Basic elements and equations of TIMES

Uwe Remme

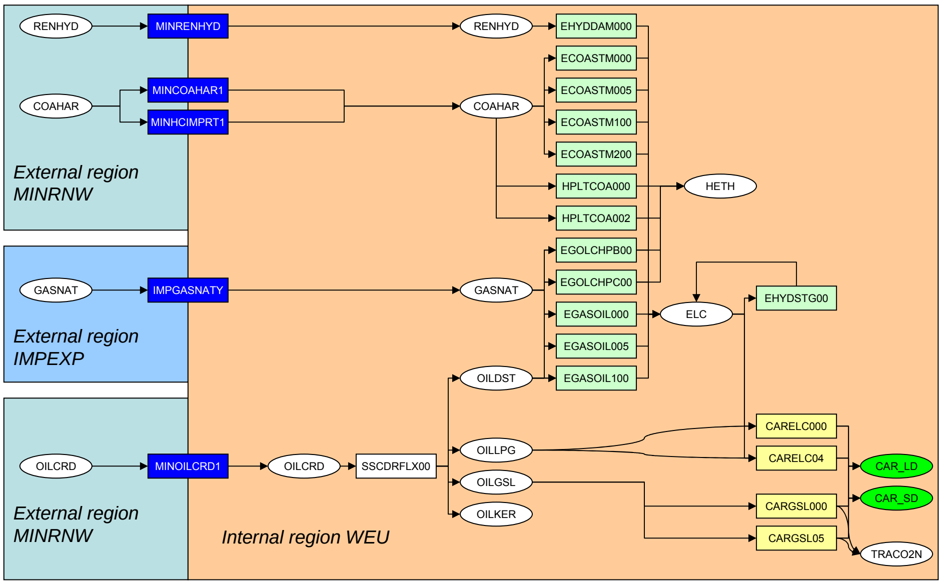
Institut für Energiewirtschaft und Rationelle Energieanwendung, Universität Stuttgart

ETSAP Workshop
28. November – 1. Dezember 2006 in Stuttgart

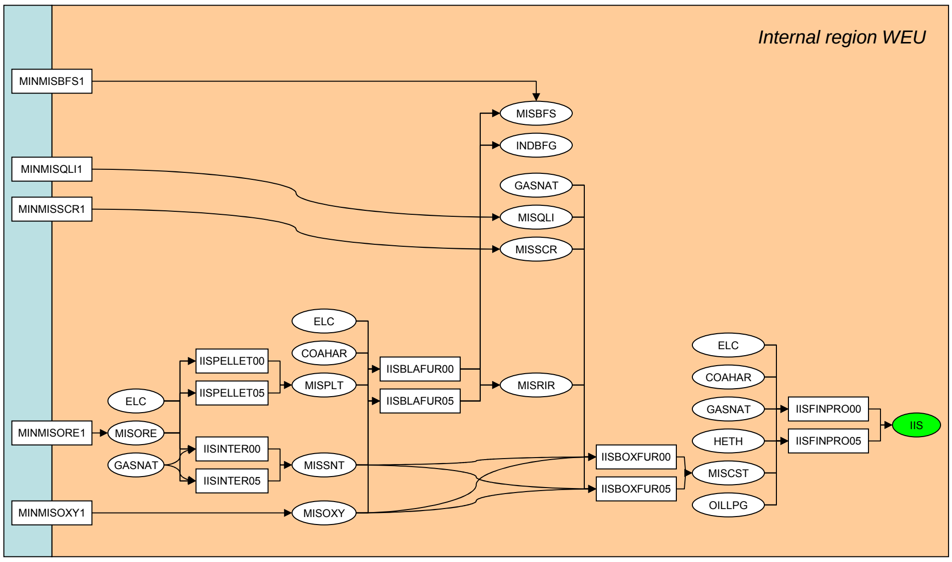
Overview

- Building the Reference Energy System (RES):
  i. Regions
  ii. Processes and Commodities
  iii. Time horizon
- Subannual time resolution: Time slices
- Variables
- Basic equations and related input data
- Objective function

