCapacity[r,s,y], MinStorageCharge[r,s,y] and CapitalCostStorage[r,s,y]. Finally by the use of **Variables** which are positive including **Storage variables** such as StorageLowerLimit[r,s,y], StorageUpperLimit[r,s,y], AccumulatedNewStorageCapacity[r,s,y], NewStorageCapacity[r,s,y], CapitalInvestmentStorage[r,s,y], DiscountedCapitalInvestmentStorage[r,s,y], SalvageValueStorage[r,s,y], DiscountedSalvageValueStorage[r,s,y], and TotalDiscountedStorageCost[r,s,y].

3.10.1. SI1_StorageUpperLimit

$$\text{StorageUpperLimit}[r,s,y] = \text{AccumulatedNewStorageCapacity}[r,s,y] + \text{ResidualStorageCapacity}[r,s,y]$$

3.10.2. SI2_StorageLowerLimit

$$\mathit{Storage}\text{Lower}\mathit{Limit}[r,s,y] = \text{MinStorageCharge}[r,s,y] \times \text{StorageUpperLimit}[r,s,y]$$

3.10.3. SI3_TotalNewStorage

$$\text{AccumulatedNewStorageCapacity}[r,s,y] = \sum_{yy>y-\text{OperationalLifeStorage}[r,s]}^{y} \text{NewStorageCapacity}[r,s,yy]$$

3.10.4. SI4_UndiscountedCapitalInvestmentStorage

$$\text{CapitalInvestmentStorage}[r,s,y] = \text{CapitalCostStorage}[r,s,y] \times \text{NewStorageCapacity}[r,s,y]$$

3.10.5. SI5_DiscountingCapitalInvestmentStorage

$$\text{DiscountedCapitalInvestmentStorage}[r,s,y] = \frac{\text{CapitalInvestmentStorage}[r,s,y]}{(1+\text{DiscountRate}[r])^{y-\min(yy)}}$$

3.10.6. SI6_SalvageValueStorageAtEndOfPeriod1

When $y + \text{OperationalLifeStorage}[r,s] - 1 \leq \max(yy)$: $\text{SalvageValueStorage}[r,s,y] = 0$

3.10.7. SI7_SalvageValueStorageAtEndOfPeriod2

$$\begin{gathered}\text{If}\\ \text{DepreciationMethod}[r] = 1\\ \&\&\ (y + \text{OperationalLifeStorage}[r,s] - 1) > \max(yy)\\ \&\&\ \text{DiscountRate}[r] = 0\\ \text{DepreciationMethod}[r] = 2\\ \&\&\ (y + \text{OperationalLifeStorage}[r,s] - 1) > \max(yy)\end{gathered}$$

$$: \text{SalvageValueStorage}[r,s,y] = \text{CapitalInvestmentStorage}[r,s,y] \times \left(\frac{1-(\max(yy)-y+1)}{\text{OperationalLifeStorage}[r,s]}\right)$$

3.10.8. SI8_SalvageValueStorageAtEndOfPeriod3

$$\begin{gathered}\text{If}\\ \text{DepreciationMethod}[r] = 1\\ \&\&\ (y + \text{OperationalLifeStorage}[r,s] - 1) > \max(yy)\\ \&\&\ \text{DiscountRate}[r] > 0\end{gathered}$$

$$: \text{SalvageValueStorage}[r,s,y] = \text{CapitalInvestmentStorage}[r,s,y] \times \left(1 - \left(\frac{(1+\text{DiscountRate}[r])^{\max(yy)-y+1} - 1}{(1+\text{DiscountRate}[r])^{\text{OperationalLifeStorage}[r,s]} - 1}\right)\right)$$

3.10.9. SI9_SalvageValueStorageDiscountedToStartYear

$$\text{DiscountedSalvageValueStorage}[r,s,y] = \frac{\text{SalvageValueStorage}[r,s,y]}{(1+\text{DiscountRate}[r])^{(\max(yy)-\min(yy)+1)}}$$

3.10.10. SI10_TotalDiscountedCostByStorage

$$\text{TotalDiscountedStorageCost}[r,s,y] = \text{DiscountedCapitalInvestmentStorage}[r,s,y] - \text{DiscountedSalvageValueStorage}[r,s,y]$$

3.11. Capital Costs (CC)

Calculating the total discounted capital cost spent in each technology for each year. This can be only achieved through the use of **Sets**, **Parameters** and **Variables**. In **Sets** we have Year (y), Region (r) and Technology (t). In addition to using **Parameters** we have

