EMB3RS-TEO-Module

Release 1.0.0

Shravan Kumar, Francesco Gardumi, Jagruti Thakur

Apr 22, 2022

CONTENTS:

1 The EMB3RS Techno-Economic Optimization Module **1**

1.1 EMB3RS project . . . 1

1.1.1 Core functionalities (CF) module . . . 1

1.1.2 GIS module . . . 2

1.1.3 TEO Module . . . 2

1.1.4 Market Module . . . 2

1.1.5 Business Module . . . 2

1.2 Introduction to EMB3RS TEO Module . . . 2

1.2.1 GitHub Repository . . . 3

1.2.2 License . . . 3

1.3 Main Features of TEO Module . . . 3

1.4 Sets . . . 4

1.5 Parameters . . . 5

1.6 Variables . . . 6

1.7 Links to other modules . . . 8

1.8 Running a test case . . . 11

1.8.1 Description of the test case . . . 11

1.8.2 Data and instruction to run the model . . . 12

1.8.3 Note on the solvers . . . 13

1.9 Results from the TEO . . . 13

1.10 Contributing to the TEO module . . . 17

1.10.1 Some resources: . . . 17

1.10.2 Bugs . . . 17

1.10.3 Ideas and Suggestions . . . 17

1.10.4 Git Workflow . . . 17

1.11 References . . . 18

CHAPTER

ONE

THE EMB3RS TECHNO-ECONOMIC OPTIMIZATION MODULE

This manual describes a modelling tool created within the Horizon 2020 EMB3RS project: a Techno-Economic Optimization module for least-cost recovery of excess industrial heat and cold. The EMB3RS project has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 847121. This module is part of a larger assessment toolbox called 'EMB3RS platform'. More information on the EMB3RS project can be found at https://www.emb3rs.eu/ .

1.1 EMB3RS project

EMB3Rs ("User-driven Energy-Matching & Business Prospection Tool for Industrial Excess Heat/Cold Reduction, Recovery and Redistribution) is a European project funded under the H2020 Programme (Grant Agreement No.847121) to develop an open-sourced tool to match potential sources of excess thermal energy with compatible users of heat and cold. More information about the EMB3RS project can be found on the EMB3RS website .

Users, like industries and other sources that produce excess heat, will provide the essential parameters, such as their location and the available excess thermal energy. The EMB3Rs platform will then autonomously and intuitively assess the feasibility of new business scenarios and identify the technical solutions to match these sources with compatible sinks. End users such as building managers, energy communities or individual consumers will be able to determine the costs and benefits of industrial excess heat and cold utilisation routes and define the requirements for implementing the most promising solutions. The EMB3Rs platform will integrate several analysis modules that will allow a full exploration of the feasible technical routes to the recovery and use of the available excess thermal energy. Several other modules are a part of the EMB3RS platform. Each module will be used to conduct a special purpose task or analysing excess heat and cold recovery. The models and their primary functionalities are specified below

1.1.1 Core functionalities (CF) module

The purpose of the CF module is to provide a comprehensive quantification of the energy flows of the EMB3RS platform objects (sinks, sources, and links) and costs associated with different options for excess H/C recovery and use. This information is used by the other analysis modules (GIS, TEO, MM and BM) to perform simulations according to user specifications. The CF module has two main functionalities:

1. Full characterization of objects – e.g., in terms of processes, equipment, building characteristics
2. To carry out a preliminary analysis of available supply and demand - described as a simulation feature within the CF.

1.1.2 GIS module

The purpose of the GIS model within EMB3Rs is to analyse possible network solutions for a given set of sources and sinks as well as an assumption of related network heat/cold losses and costs. The GIS thereby finds such a network solution along the existing Open Street Map (OSM) Road Network connecting all sources and sinks. It currently outputs a graph/map that lets the user check the specifications of every single pipe element from the network found and a table that illustrates all source/sink specific losses, costs, network length and installed pipe capacity.

1.1.3 TEO Module

The TEO module identifies least-cost combinations of technologies for using and conveying excess heating or cooling (HC) from defined sources to defined sinks. The user (representing the excess heat producer - i.e., source – or a demand point – i.e., sink) wants to evaluate the least-cost options of utilising excess HC generated to meet the heating/cooling demand for one or more known/assumed sinks. The objective of the optimisation is to find the least-cost mix of technologies (in terms of installed capacities – typically, in power units) and match between sources and sinks (in terms of energy flows) that satisfies the demands under constraints dictated by regulation, availability of heat, load profiles, techno-economic characteristics of technologies, investment plans.

1.1.4 Market Module

The Market Module (MM) will provide the user with economic and fairness indicators like energy transaction, market price, social welfare, fairness among prices. This will be done by short-term and long-term market analyses that simulate various market structures and incorporate business conditions and network models. The MM will consider the existing Pool market as well as new forms of a decentralized market based on peer-to-peer and community systems. The modelling of heat/cold sources and sinks will include flexibility, offering price and business preferences.

1.1.5 Business Module

Business Model Module evaluates various business models for DHC which incorporates excess heat. This is done by calculating matrices like Net Present Value (NPV), Levelized Cost of Heat (LCOH) and Internal Rate of Return (IRR) under different ownership structures and market frameworks.

1.2 Introduction to EMB3RS TEO Module

The Techno-Economic Optimization (TEO) module identifies least-cost combinations of technologies for using and conveying excess Heat and Cold (HC) from defined sources to defined sinks. The user (representing the excess heat producer - i.e., source – or a demand point – i.e., sink) wants to evaluate the options of utilizing excess HC generated to meet the heating/cooling demand for one or more known/assumed sinks. The objective of the optimisation is to find the least-cost mix of technologies and match between sources and sinks that satisfies the demands under constraints dictated by regulation, availability of heat, load profiles, techno-economic characteristics of technologies, investment plans, etc. The mix of technologies may include the District Heating/Cooling Network (DH/CN), technologies to upgrade the Temperature level on the sink or the source side, thermal storage on the sink or the source side, as well as heating alternatives.

The main requirements of the tool for building the techno-economic optimisation module within the EMB3RS framework are:

- High temporal resolution – Daily to hourly
- Low simulation time
- High flexibility and ability to be modified

- Interaction and interoperation with other modules
- Open access

The techno-economic optimisation module should provide the following results: • The technology mix (in terms of existing and newly installed yearly capacities in terms of energy flows throughout the supply-demand chain)

- Share of each technology in meeting the demand in any time step of the analysis (where the time resolution is defined by the user within certain limits) and throughout the analysis period
- Annual costs (investment, fuel, operation & maintenance, Levelized costs of heat (LCOH) etc.) associated with the technologies
- Emissions, emission savings and emission costs over the defined period.

1.2.1 GitHub Repository

The standalone version of the TEO module can be found on the GitHub reposiorty here .

1.2.2 License

The TEO module is licensed under Apache License Version 2.0, January 2004. You can find more information about the License here . The lincese file cna be found on the GitHub repository here .

1.3 Main Features of TEO Module

- The TEO module optimizes the matching between the different sources and the sinks, while taking into account various technical and economic constraints, such as demand profiles, technology cost, efficiencies and losses while also considering thermal energy storage.
- Optimal mix of investments in technologies and optimal capacities in storage and district heating network can be determined. The operation of the technologies and the intra annual heat supply is also optimized in the module.
- The module can also analyze the competition between centralized and de-centralized. The current input data includes competition between waste heat sources and decentralized solar thermal based heating solutions
- The TEO module optimizes the matching between the different sources and the sinks, while taking into account various technical and economic constraints, such as demand profiles, technology cost, efficiencies and losses while also considering thermal energy storage.
- The TEO module carries out a socio-economic type of optimisation, where the total system cost is minimised, irrespective of who bears it. It does not take a policy-maker, or investor, or business perspective.
- The time domain, time resolution and technological options are flexible and chosen by the user. For example, an analysis can be carried out for a time domain of 5, or 10 or 30 years. Similarly, the time resolution can be of few time steps in a year, up to 8760 hourly time steps. The types of technologies that can be modelled include: heat exchanger, heat pumps, boilers etc. The module is a model generator, where none of the above is pre-defined. The user must define all the technologies and the commodiies and crate links between the two.
- The module relies on two core types of objects: Technologies and Fuels. These are very flexibly defined, so that many different processes and commodities can be represented in a model. A Technology is nothing but a process - I.e. a box – with inputs, outputs, a transfer function between them, and several associated techno-economic characteristics. A Fuel is any commodity entering or exiting a Technology. Therefore, with a Technology the user may represent a heat exchanger or a heat pump and for the Fuel, the user may represent electricity or the excess heat stream.