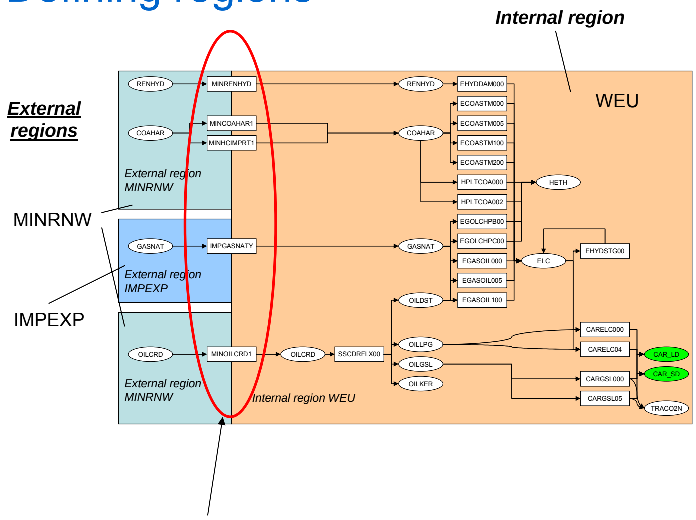
RES of WEU – Part 1



RES of WEU – Part 2



Defining regions



All regions

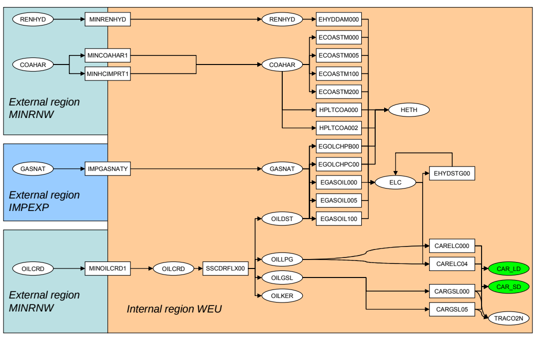
SET ALL_REG
/
WEU
IMPEXP
MINRNW
/;

Internal region

SET REG(ALL_REG)
/
WEU
/;

- External and internal regions are linked by exchange processes.

Defining commodities



Commodity groups

SET COM_GRP
/
COAHAR
OILCRD
OILLPG
OILGSL
OILKER
OILDST
GASNAT
RENHYD
ELC
HETH
…
/;

Commodities

SET COM(COM_GRP)
/
COAHAR
OILCRD
OILLPG
OILGSL
OILKER
OILDST
GASNAT
RENHYD
ELC
HETH
…
/;

- Each commodity is a commodity group with only one member: the commodity itself.
- Commodity unit must be specified, flows are measured in commodity units.
- Commodity type has to be specified: NRG (energy carrier), DEM (demand), ENV (environmental indicators), FIN (financial)

Defining processes

- Each process is member of the set PRC.
- Input and output commodities by entries in set TOP
- Primary commodity group (PCG):
  i. Group of commodity flows defining activity of a process
  ii. Activity unit has to be given.
  } ⇒Set PRC_ACTUNT (r,prc,cg,units_act)
- Capacity unit has to be specified.
- Commodity units defines flow units.
- Activity and commodity unit of PCG different ⇒ PRC_ACTFLO converting activity units to flow units of PCG
- Capacity unit and activity unit different ⇒ PRC_CAPACT converting capacity units to activity unit
- User is responsible for correct conversion factors!
- All commodity groups (CGs) related to the process ⇒ set PRC_CG

Defining processes

```
SET PRC
/
...
SSCCDRFLX00
...
/;
SET TOP (REG,PRC,COM,IO)
/
...
WEU.SSCCDRFLX00.OILCRD.IN
WEU.SSCCDRFLX00.OILDST.OUT
WEU.SSCCDRFLX00.OILLPG.OUT
...
/;
SET COM_GRP
/
...
SSCDRFLX00_NRGO
...
/;

SET COM_GMAP(REG,CG,COM)
/
...
WEU. SSCDRFLX00_NRGO.OILDST
WEU. SSCDRFLX00_NRGO.OILLPG
...
/;

SET PRC_CG(REG,PRC,CG)
/
...
WEU. SSCCDRFLX00.(
WEU. SSCCDRFLX00.(
...
/;
```