Global parameters like DiscountRate[r] and **Technology costs** such as CapitalCost[r,t,y]. Finally we use **Variables** which are positive including **Capacity variables** like NewCapacity[r,t,y] and **Costing variables** such as CapitalInvestment[r,t,y] and DiscountedCapitalInvestment[r,t,y]

3.11.1. CC1_UndiscountedCapitalInvestment

$$\text{CapitalInvestment}[r,t,y] = \text{CapitalCost}[r,t,y] \times \text{NewCapacity}[r,t,y]$$

3.11.2. CC2_DiscountingCapitalInvestment

$$\text{DiscountedCapitalInvestment}[r,t,y] = \frac{\text{CapitalInvestment}[r,t,y]}{(1+\text{DiscountRate}[r])^{(y-\min(yy))}}$$

3.12. Salvage Value (SV)

Calculates the portion of the initial capital cost that can be retrieved at the end of a technologies operational life. One of two depreciation methodologies, straight line or sinking fund, can be used to compute the salvage value. This operation can be accomplished through the use of **Sets** in which we have Year (y), Region (r) and Technology (t), and Storage (s). Additionally through the use of **Parameters** which involve **Global parameters** such as DiscountRate[r] which is strictly positive and DepreciationMethod[r], **Techonoly costs** like CapitalCost[r,t,y] and in **Performance** we have OperationalLifeStorage[r,s]. By ultimately using **Variables** which are positive we have **Capacity variables** like NewCapacity[r,t,y] and **Costing variables** such as SalvageValue[r,t,y] and DiscountedSalvageValue[r,t,y].

3.12.1. SV1_SalvageValueAtEndOfPeriod1

$$\text{If}$$
$$\text{DepreciationMethod}[r] = 1$$
$$\&\&\ (y + \text{OperationalLifeStorage}[r,s] - 1) > \max(yy)$$
$$\&\&\ \text{DiscountRate}[r] > 0$$
$$\text{SalvageValue}[r,t,y] = \text{CapitalCost}[r,t,y] \times \text{NewCapacity}[r,t,y] \times$$
$$(1 - (\frac{((1+\text{DiscountRate}[r])^{\max(yy)-y+1)} - 1)}{((1+\text{DiscountRate}[r])^{\text{OperationalLife}[r,t]} - 1)}$$

3.12.2. SV2_SalvageValueAtEndOfPeriod2

$$\text{If}$$
$$\text{DepreciationMethod}[r] = 1$$
$$\&\&\ (y + \text{OperationalLifeStorage}[r,s] - 1) > \max(yy)$$
$$\&\&\ \text{DiscountRate}[r] = 0$$
$$\text{DepreciationMethod}[r] = 2$$
$$\&\&\ (y + \text{OperationalLifeStorage}[r,s] - 1) > \max(yy)$$
$$\text{SalvageValue}[r,t,y] = \text{CapitalCost}[r,t,y] \times \text{NewCapacity}[r,t,y] \times (\frac{(\max(yy) - y + 1)}{\text{OperationalLife}[r,t]})$$

3.12.3. SV3_SalvageValueAtEndOfPeriod3

$$\text{If}$$
$$(y + \text{OperationalLifeStorage}[r,s] - 1) \leq \max(yy)$$
$$\text{SalvageValue}[r,t,y] = 0$$

3.12.4. SV4_SalvageValueDiscountedToStartYear

$$\text{DiscountedSalvageValue}[r,t,y] = \frac{\text{SalvageValue}[r,t,y]}{(1+\text{DiscountRate}[r])^{(1+\max(yy)-\min(yy))}}$$

3.13. Operating Costs (OC)

Calculates each technology's total variable and fixed operational expenses for each year. This operation can only be accomplished through the use of **Sets** which include Year (y), Region (r), Technology (t) and Mode_Of_Operation (m). Moreover we use **Parameters** in which we have **Global parameters** such as DiscountRate[r] which is strictly positive, and in **Technology costs** we have VariableCost[r,t,m,y] and FixedCost[r,t,y]. Eventually we use **Variables** which are positive including **Activity variables** such as

TotalAnnualTechnologyActivityByMode[r,t,m,y], **Capacity variables** we have TotalCapacityAnnual[r,t,y] and **Costing variables** we use AnnualVariableOperatingCost[r,t,y], AnnualFixedOperatingCost[r,t,y], OperatingCost[r,t,y] and DiscountedOperatingCost[r,t,y].