1.4 Sets

In OSeMOSYS, like usually in linear programs, sets, parameters and variables are defined. In this section, the sets, parameters and variables existing in OSeMOSYS are listed and briefly described. Illustrative examples on how to assign values for each of these will be given in the following Sections.

The 'sets' define the physical structure of a model, usually independent from the specific scenarios which will be run. They define the time domain and time split, the spatial coverage, the technologies and energy vectors to be considered, etc. For instance, when a variable is defined as a function of the set 'YEAR' it will be indicated as **variablename[y]** at it will be computed for every year listed in the set. The sets of OSeMOSYS are presented in the Table below.

Name	Description	Index
YEAR	It represents the time frame of the model, it contains all the years to be considered in the study.	y
TECHNOLOGY	It includes any element of the energy system that changes a commodity from one form to another, uses it or supplies it. All system components are set up as a 'technology' in OSeMOSYS. As the model is an abstraction, the modeller is free to interpret the role of a technology at will, where relevant. It may for example represent a single real technology (such as a power plant) or can represent a heavily aggregated collection of technologies (such as the stock of several million light bulbs), or may even simply be a 'dummy technology', perhaps used for accounting purposes.	t
TIMESLICE	It represents the time split of each modelled year, therefore the time resolution of the model. Common to several energy systems modelling tools (incl. MESSAGE / MARKAL / TIMES), the annual demand is 'sliced' into representative fractions of the year. It is necessary to assess times of the year when demand is high separately from times when demand is low, for fuels that are expensive to store. In order to reduce the computation time, these 'slices' are often grouped. Thus, the annual demand may be split into aggregate seasons where demand levels are similar (such as 'summer, winter and intermediate'). Those seasons may be subdivided into aggregate 'day types' (such as workdays and weekends), and the day further sub divided (such as into day and night) depending on the level of demand.	l
FUEL	It includes any energy vector, energy service or proxies entering or exiting technologies. These can be aggregate groups, individual flows or artificially separated, depending on the requirements of the analysis.	f
EMISSION	It includes any kind of emission potentially deriving from the operation of the defined technologies. Typical examples would include atmospheric emissions of greenhouse gasses, such as CO2.	e
MODE_OF_OPERATION	It defines the number of modes of operation that the technologies can have. If a technology can have various input or output fuels and it can choose the mix (i.e. any linear combination) of these input or output fuels, each mix can be accounted as a separate mode of operation. For example, a CHP plant may produce heat in one mode of operation and electricity in another.	m
REGION	It sets the regions to be modelled, e.g. different countries. For each of them, the supply-demand balances for all the energy vectors are ensured, including trades with other regions. In some occasions it might be computationally more convenient to model different countries within the same region and differentiate them simply by creating ad hoc fuels and technologies for each of them.	r
SEASON	It gives indication (by successive numerical values) of how many seasons (e.g. winter, intermediate, summer) are accounted for and in which order. This set is needed if storage facilities are included in the model.	ls
DAYTYPE	It gives indication (by successive numerical values) of how many day types (e.g. workday, weekend) are accounted for and in which order. This set is needed if storage facilities are included in the model.	ld
DAILYTIMEBRACKET	It gives indication (by successive numerical values) of how many parts the day is split into (e.g. night, morning, afternoon, evening) and in which order these parts are sorted. This set is needed if storage facilities are included in the model.	lh
STORAGE	It includes storage facilities in the model.	s

1.5 Parameters

The parameters are the user-defined numerical inputs to the model. While usually the structure of a model, therefore the sets, remains fixed across scenarios, it is common practice to change the values of some parameters when running different scenarios and/or sensitivity analyses. As will be clear in the following, each parameter is a function of the elements in one or more sets. For instance, CapitalCost[r, t, and y] indicates that the capital cost is a function of the region (r), the technology (t) and the year (y). A list and brief description of the parameters declared in the master version of OSeMOSYS is given in the below table [3].

Parameter Name	Description
YearSplit[l,y]	Duration of a modelled time slice, expressed as a fraction of the year. The sur
DiscountRateTech[r,t]	Technology specific value for the discount rate, expressed in decimals (e.g. 0.
DiscountRateSto[r,t]	Storage specific value for the discount rate, expressed in decimals (e.g. 0.05).
DaySplit[lh,y]	Length of one DailyTimeBracket in one specific day as a fraction of the year (
Conversionls[l,ls]	Binary parameter linking one TimeSlice to a certain Season. It has value 0 if
Conversionld[ld,l]	Binary parameter linking one TimeSlice to a certain DayType. It has value 0
Conversionlh[lh,l]	Binary parameter linking one TimeSlice to a certain DaylyTimeBracket. It ha
DaysInDayType[ls,ld,y]	Number of days for each day type, within one week (natural number, ranging
TradeRoute[r,rr,f,y]	Binary parameter defining the links between region r and region rr, to enable
DepreciationMethod[r]	Binary parameter defining the type of depreciation to be applied. It has value
Demands	
SpecifiedAnnualDemand[r,f,y]	Total specified demand for the year, linked to a specific ‘time of use’ during th
SpecifiedDemandProfile[r,f,l,y]	Annual fraction of energy-service or commodity demand that is required in ea
AccumulatedAnnualDemand[r,f,y]	Accumulated Demand for a certain commodity in one specific year. It cannot
Performance	
CapacityToActivityUnit[r,t]	Conversion factor relating the energy that would be produced when one unit o
AvailabilityFactor[r,t,y]	Capacity available eachTimeSlice expressed as a fraction of the total installed
OperationalLife[r,t]	Maximum time a technology can run in the whole year, as a fraction of the ye
ResidualCapacity[r,t,y]	Useful lifetime of a technology, expressed in years.
InputActivityRatio[r,t,f,m,y]	Capacity available from before the modelling period.
OutputActivityRatio[r,t,f,m,y]	Rate of use of a commodity by a technology, as a ratio of the rate of activity.
Technology costs	
OuputModeofOpeartion[r,t,m,y],	Binary parameter indicating the mode of operation in which a technology has
CapitalCost[r,t,y]	Capital investment cost of a technology, per unit of capacity.
VariableCost[r,t,m,y]	Cost of a technology for a given mode of operation (Variable O&M cost), per
FixedCost[r,t,y]	Fixed O&M cost of a technology, per unit of capacity.
Storage	
TechnologyToStorage[r,t,s,m]	Binary parameter linking a technology to the storage facility it charges. It has
TechnologyFromStorage[r,t,s,m]	Binary parameter linking a storage facility to the technology it feeds. It has va
StorageLevelStart[r,s]	Level of storage at the beginning of first modelled year, in units of activity.
StorageMaxChargeRate[r,s]	Maximum charging rate for the storage, in units of activity per year.
StorageMaxDischargeRate[r,s]	Maximum discharging rate for the storage, in units of activity per year.
MinStorageCharge[r,s,y]	It sets a lower bound to the amount of energy stored, as a fraction of the maxi
OperationalLifeStorage[r,s]	Useful lifetime of the storage facility.
CapitalCostStorage[r,s,y]	Binary parameter linking a technology to the storage facility it charges. It has
ResidualStorageCapacity[r,s,y]	Binary parameter linking a storage facility to the technology it feeds. It has va
Capacity constraints	
CapacityOfOneTechnologyUnit[r,t,y]	Capacity of one new unit of a technology. In case the user sets this parameter
TotalAnnualMaxCapacity[r,t,y]	Total maximum existing (residual plus cumulatively installed) capacity allowe
TotalAnnualMinCapacity[r,t,y]	Total minimum existing (residual plus cumulatively installed) capacity allowe