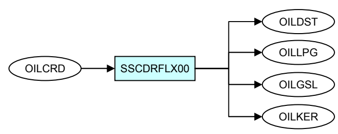
Process name

Topology

User defined commodity group

Members of commodity group

Commodity groups related to SSCCDRFLX00




```
SET PRC_CAPUNT(REG,PRC,CG,UNITS_CAP)
/
...
WEU.SSCCDRFLX00.SSCCDRFLX00_NRGO.PJA
...
/;

SET PRC_ACTUNT(REG,PRC,CG,UNITS_ACT)
/
...
WEU.SSCCDRFLX00.SSCCDRFLX00_NRGO.PJ
...
/;

PARAMETER PRC_ACTFLO(REG,ALLYEAR,PRC,CG)
/
/;

PARAMETER PRC_CAPACT(REG,PRC)
/
WEU.SSCCDRFLX00     1.0
/;
```

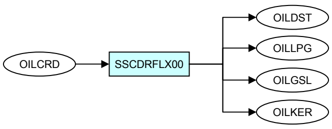
Capacity unit

PCG is SSCCDRFLX00 and activity unit PJ

No Conversion factor needed

Conversion factor is 1

Process definition in VEDA-FE



Activity = sum of all eNRGy output flows

Topology

Capacity unit: PJ/a
Activity unit: PJ

Efficiency from output flows NRGO to crude oil input

				~FI_ST: TCH, PRC, PRE						
				~FI_UT: TCAP=PJ/a; TACT=PJ						
Existing Flexible Refinery				~FI_T						
Sets	TechName	TechDesc	Comm-IN	Comm-OUT	Share~UP	EFF~NRGO	Life	BNDACT~UP	BNDACT~UP~2050	ENVACT
\D_TC: PCG=NRGO;	SSCDRFLX00	Flexible Refinery	OILCRD			1.05	50			
				OILLPG	0.5000					
				OILGSL	0.5000					
				OILKER	0.5000					
				OILDST	0.5000					

Defining processes contd.

```
SET PRC
/
...
CARGSL00
...
/;
SET TOP (REG,PRC,COM,IO)
/
...
WEU.CARGSL00.OILGSL.IN
WEU.CARGSL00.CAR_LD.OUT
WEU.CARGSL00.CAR_SD.OUT
/;

SET PRC_CG(REG,PRC,CG)
/
...
WEU.CARGSL00.CARGSL00_DEMO
...
/;

SET PRC_CAPUNT(REG,PRC,CG,UNITS_CAP)
/
...
WEU.CARGSL00.CARGSL00_DEMO. 000units
...
/;

SET PRC_ACTUNT(REG,PRC,CG,UNITS_ACT)
/
...
WEU.CARGSL00.CARGSL_DEMO.MVKMS
...
/;
```

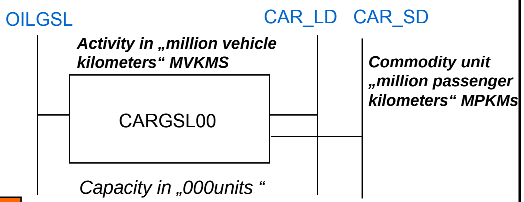
Process name

Topology

Commodity groups related to CAR

Capacity unit

PCG = total demand output and activity unit in MVKMS



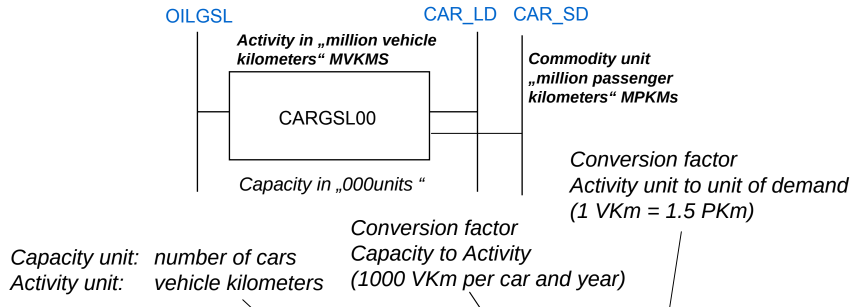

```
PARAMETER PRC_ACTFLO(REG,ALLYEAR,PRC,CG)
/
WEU.1990.CARGSL00.CARGSL00_DEMO
/;

PARAMETER PRC_CAPACT(REG,PRC)
/
WEU.CARGSL00      0.001
/;
```