3.13.1. OC1_OperatingCostsVariable

$$\text{AnnualVariableOperatingCost}[r,t,y] = \sum_{m}(\text{TotalAnnualTechnologyActivityByMode}[r,t,m,y] \times \text{VariableCost}[r,t,m,y])$$

3.13.2. OC2_OperatingCostsFixedAnnual

$$\text{AnnualFixedOperatingCost}[r,t,y] = \text{TotalCapacityAnnual}[r,t,y] \times \text{FixedCost}[r,t,y]$$

3.13.3. OC3_OperatingCostsTotalAnnual

$$\text{OperatingCost}[r,t,y] = \text{AnnualFixedOperatingCost}[r,t,y] + \text{AnnualVariableOperatingCost}[r,t,y]$$

3.13.4. OC4_DiscountedOperatingCostsTotalAnnual

$$\text{DiscountedOperatingCost}[r,t,y] = \frac{\text{OperatingCost}[r,t,y]}{(1+\text{DiscountRate}[r])^{(y-\min(yy)+0.5)}}$$

3.14. Total Discounted Costs (TDC)

Calculating the total discounted system cost throughout the entire model period to give the TotalDiscountedCost. In the objective function of the model, this is the variable that is minimised. This operation is attained by using **Sets** such as Year (y), Region (r), Technology (t) and Storage (s). In this same operation we use **Variables** which are positive including **Storage variables** like TotalDiscountedStorageCost[r,s,y] and **Costing** including variablesDiscountedOperatingCost[r,t,y], DiscountedCapitalInvestment[r,t,y], TotalDiscountedCostByTechnology[r,t,y], DiscountedSalvageValue[r,t,y], TotalDiscountedCost[r,y] and in **Reserve Margin variables** we use DiscountedTechnologyEmissionsPenalty[r,t,y].

3.14.1. TDC1_TotalDiscountedCostByTechnology

$$\begin{aligned}\text{TotalDiscountedCostByTechnology}[r,t,y] &= \text{DiscountedOperatingCost}[r,t,y] + \text{DiscountedCapitalInvestment}[r,t,y] \\ &+ \text{DiscountedTechnologyEmissionsPenalty}[r,t,y] \\ &- \text{DiscountedSalvageValue}[r,t,y]\end{aligned}$$

3.14.2. TDC2_TotalDiscountedCost

$$\text{TotalDiscountedCost}[r,y] = \sum_{t}\text{TotalDiscountedCostByTechnology}[r,t,y] + \sum_{s}\text{TotalDiscountedStorageCost}[r,s,y]$$

3.15. Total Capacity Constraints (TCC)

Guarantees that each technology's total capacity is greater than or less than the user-defined parameters TotalAnnualMinCapacityInvestment and TotalAnnualMaxCapacityInvestment in each year. This operation can be accomplished through the use of **Sets** like Year (y) Region (r), and Technology (t). Also through the use of **Parameters** which are strictly positive including **Capacity constraints** such as TotalAnnualMaxCapacity[r,t,y] and TotalAnnualMinCapacity[r,t,y]. Lastly by the use of **Variables** which are positive such as **Capacity variables** like TotalCapacityAnnual[r,t,y].

3.15.1. TCC1_TotalAnnualMaxCapacityConstraint

$$\text{TotalCapacityAnnual}[r,t,y] \leq \text{TotalAnnualMaxCapacity}[r,t,y]$$

3.15.2. TCC2_TotalAnnualMinCapacityConstraint

$$\text{TotalCapacityAnnual}[r,t,y] \geq \text{TotalAnnualMinCapacity}[r,t,y]$$

3.16. New Capacity Constraints (NCC)

Guarantees that each year's new capacity for each technology deployed is larger than or less than the user-defined parameters TotalAnnualMinCapacityInvestment and TotalAnnualMaxCapacityInvestment, accordingly. This process can be only accomplished through the use of **Sets** such as Year (y), Region (r) , and Technology (t). In addition to the use of **Parameters** which are strictly positive including **Investment constraints** TotalAnnualMaxCapacityInvestment[r,t,y] and TotalAnnualMinCapacityInvestment[r,t,y]. Ulimately through the use of **Variables** which are positive including **Capacity variables** NewCapacity[r,t,y].