Investment constraints	
TotalAnnualMaxCapacityInvestment[r,t,y]	Maximum capacity of a technology, expressed in power units.
TotalAnnualMinCapacityInvestment[r,t,y]	Minimum capacity of a technology, expressed in power units.
Activity constraints	
TotalTechnologyAnnualActivityUpperLimit[r,t,y]	Total maximum level of activity allowed for a technology in one year.
TotalTechnologyAnnualActivityLowerLimit[r,t,y]	Total minimum level of activity allowed for a technology in one year.
TotalTechnologyModelPeriodActivityUpperLimit[r,t]	Total maximum level of activity allowed for a technology in the entire modell
TotalTechnologyModelPeriodActivityLowerLimit[r,t]	Total minimum level of activity allowed for a technology in the entire modelle
Reserve margin	
ReserveMarginTagTechnology[r,t,y]	Binary parameter tagging the technologies that are allowed to contribute to th
ReserveMarginTagFuel[r,f,y]	Binary parameter tagging the fuels to which the reserve margin applies. It has
ReserveMargin[r,y]	Minimum level of the reserve margin required to be provided for all the tagge
RE Generation target	
RETagTechnology[r,t,y]	Binary parameter tagging the renewable technologies that must contribute to
RETagFuel[r,f,y]	Binary parameter tagging the fuels to which the renewable target applies to. I
REMinProductionTarget[r,y]	Minimum ratio of all renewable commodities tagged in the RETagCommodity
Emissions	
EmissionActivityRatio[r,t,e,m,y]	Emission factor of a technology per unit of activity, per mode of operation.
EmissionsPenalty[r,e,y]	Penalty per unit of emission.
AnnualExogenousEmission[r,e,y]	It allows the user to account for additional annual emissions, on top of those c
AnnualEmissionLimit[r,e,y]	Annual upper limit for a specific emission generated in the whole modelled re
ModelPeriodExogenousEmission[r,e]	It allows the user to account for additional emissions over the entire modelled
ModelPeriodEmissionLimit[r,e]	Annual upper limit for a specific emission generated in the whole modelled re
StorageUvalue[r,s]	Heat transfer co-efficient of the thermal energy storage tank.
StorageFlowTemperature[r,s]	The temperature of water inflow into thermal energy storage.
StorageReturnTemperature[r,s]	The return water temperature in the heating grid where the thermal energy sto
StorageAmbientTemperature[r,s]	The ambient temperature of the locations where the thermal energy storage is
StorageL2D[r,s]	Binary parameter which indicates the length to diameter ratio of the thermal e
Storagetagheating[r,s]	Binary parameter indicating whether the thermal energy storage is connected
Storagetagcooling[r,s]	Binary parameter indicating whether the thermal energy storage is connected

1.6 Variables

The variables are the outputs computed by the code. As much as the parameters, also the variables are functions of the elements in one or more sets. In the following Table, a list and brief description of all the variables computed by the code of OSeMOSYS (in its full version) is given. As will be explained next in this manual, a shortened version of OSeMOSYS has been created, to improve the computational capability at the expenses of the readability of the code. In such version, only some of the variables here listed are computed. When reasonable, the domain of several variables has been constrained to be positive, in order to decrease the size of the solution space and therefore the computational effort. The list of variables is shown in the below table [3].

Name	Description
RateOfDemand[r,l,f,y]>=0	Intermediate variable. It represents the energy that would be demanded
Demand[r,l,f,y]>=0	Demand for one fuel in one time slice.
RateOfStorageCharge[r,s,ls,ld,lh,y]	Intermediate variable. It represents the commodity that would be charg
RateOfStorageDischarge[r,s,ls,ld,lh,y]	Intermediate variable. It represents the commodity that would be disch
NetChargeWithinYear[r,s,ls,ld,lh,y]	Net quantity of commodity charged to storage facility s in year y. It is a

NetChargeWithinDay[r,s,ls,ld,lh,y]	Net quantity of commodity charged to storage facility s in daytype ld. I
StorageLevelYearStart[r,s,y]>=0	Level of stored commodity in storage facility s in the first time step of y
StorageLevelYearFinish[r,s,y]>=0	Level of stored commodity in storage facility s in the last time step of y
StorageLevelSeasonStart[r,s,ls,y]>=0	Level of stored commodity in storage facility s in the first time step of s
StorageLevelDayTypeStart[r,s,ls,ld,y]>=0	Level of stored commodity in storage facility s in the first time step of d
StorageLevelDayTypeFinish[r,s,ls,ld,y]>=0	Level of stored commodity in storage facility s in the last time step of d
StorageLowerLimit[r,s,y]>=0	Minimum allowed level of stored commodity in storage facility s, as a f
StorageUpperLimit[r,s,y]>=0	Maximum allowed level of stored commodity in storage facility s. It co
AccumulatedNewStorageCapacity[r,s,y]>=0	Cumulative capacity of newly installed storage from the beginning of th
NewStorageCapacity[r,s,y]>=0	Capacity of newly installed storage in year y.
StorageLevelTimesliceStart[r,s,l,y]	Energy stored in storage in timeslice l.
StorageLosses[r,s,l,y]	Thermal energy losses from the storage in timeslice l.
CapitalInvestmentStorage[r,s,y]>=0	Undiscounted investment in new capacity for storage facility s. Derived
DiscountedCapitalInvestmentStorage[r,s,y]>=0	Investment in new capacity for storage facility s, discounted through the
SalvageValueStorage[r,s,y]>=0	Salvage value of storage facility s in year y, as a function of the parame
DiscountedSalvageValueStorage[r,s,y]>=0	Salvage value of storage facility s, discounted through the parameter D
TotalDiscountedStorageCost[r,s,y]>=0	Difference between the discounted capital investment in new storage fa
NumberOfNewTechnologyUnits[r,t,y]>=0, integer	Number of newly installed units of technology t in year y, as a function
NewCapacity[r,t,y]>=0	Newly installed capacity of technology t in year y.
AccumulatedNewCapacity[r,t,y]>=0	Cumulative newly installed capacity of technology t from the beginning
TotalCapacityAnnual[r,t,y]>=0	Total existing capacity of technology t in year y (sum of cumulative new
RateOfActivity[r,l,t,m,y] >=0	Intermediate variable. It represents the activity of technology t in one r
RateOfTotalActivity[r,t,l,y] >=0	Sum of the RateOfActivity of a technology over the modes of operation
TotalTechnologyAnnualActivity[r,t,y] >=0	Total annual activity of technology t.
TotalAnnualTechnologyActivityByMode[r,t,m,y] >=0	Annual activity of technology t in mode of operation m.
TotalTechnologyModelPeriodActivity[r,t]	Sum of the TotalTechnologyAnnualActivity over the years of the mode
RateOfProductionByTechnologyByMode[r,l,t,m,f,y] >=0	Intermediate variable. It represents the quantity of fuel f that technolog
RateOfProductionByTechnology[r,l,t,f,y] >=0	Sum of the RateOfProductionByTechnologyByMode over the modes o
ProductionByTechnology[r,l,t,f,y] >=0	Production of fuel f by technology t in time slice l.
ProductionByTechnologyAnnual[r,t,f,y] >=0	Annual production of fuel f by technology t.
RateOfProduction[r,l,f,y] >=0	Sum of the RateOfProductionByTechnology over all the technologies.
Production[r,l,f,y] >=0	Total production of fuel f in time slice l. It is the sum of the Production
RateOfUseByTechnologyByMode[r,l,t,m,f,y] >=0	Intermediate variable. It represents the quantity of fuel f that technolog
RateOfUseByTechnology[r,l,t,f,y] >=0	Sum of the RateOfUseByTechnologyByMode over the modes of opera
UseByTechnologyAnnual[r,t,f,y] >=0	Annual use of fuel f by technology t.
UseByTechnology[r,l,t,f,y] >=0	Use of fuel f by technology t in time slice l.
Use[r,l,f,y] >=0	Total use of fuel f in time slice l. It is the sum of the UseByTechnology
Trade[r,rr,l,f,y]	Quantity of fuel f traded between region r and rr in time slice l.
TradeAnnual[r,rr,f,y]	Annual quantity of fuel f traded between region r and rr. It is the sum o
ProductionAnnual[r,f,y] >=0	Total annual production of fuel f. It is the sum of the variable Productio
UseAnnual[r,f,y] >=0	Total annual use of fuel f. It is the sum of the variable Use over all tech
CapitalInvestment[r,t,y] >=0	Undiscounted investment in new capacity of technology t. It is a functio
DiscountedCapitalInvestment[r,t,y] >=0	Investment in new capacity of technology t, discounted through the par
SalvageValue[r,t,y] >=0	Salvage value of technology t in year y, as a function of the parameters
DiscountedSalvageValue[r,t,y] >=0	Salvage value of technology t, discounted through the parameter Disco
OperatingCost[r,t,y] >=0	Undiscounted sum of the annual variable and fixed operating costs of te
DiscountedOperatingCost[r,t,y] >=0	Annual OperatingCost of technology t, discounted through the paramet
AnnualVariableOperatingCost[r,t,y] >=0	Annual variable operating cost of technology t. Derived from the Total
AnnualFixedOperatingCost[r,t,y] >=0	Annual fixed operating cost of technology t. Derived from the TotalCap
TotalDiscountedCostByTechnology[r,t,y] >=0	Difference between the sums of discounted operating cost / capital cost