Persons per car

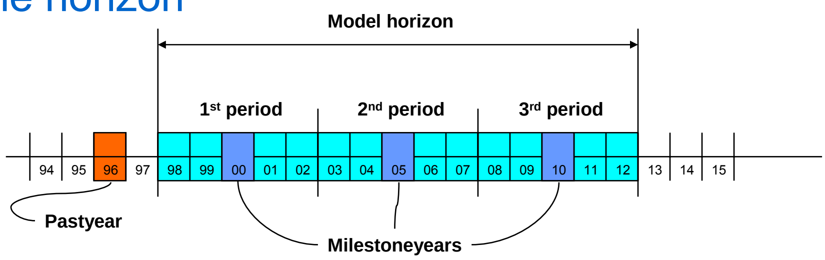
VKM/car = 1000 VKM/car = 1 MVKM/1000cars

Process definition in VEDA-FE



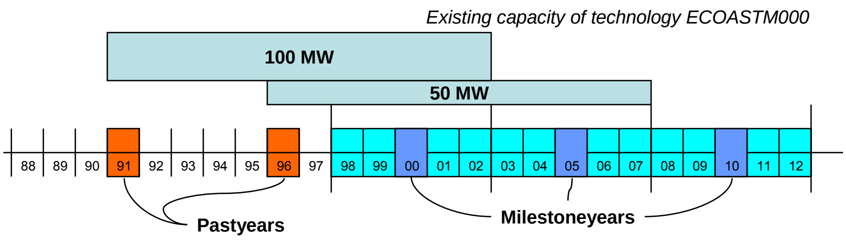
	~FI_Region: WEU,ROW										
				~FI_ST: TCH, DMD				~FI_TC: PRC_VINT=Yes			
				~FI_UT: TCAP=000 units; TACT=MVKms							
Existing Cars				~FI_T							
TechName	TechDesc	Comm-IN	Comm-IN-A	Comm-OUT	CEFF	Cons	Cap2Act	ACTFLO~DEMO	Availability	Life	Stock
				\I:	MKms/PJ		stock/demand	Passenger/Car	Max Ann Km	Years	'000 Units
CARGSL00	Gasoline Car - Dual mode	OILGSL					0.001	1.5	20000	20	**23.0**
				CAR_LD	1.2						
				CAR_SD	0.9						

Time horizon



- Different period durations are possible
- Different type of years:
  i. MILESTONYR
  ii. PASTYEAR
  iii. MODLYEAR = MILESTONYR + PASTYEAR
  iv. DATAYEAR: years with input data, input data are inter-/extrapolated to milestoneyears

Past investments



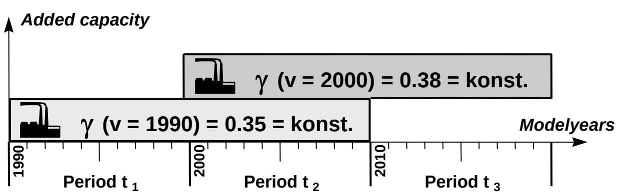
- Specification of existing capacity by past investments in their **vintage/past years** (NCAP_PASTI(r,t,prc)):

```
PARAMETER NCAP_PASTI(REG,ALLYEAR,PRC)
/         WEU.1991.ECOASTM000   100
          WEU.1996.ECOASTM000    50  /
```

- Alternative specification of residual curve for **Milestoneyears** PRC_RESID(r,t,prc)

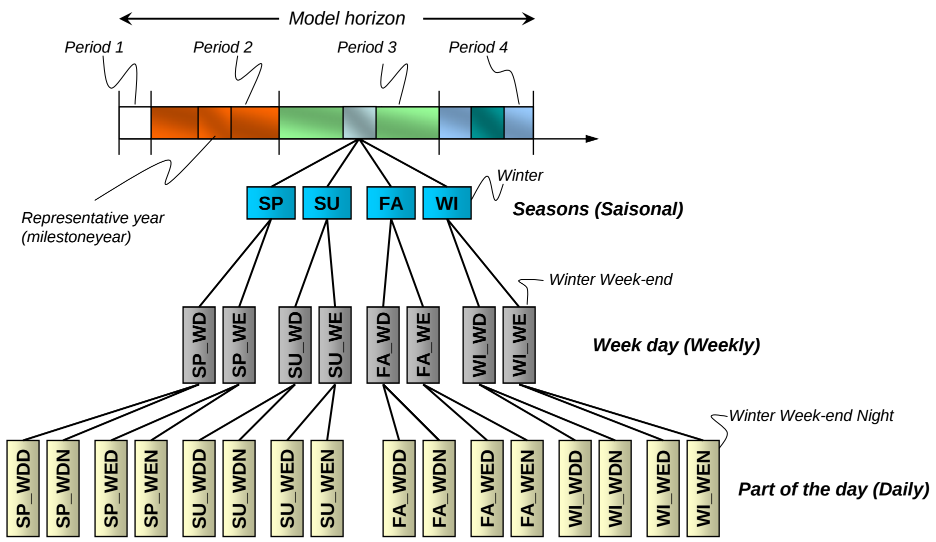
```
PARAMETER PRC_RESID(REG,ALLYEAR,PRC)
/         WEU.2000.ECOASTM000   150
          WEU.2005.ECOASTM000    50  /
```