3.16.1. NCC1_TotalAnnualMaxNewCapacityConstraint

$$\text{NewCapacity}[r,t,y] \leq \text{TotalAnnualMaxCapacityInvestment}[r,t,y]$$

3.16.2. NCC2_TotalAnnualMinNewCapacityConstraint

$$\text{NewCapacity}[r,t,y] \geq \text{TotalAnnualMinCapacityInvestment}[r,t,y]$$

3.17. Annual Capacity Constraints (AAC)

Guarantees that each technology's total annual activity is greater than or less than the user-defined parameters TotalTechnologyAnnualActivityLowerLimit and TotalTechnologyAnnualActivityUpperLimit, respectively. This process can be attained through the use of **Sets** including Year (y), Region (r), TimeSlice (l) and Technology (t). As well as the use of **Parameters** which include **Global parameters** like YearSplit[l,y] and **Activity constraints** which are strictly positive such as TotalTechnologyAnnualActivityUpperLimit[r,t,y] and TotalTechnologyAnnualActivityLowerLimit[r,t,y]. Lastly through the use of **Variables** which are positive including **Activity variables** we use RateOfTotalActivity[r,t,l,y] and TotalTechnologyAnnualActivity[r,t,y].

3.17.1. AAC1_TotalAnnualTechnologyActivity

$$\text{TotalTechnologyAnnualActivity}[r,t,y] = \sum_{l} (\text{RateOfTotalActivity}[r,t,l,y] \times \text{YearSplit}[l,y])$$

3.17.2. AAC2_TotalAnnualTechnologyActivityUpperLimit

$$\text{TotalTechnologyAnnualActivity}[r,t,y] \leq \text{TotalTechnologyAnnualActivityUpperLimit}[r,t,y]$$

3.17.3. AAC3_TotalAnnualTechnologyActivityLowerLimit

$$\text{TotalTechnologyAnnualActivity}[r,t,y] \geq \text{TotalTechnologyAnnualActivityLowerLimit}[r,t,y]$$

3.18. Total Activity Constraints (TAC)

Guarantees that the total activity of each technology is greater than or less than the user-defined parameters for the entire model period, TotalTechnologyModelPeriodActivityLowerLimit and TotalTechnologyModelPeriodActivityUpper-Limit respectively. This process can be accomplished through the use of **Sets** such as Year (y), Region (r), and Technology (t). Additionally through the use of **Parameters,** which involve **Activity constraints which are strictly positive** like TotalTechnologyModelPeriodActivityUpperLimit[r,t,y] and TotalTechnologyModelPeriodActivityLowerLimit[r,t,y]. Ultimately through the use of **Variables** which include positive **Activity variables** such as TotalTechnologyModelPeriodActivity[r,t] and TotalTechnologyAnnualActivity[r,t,y].

3.18.1. TAC1_TotalModelHorizonTechnologyActivity

$$\text{TotalTechnologyModelPeriodActivity}[r,t] = \sum_{y} \text{TotalTechnologyAnnualActivity}[r,t,y]$$

3.18.2. TAC2_TotalModelHorizonTechnologyActivityUpperLimit

$$\text{TotalTechnologyModelPeriodActivity}[r,t] \leq \text{TotalTechnologyModelPeriodActivityUpperLimit}[r,t]$$

3.18.3. TAC3_TotalModelHorizenTechnologyActivityLowerLimit

$$\text{TotalTechnologyModelPeriodActivity}[r,t] \geq \text{TotalTechnologyModelPeriodActivityLowerLimit}[r,t]$$

3.19. Reserve Margin Constraints (RM)

Guarantees that the user-defined ReserveMargin is maintained by installing sufficient reserve capacity of particular technologies (ReserveMarginTagTechnology = 1). This process can be successful through the use of **Sets** like Year (y), Region (r), TimeSlice (l) and Fuel (f). As well as the use of **Parameters** in **terms** of **Performance** such as CapacityToActivityUnit[r,t] and **Reserve Margin** in which we use ReserveMarginTagTechnology[r,t,y], ReserveMarginTagFuel[r,f,y] and ReserveMargin[r,y]. Also through the use of **Variables** which are positive including **Activity variables** such as RateOfProduction[r,l,f,y], **Capacity variables** like TotalCapacityAnnual[r,t,y], and **Reserve Margin Variables** such as TotalCapacityInReserveMargin[r,y] and DemandNeedingReserveMargin[r,l,y].