TotalDiscountedCost[r,y] >=0	Sum of the TotalDiscountedCostByTechnology over all the technologie
ModelPeriodCostByRegion[r] >=0	Sum of the TotalDiscountedCost over all modelled years.
TotalCapacityInReserveMargin[r,y] >=0	Total available capacity of the technologies required to provide reserve
DemandNeedingReserveMargin[r,l,y] >=0	Quantity of fuel produced that is assigned to a target of reserve margin.
TotalREProductionAnnual[r,y]	Annual production by all technologies tagged as renewable in the mode
RETotalProductionOfTargetFuelAnnual[r,y]	Annual production of fuels tagged as renewable in the model. Derived
AnnualTechnologyEmissionByMode[r,t,e,m,y] >=0	Annual emission of agent e by technology t in mode of operation m. D
AnnualTechnologyEmission[r,t,e,y] >=0	Sum of the AnnualTechnologyEmissionByMode over the modes of ope
AnnualTechnologyEmissionPenaltyByEmission[r,t,e,y] >=0	Undiscounted annual cost of emission e by technology t. It is a functio
AnnualTechnologyEmissionsPenalty[r,t,y] >=0	Total undiscounted annual cost of all emissions generated by technolog
DiscountedTechnologyEmissionsPenalty[r,t,y] >=0	Annual cost of emissions by technology t, discounted through the Disc
AnnualEmissions[r,e,y] >=0	Sum of the AnnualTechnologyEmission over all technologies.
ModelPeriodEmissions[r,e] >=0	Total system emissions of agent e in the model period, accounting for b

1.7 Links to other modules

This section describes the links between the TEo and the other modules in the EMB3RS platform.

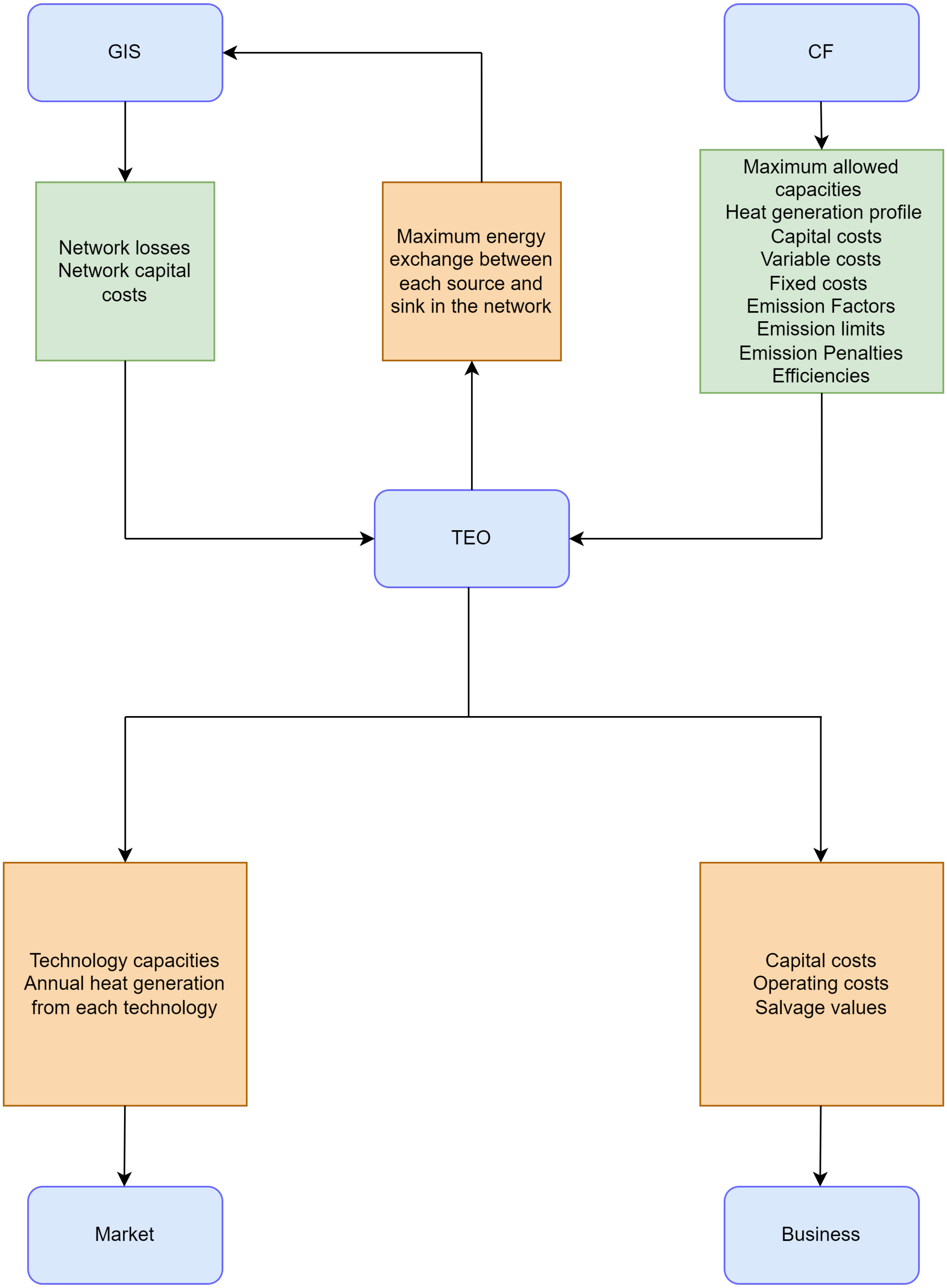
The TEO module is linked to all the modules in the EMB3RS. The module obtains inputs from the CF module, has a two-way input-output link with the GIS module and provides inputs for the market and business modules.

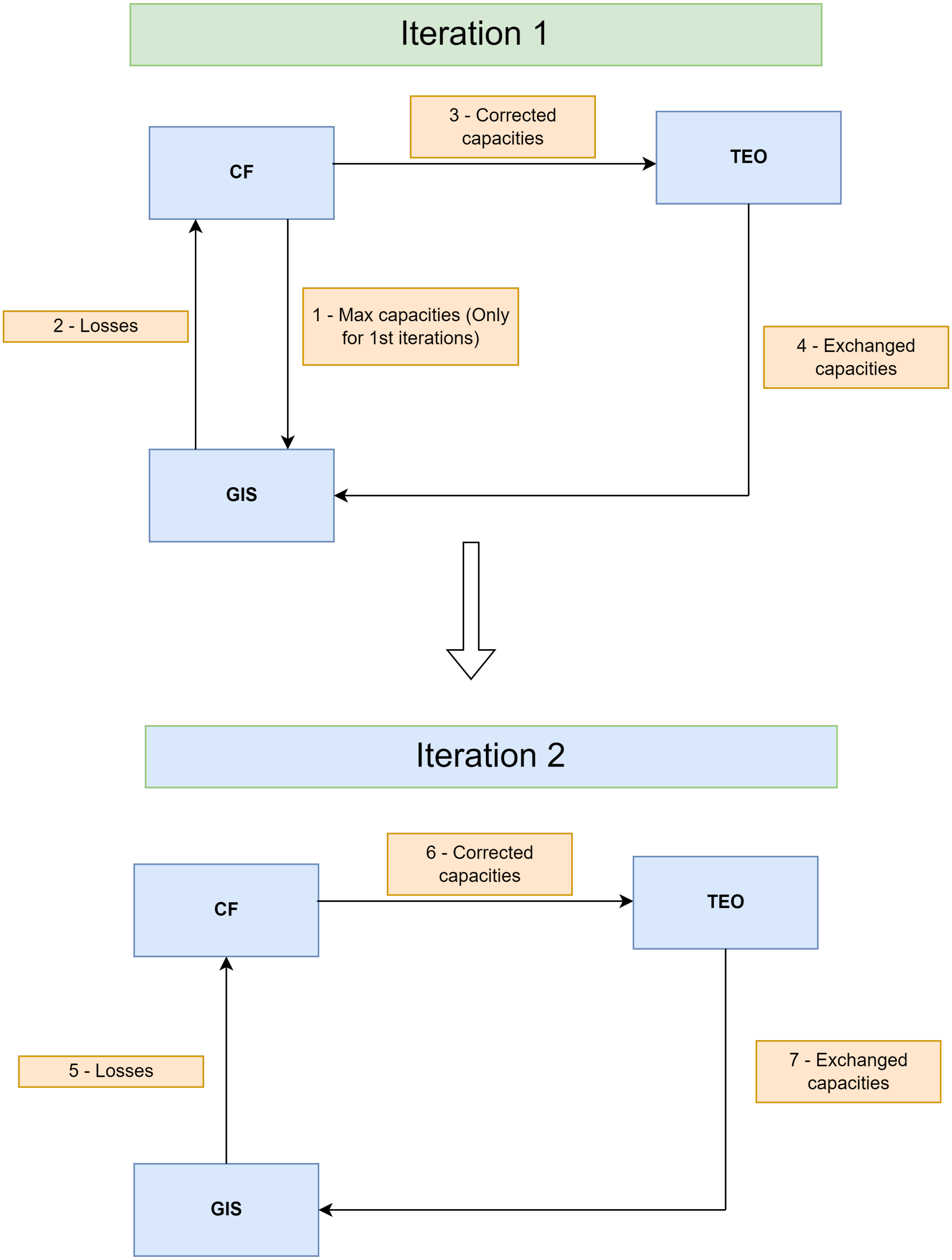
The below figure shows the inputs to the TEO and the outputs from the TEO to the modules.

The CF accesses the emission factors, limits and penalties from the regulatory framework. These are then passed to the TEO.

The TEO module is linked to the GIS and CF module to design the capacities of the heat generation technologies and the heating network. The outputs from TEO are given to the GIS module for the iterative process. Other than this, the output from the TEO module provides some inputs to the market and business module. The iteration between the TEO, the CF and the GIS modules will take place in two steps. In the first step, the GIS module will provide values for DHC costs and average network losses considering all possible connections in the network. The losses in the network consist both of energy and exergy losses. The CF module calculates the maximum possible heat generation capacities at the sources based on the temperature availability. To compensate for the exergy losses, the technologies on the source side must produce heat at higher temperatures. The CF module will account for the exergy losses and calculate the maximum possible heat generation capacities for all the sources side technologies. In some cases, additional temperature-boosting technologies such as heat pumps would be needed to overcome the exergy losses. The CF will provide the corrected maximum capacities of the technologies to the TEO. TEO will then determine the least cost matching of sources and sinks considering the energy losses in the network. Since the loss values from the GIS are as power losses (in terms of kW), these losses are added to the sink demand in each hour. The TEO then determines the optimal matching of sources and sinks. Based on this, the exchange capacities between the sources and the sinks i.e. the maximum exchange between each source and sinks are calculated. The exchange capacities indicate the hourly heat flow from each source to each sink in the network. The maximum hourly heat exchange between each source and sink will be used by the GIS module to design the pipe capacities in the district heating network. The sources and the sinks in the network and maximum hourly heat exchange between each source and sink will be fed back to the GIS. In Some cases, the TEO might discard certain sources due to the lack of profitability. This information is also passed on to the GIS module.

In the second step, the GIS will use the calculated maximum exchange capacities to determine the accurate losses and the investments costs of the DHC. These losses are once again fed into the CF to determine the corrected maximum capacities accounting for the exergy losses and forward this information to the TEO. These results are then fed into the TEO to obtain the accurate least-cost mix of technologies. A schematic of the iteration is shown in figure below.


GIS
CF
Network losses
Network capital costs
Maximum energy exchange between each source and sink in the network
Maximum allowed capacities
Heat generation profile
Capital costs
Variable costs
Fixed costs
Emission Factors
Emission limits
Emission Penalties
Efficiencies
TEO
Technology capacities
Annual heat generation from each technology
Capital costs
Operating costs
Salvage values
Market
Business


Iteration 1
3 - Corrected capacities
CF
TEO
2 - Losses
1 - Max capacities (Only for 1st iterations)
4 - Exchanged capacities
GIS
Iteration 2
6 - Corrected capacities
CF
TEO
5 - Losses
7 - Exchanged capacities
GIS

TEO-CF-GIS iterations.

In every iteration, the loss value from the GIS is monitored and used as a critical value of stopping the iteration. When the difference in the loss values in two consecutive iterations is below 0.01%, the iterations are stopped.

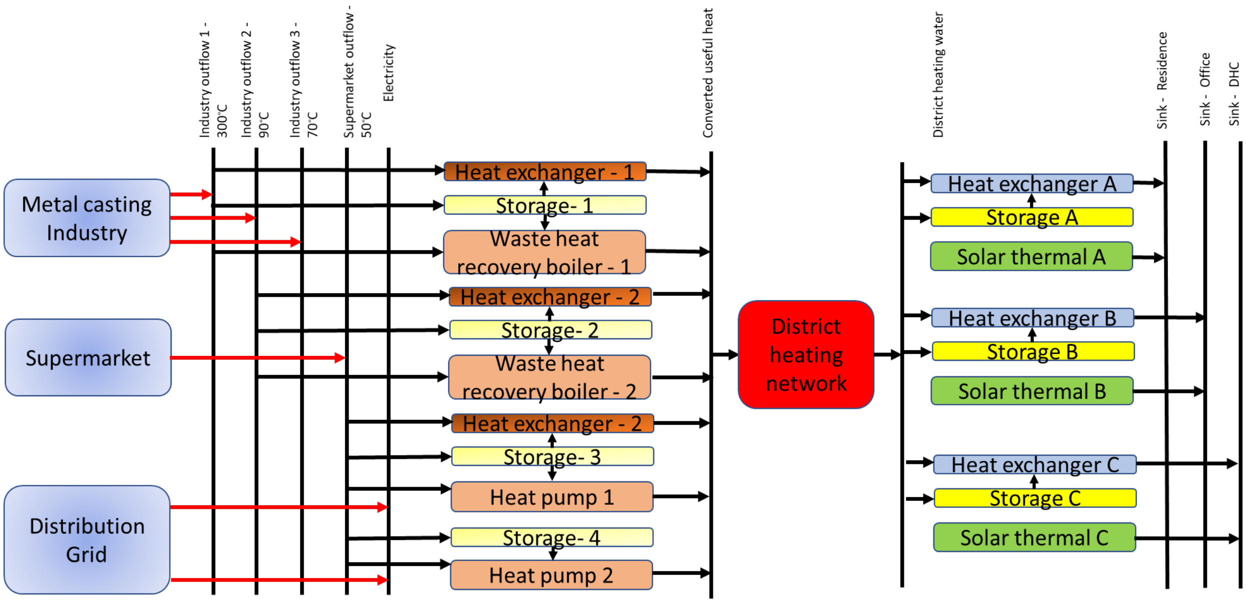
The market and business module directly use the TEO results. The market module uses the installed capacities of the different technologies to calculate the dispatch from each technology. The business module uses the capital investment and operation and maintenance costs of the different technologies and storage, and the salvage values to analyse the financial feasibility of the project.