3.19.1. RM1_ReserveMargin_TechnologiesIncluded_In_Activity_Units

$$\text{TotalCapacityInReserveMargin}[r,y] = \sum_{t}(\text{TotalCapacityAnnual}[r,t,y] \times \text{ReserveMarginTagTechnology}[r,t,y] \times \text{CapacityToActivityUnit}[r,t])$$

3.19.2. RM2_ReserveMargin_FuelsIncluded

$$\text{DemandNeedingReserveMargin}[r,l,y] = \sum_{f}(\text{RateOfProduction}[r,l,f,y] \times \text{ReserveMarginTagFuel}[r,f,y])$$

3.19.3. RM3_ReserveMargin_Constraint

$$\text{TotalCapacityInReserveMargin}[r,y] \geq \text{DemandNeedingReserveMargin}[r,l,y] \times \text{ReserveMargin}[r,y]$$

3.20. RE Production Target (RE)

Guarantees that production from renewable energy technologies (RETagTechnology = 1) meets the user-defined renewable energy (RE) target. This procedure can be achieved through the use of **Sets** which include Year (y), Region (r), TimeSlice (l), Fuel (f) and Technology (t). In addition, through the use **Parameters** in terms of **Global parameters** like YearSplit[l,y] and **RE Generation target** such as RETagTechnology[r,t,y], RETagFuel[r,f,y] and REMinProductionTarget[r,y]. Eventually through the use of **Variables** which are positive including **Activity variables** we have ProductionByTechnologyAnnual[r,t,f,y], ProductionByTechnology[r,l,t,f,y], RateOfProduction[r,l,f,y], RateOfUseByTechnology[r,l,t,f,y] and UseByTechnologyAnnual[r,t,f,y]. Finally we use **Reserve Margin variables** such as TotalREProductionAnnual[r,y] and RETotalProductionOfTargetFuelAnnual[r,y].

3.20.1. RE1_FuelProductionByTechnologyAnnual

$$\text{ProductionByTechnologyAnnual}[r,t,f,y] = \sum_{l}\text{ProductionByTechnology}[r,l,t,f,y]$$

3.20.2. RE2_TechIncluded

$$\text{TotalREProductionAnnual}[r,y] = \sum_{t,f}\text{ProductionByTechnologyAnnual}[r,t,f,y] \times \text{RETagTechnology}[r,t,y]$$

3.20.3. RE3_FuelIncluded

$$\text{RETotalProductionOfTargetFuelAnnual}[r,y] = \sum_{l,f}(\text{RateOfProduction}[r,l,f,y] \times \text{YearSplit}[l,y] \times \text{RETagFuel}[r,f,y])$$

3.20.4. RE4_EnergyConstraint

$$\text{TotalREProductionAnnual}[r, y] \geq \text{REMinProductionTarget}[r, y] \times \text{RETotalProductionOfTargetFuelAnnual}[r, y]$$

3.20.5. RE5_FuelUseByTechnologyAnnual

$$\text{UseByTechnologyAnnual}[r, t, f, y] = \sum_{l} (\text{RateOfUseByTechnology}[r, l, t, f, y] \times \text{YearSplit}[l, y])$$

3.21. Emissions Accounting (E)

Calculates each technology's annual and model period emissions for each category of emission. It also estimates any relevant total emission penalties. Finally, it guarantees that emissions do not exceed predetermined limitations, which can be set for each year or the entire model period. This operation can be accomplished through the use of **Sets** such as Year (y), Region (r), Technology (t), Emission (e) and Mode_Of_Operation (m). As well as through using **Parameters** including **Global Parameters** which are strictly positive like DiscountRate[r] and in **Emissions** we use EmissionActivityRatio[r,t,e,m,y], EmissionsPenalty[r,e,y], ModelPeriodEmissionLimit[r,e], ModelPeriodExogenousEmission[r,e], AnnualEmissionLimit[r,e,y] and AnnualExogenousEmission[r,e,y]. Ultimately through the use of **Variables** which are positive including **Activity variables** TotalAnnualTechnologyActivityByMode[r,t,m,y]. **And Reserve Margin variables** such as AnnualTechnologyEmissionByMode[r,t,e,m,y], AnnualTechnologyEmission[r,t,e,y], AnnualTechnologyEmissionPenaltyByEmission[r,t,e,y], AnnualTechnologyEmissionsPenalty[r,t,y], DiscountedTechnologyEmissionsPenalty[r,t,y], AnnualEmissions[r,e,y] and ModelPeriodEmissions[r,e].