1.8 Running a test case

1.8.1 Description of the test case

This section introduces the user to the basic components of any application of TEO and describes the steps for the creation of a model. To this end, a sample case study is used and examples from it are shown throughout the section.

The sample case study for TEO, represents a case of industrial excess heat recovery and use. The simple use case consists of two excess heat producers Supermarket and Metal casting Industry and three sink points District Heating and Cooling grid (DHC), office buildings and residential buildings. The excess heat load profiles for the source and sink points are be used. The boundaries of the system represented in the case study are shown in the schematic representation shown in the figure below.



The Reference Energy System (RES) of the Simple case study.

In the RES, the rectangles represent the technologies, the arrows represent the flow of energy and the vertical lines represent fuels. The RES is read from the left to the right. The primary energy supply side is on the left and the final energy demand side is on the right. The source nodes, metal casting industry and the supermarket, are modelled as technologies, whereas the sink nodes, being residential buildings, office buildings and DHC are modelled as fuels. The distribution grid has also been added as a technology as some components can either require or produce electricity. For the test case, the heat pumps in the system will require electricity to operate. The first set of vertical lines represent fuels at the primary level. These primary fuels consist of electricity, waste heat from the outflows of the metal casting industry at three different temperatures, and waste heat from supermarket. The waste heat from each source is supplied to a set of technologies. Here, we assume a Heat Exchanger (HE), Waste Heat Recovery Boiler (WHRB) for the first waste heat outflow from the industry. Similarly, the second outflow makes use of a HE and a WHRB. The third outflow is provided to a Heat Pump (HP) and a HE. The supermarket waste heat is provided with a HP.

The outflow temperatures of the metal casting industry are to be at 300°C, 90°C and 70°C Celsius for its three outflows whereas the waste heat from the supermarket is at 50°C. On the sinks' side, the DHC demand temperature is the same as the average supply temperature of the DHN, which is at 90°C. The demand temperature profile of the office buildings is the same as that of the residential buildings at 90°C. The sources and the sinks are equipped with storages. This implies that the generation and supply technologies be connected to a storage technology, and also, the demand technologies be connected to storage systems.

Only one storage can be chosen for a set of technologies. Here, each technology is connected to a storage option. Based on the technology that is selected from the optimization process, the energy will be stored from the technology in the corresponding storage.The technologies are assessed for feasibility and selected in the process. The converted useful heat at the suitable temperature (here, the network temperature) is stored and also supplied to the secondary fuel level.

The secondary level fuel is the converted useful heat. The converted useful heat is then supplied to the District Heating Network, which is modelled as a technology. The network is similar to the distribution grid being modelled as a technology, and they both account for losses. The heat from the network is supplied to all demand points by first being transformed into a tertiary level fuel of district heating water. To further assess the feasibility of the demand side system, solar technologies have been added. Solar thermal technologies have been added to all sinks.

1.8.2 Data and instruction to run the model

The input file for the prototype is 'Input_file_TEO.xlsx', which can be accessed at 'LINK'. To run the TEO, the code files named 'TEO_Model', 'TEO_functions', TEO_running_file', and the input file must be downloaded and saved in a specific manner. A main folder called 'TEO' must be created and the code will be downloaded into this folder. Within this main folder, two sub folders named 'Input_data' and 'Output_data' must be created. The input file must be saved into the 'Input_data' folder. A representation of how the files must be organised is shown below.

TEO (Main folder)

o Input_data

Input_file_TEO.xlsx

o Output_data

o TEO_Model

o TEO_functions

o TEO_running_file

Once the TEO analyses in completed, the results file named 'Input_file_TEO_Results.xlsx' will be save in 'Output_data' sub-folder. Once the files are downloaded and the folder structure is established, the model can be run using the TEO_running_file. The name of the input directory and the input file must be checked in the TEO_running_file. Since the name of the input file is to be checked and altered, it is advisable to open the TEO_running_file in a python IDE or a in other python notebook interface such as 'Jupyter lab' or 'Visual studio code'. Both these are freeware and can be downloaded. The TEO module can output results in two formats, excel and csv. The preference for the output format can also be set in the TEO_running_file by specifying a 'True' or 'False' next to the output formats in the TEO_running_file.

1.8.3 Note on the solvers

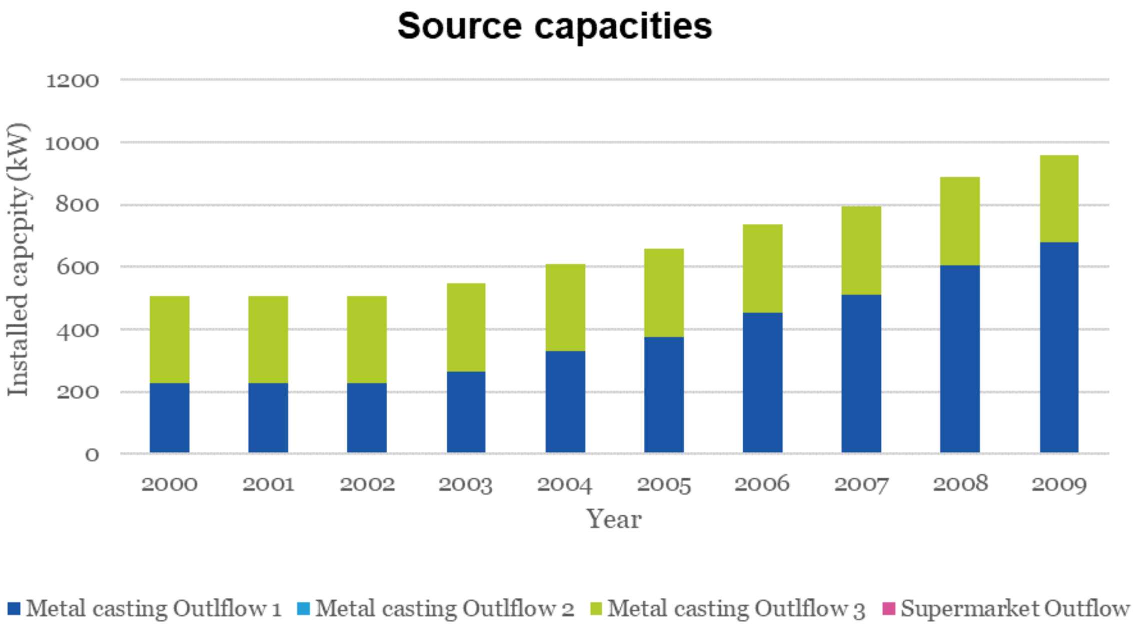
Two solvers, GLPK (GNU linear programming kit) and CBC (Coin-or branch and cut) are inbuilt in the PULP. In order to use other solvers, they should be downloaded and installed. Instruction for this can be found at 'https: //coin-or.github.io/pulp/guides/how_to_configure_solvers.html'. After the installation of the solver, the solver path needs to be added as an environment variable and then should be called into python using solver commands. The user can analyse the data based on the results saved in the output file. The user can also use other solvers such as CPLEX and Gurobi to run the TEO. The solver name and path must be specified in the TEO_running_file.

1.9 Results from the TEO

If run correctly, the Test Case will provide the results shown in the following. Note that these results are only a selection of the key outputs, condensing some of the important insights. Outputs for all the 'variables' listed in the previous sub-sections are calculated. They can also be extracted and visualised.

The technology mix (in terms of existing and newly installed yearly capacities in terms of energy flows throughout the supply-demand chain)

The installed capacities at the different sources is shown in Figure below. The model only uses 2 source outflows from the Metal casting industry. The excess heat from the supermarket is not used use due to the low temperatures and thus needs expensive investments.

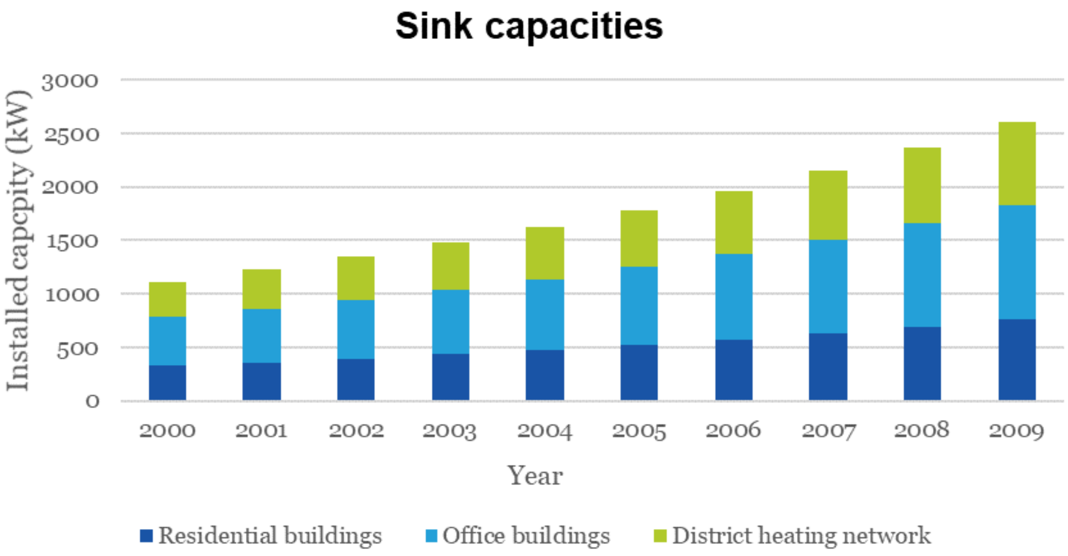


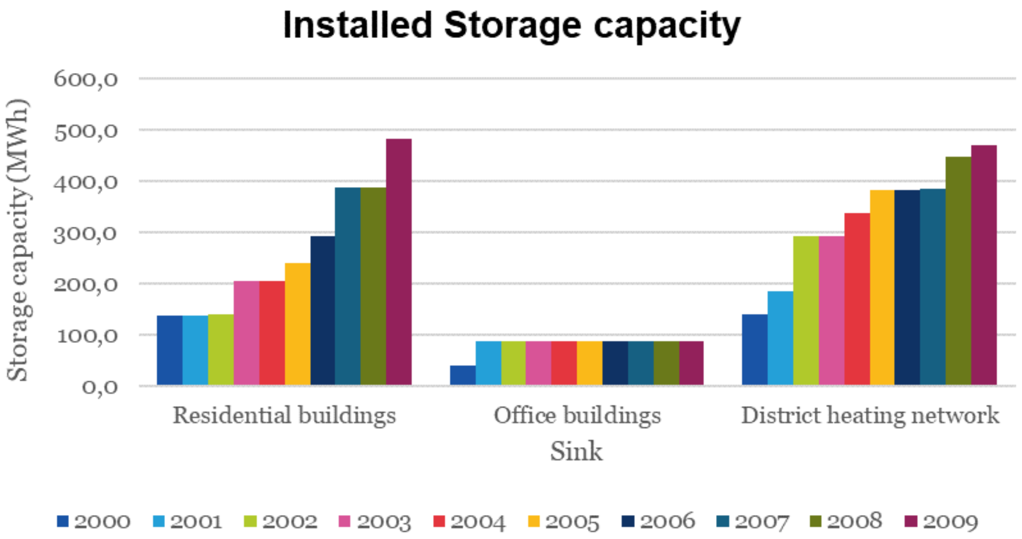
Source capacities

The installed capacities at the different sinks is shown in the Figure below. The sinks capacities are based on the sink demand. As the demand grow over the years, the capacity is also increased.

Sink capacities

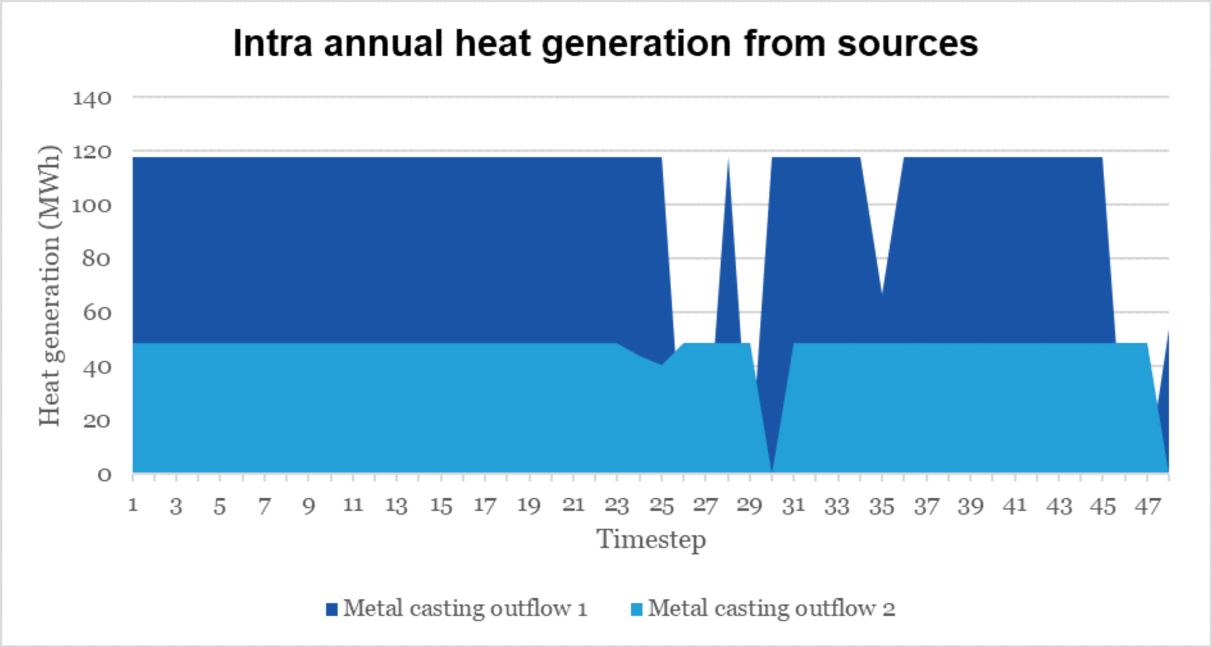
The installed capacities of storages is shown in the Figure below. The model only installs storages at the sink site since it is easier to control installed capacities in the DHN if the storage is located after the DHN. The storage capacities are also based on the sink demand. As the demand grow over the years, the storage capacity is also increased.


Sink capacities
Installed capcpity (kW)
3000
2500
2000
1500
1000
500
0
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
Year
Residential buildings
Office buildings
District heating network


Installed Storage capacity
Storage capacity (MWh)
600,0
500,0
400,0
300,0
200,0
100,0
0,0
Residential buildings
Office buildings
District heating network
Sink
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009

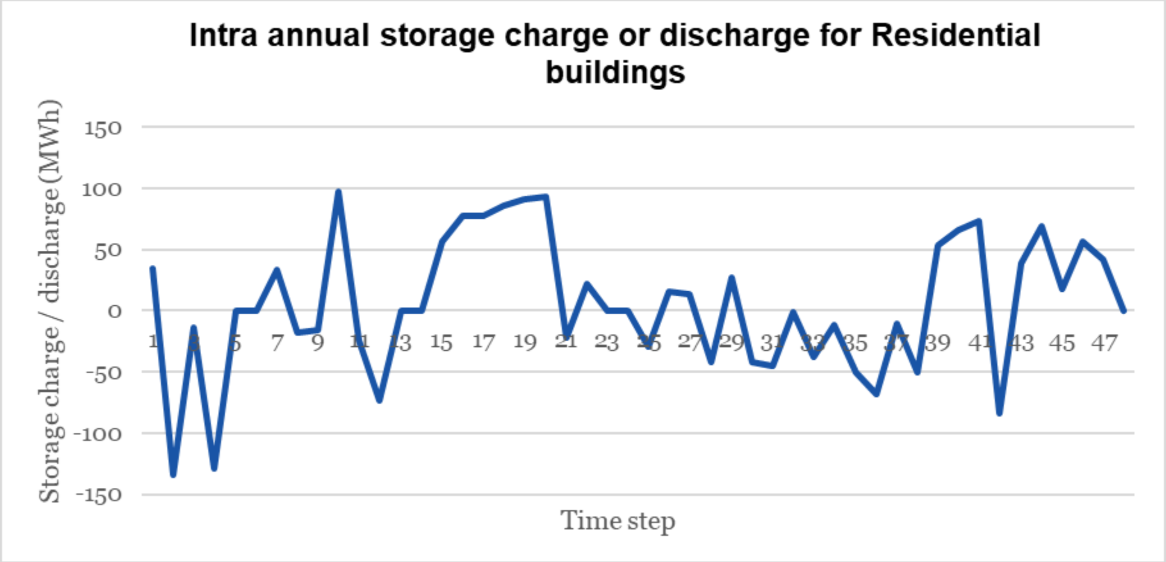
Installed capacity of storages

The Intra annual heat generation from sources is shown in the Figure below. The heat generation from the sources does not follow the demand profile due to the storages in the system. We see that the heat generation is constant in most time steps and there are drastic variations in a few TimeSlices. This is because of the charging and discharging from the storage which is seen next.