3.21.1. E1_AnnualEmissionProductionByMode

$$\text{AnnualTechnologyEmissionByMode}[r, t, e, m, y] = \text{EmissionActivityRatio}[r, t, e, m, y] \times \text{TotalAnnualTechnologyActivityByMode}[r, t, m, y]$$

3.21.2. E2_AnnualEmissionProduction

$$\text{AnnualTechnologyEmission}[r, t, e, y] = \sum_{m} \text{AnnualTechnologyEmissionByMode}[r, t, e, m, y]$$

3.21.3. E3_EmissionsPenaltyByTechAndEmission

$$\text{AnnualTechnologyEmissionPenaltyByEmission}[r, t, e, y] = \text{AnnualTechnologyEmission}[r, t, e, y] \times \text{EmissionsPenalty}[r, e, y]$$

3.21.4. E4_EmissionsPenaltyByTechnology

$$\text{AnnualTechnologyEmissionsPenalty}[r, t, y] = \sum_{e} \text{AnnualTechnologyEmissionPenaltyByEmission}[r, t, e, y]$$

3.21.5. E5_DiscountedEmissionsPenaltyByTechnology

$$\text{DiscountedTechnologyEmissionsPenalty}[r, t, y] = \frac{\text{AnnualTechnologyEmissionsPenalty}[r, t, y]}{(1 + \text{DiscountRate}[r])^{(y - \min(yy) + 0.5)}}$$

3.21.6. E6_EmissionsAccounting1 {r in REGION, e in EMISSION, y in YEAR}:

$$\text{AnnualEmissions}[r, e, y] = \sum_{t} \text{AnnualTechnologyEmission}[r, t, e, y]$$

3.21.7. E7_EmissionsAccounting2 {r in REGION, e in EMISSION}:

$$\sum_{y} \text{AnnualEmissions}[r, e, y] = \text{ModelPeriodEmissions}[r, e] - \text{ModelPeriodExogenousEmission}[r, e]$$

3.21.8. E8_AnnualEmissionsLimit

$$\text{AnnualEmissionLimit}[r, e, y] \geq \text{AnnualEmissions}[r, e, y] + \text{AnnualExogenousEmission}[r, e, y]$$

3.21.9. E9_ModelPeriodEmissionsLimit {r in REGION, e in EMISSION}:

$$ModelPeriodEmissions[r, e] <= ModelPeriodEmissionLimit[r, e]$$

4- CONCLUSION:

This paper summarized the mathematical algebraic formulation, with a plain description of sets parameters and variables used in each block of OSeMOSYS. What makes the use of OSeMOSYS interesting is its open accessible nature, clear levels of abstraction and the potential for its use and development of an online community, where new development ideas can be proposed, showcased, and executed. This is aimed at promoting meaningful model development through academic projects in particular (including, for example, postgraduate student input in the form of thesis work). Open workshops (such as those hosted by KTH (Royal Swedish Institute of Technology)).

BIBLIOGRAPHY:

1. **Dato, P., Dato, P., & Grenoble, U.** (2017). *La transition énergétique To cite this version : HAL Id : tel-01445740.*
2. **Doukas, H., Flamos, A., & Lieu, J.** (2019). *Understanding Risks and Uncertainties in Energy and Climate Policy.*
3. **Howells, M., Rogner, H., Strachan, N., Heaps, C., Huntington, H., Kypreos, S., Hughes, A., Silveira, S., DeCarolis, J., Bazillian, M., & Roehrl, A.** (2011). OSeMOSYS: The Open Source Energy Modeling System. An introduction to its ethos, structure and development. *Energy Policy*, *39*(10), 5850–5870. https://doi.org/10.1016/j.enpol.2011.06.033.
4. **Welsch, M., Howells, M., Bazilian, M., DeCarolis, J. F., Hermann, S., & Rogner, H. H.** (2012). Modelling elements of Smart Grids - Enhancing the OSeMOSYS (Open Source Energy Modelling System) code. *Energy*, *46*(1), 337–350. https://doi.org/10.1016/j.energy.2012.08.017.
5. *OSeMOSYS Documentation Release 0.0.1 KTH-dESA.* (2021).
6. http://www.osemosys.org
7. https://github.com/OSeMOSYS/OSeMOSYS