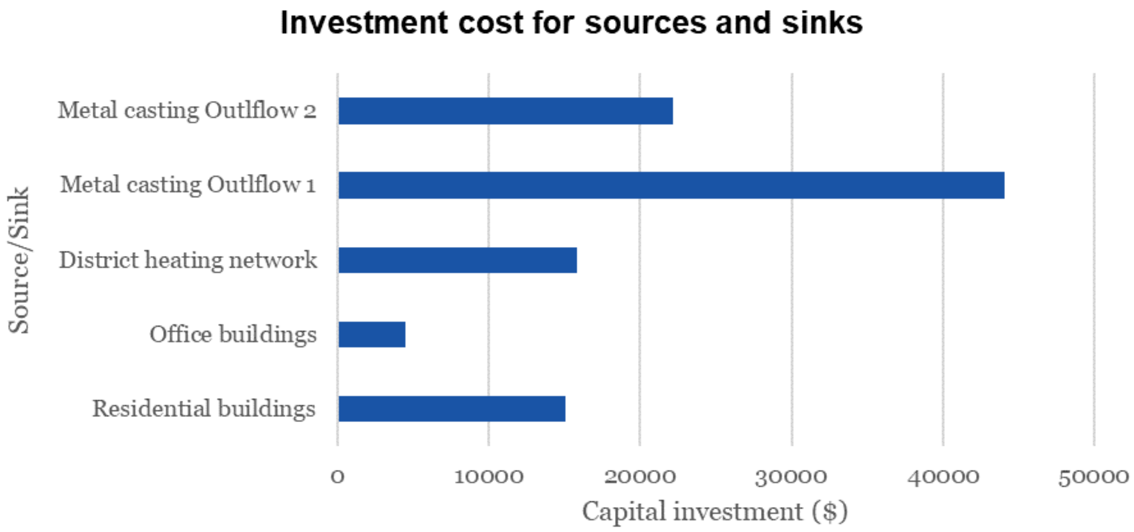
Intra annual heat generation from sources

The Intra annual storage charge or discharge for Residential buildings is shown in Figure below. The storage is continuously cycled according to the sink demand profiles.



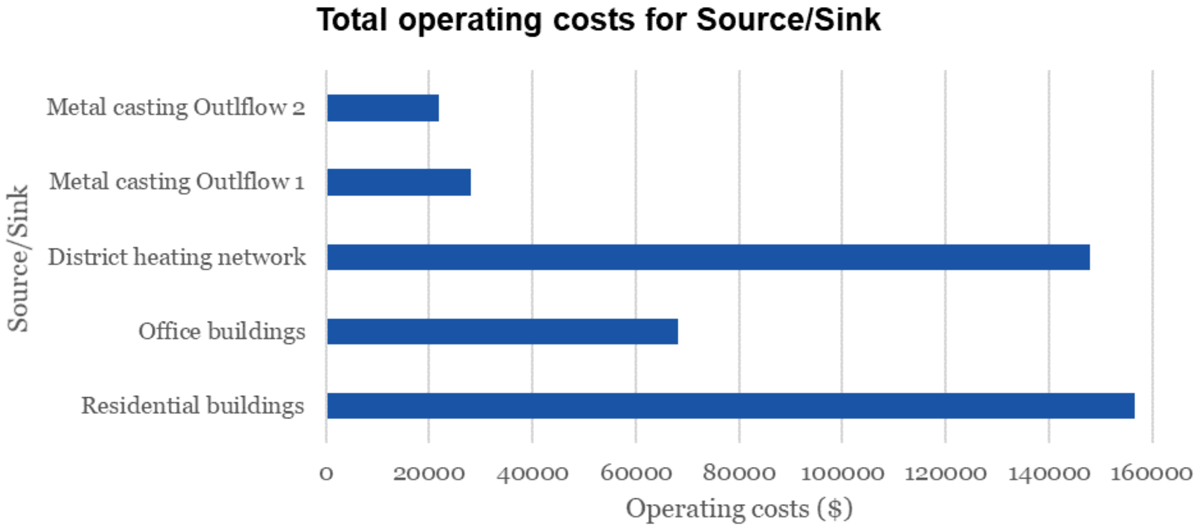
Intra annual storage operation

Annual costs (investment, fuel, operation & maintenance) associated with the technologies for the sources and the sinks is shown in the Figure below.



Investment Costs

The operation and maintenance costs of the sources and the sinks is shown in the Figure below.



Operating costs

1.10 Contributing to the TEO module

We are very grateful that you are looking into how you can contribute to TEO module of the EMB3RS project. The TEO Module is used to conduct a source sink matching for optimal recovery and use of industrial excess heat.

Contributing to the TEO module is open to everyone who is interested, and we adopt an inclusive and open policy which is described in our code of conduct

1.10.1 Some resources:

The main EMB3RS website is a good place to get started with EMB3RS You can find information on how to donwload the moduel and run it in the read me file on the repository or in the documentation.

The TEO is based on the open source energy system modeling tool OSeMOSYS The main OSeMOSYS website is a good place to get started with OSeMOSYS The forum is a great place to ask questions and search for answers from the knowledgeable community

1.10.2 Bugs

If you find a programming error in the TEO Module implementations, please submit an Issue in the repository. Follow the issue template for submitting a bug.

If you find a more fundamental issue which you think is related with the formulation of TEO please submit the issue at the repository and specify the .

Errors, typos or spelling mistakes in the documentation

1.10.3 Ideas and Suggestions

If you have a great idea for how the TEO could be improved, or to suggest a useful addition to the model, please submit a feature request.

1.10.4 Git Workflow

- To work with the TEO code bases, please follow the forking workflow recommended for contributing to open-source projects. The steps below assume you have a Github account.
- Fork the repository to which you wish to contribute by clicking the grey fork button or visiting https://github.com/EMB3RS-TEO-Module/fork
- Clone your fork of the repository git clone http://github.com/<user>/EMB3RS-TEO-Module
- Create a new branch on which you will commit your changes git checkout -b <branchname>
- Do the work and stage and commit your changes: git add . . . , git commit -m "A nice descriptive message"
- Push the changes to your fork git push -u <branchname> origin/<branchname>
- Submit a pull request from your fork of the repository to the master branch of the original repository.
- The pull request is reviewed. Any changes required by the review can be performed on the same branch and pushed to the forked repo as in the steps above.
- Once the pull request has been reviewed and accepted, you may delete your local copy of the branch git branch -d <branchname> and update your copy of the master branch git checkout master, git pull origin master

1.11 References

[1] M. Howells et al., "OSeMOSYS: The Open Source Energy Modeling System. An introduction to its ethos, structure and development.," Energy Policy, vol. 39, no. 10, pp. 5850–5870, 2011, doi: 10.1016/j.enpol.2011.06.033.

[2] D. Dreier and M. Howells, "OSeMOSYS-PuLP: A Stochastic Modeling Framework for Long-Term Energy Systems Modeling," Energies, vol. 12, no. 7, p. 1382, Apr. 2019, doi: 10.3390/en12071382.

[3] "Introduction to OSeMOSYS — OSeMOSYS 0.0.1 documentation." https://osemosys.readthedocs.io/en/latest/manual/Introduction.html (accessed Feb. 04, 2022).