Vintaging



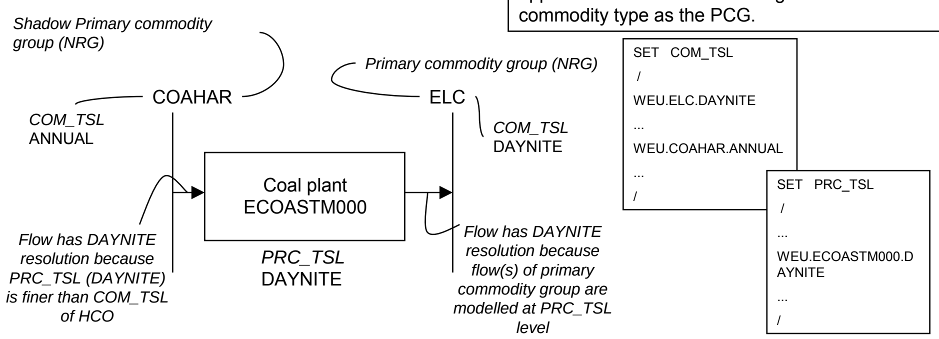
- Process can be specified as vintaged one by entry in set PRC_VINT.
- The characteristics of a vintaged process can be distinguished by its vintage year, e.g. process flow variables have as additional index of the current period *t* the construction period *v*: *VAR_FLO(r,v,t,p,c,s)*

Timeslices



Timeslices contd.

SPG(Shadow Primary Commodity Group) is defined as the group of commodities on the opposite side of the PCG having the same commodity type as the PCG.



- Commodity timeslice level COM_TSL ⇒ timeslice level of commodity balance equation (default ANNUAL)
- Process timeslice level PRC_TSL ⇒ timeslice level of activity variable and corresponding flow variables
- All other flows are modelled at the finest level of COM_TSLevel of the SPG or PRC_TSL.
- Possible to define individual timeslices for commodities (COM_TS) and process (PRC_TS) defining the timeslices for which the commodity or process is available.

Model matrix

Variables

Equation sign

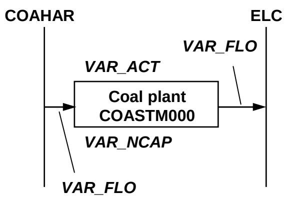
Equations

	ACT	CAP	COMNET	FLO	NCAP	OBJINV	OBJFIX	OBJVAR	OBJSAL	OBJz		RHS
EQ_OBJ						+	+	+	-	-	E	0
EQ_OBJFIX					+		-				E	-
EQ_OBJINV					+	-					E	-
EQ_OBJSALV					+				-		E	-
EQ_OBJVAR	m			+				-			E	0
EQ_ACTFLO	+			-							E	0
EQL_ACTBND	+										L	+
EQL_CAPACT	+				-						L	+
EQE_CAPACT	+				-						E	+
EQG_COMBAL	m		+	m							G	+
EQE_COMBAL	m		m	+							E	0
EQE_CPT		+			-						E	+
EQG_CPT					-						G	-
EQL_CPT					-						L	-
EQE_INSHR				m							E	0
EQ_IRE	m										E	0
EQG_OUTSHR				m							G	0
EQE_OUTSHR				m							E	0
EQ_PTRANS	+			m							E	0
EQG_UCRT	m			+							G	0
EQL_UCR	+										L	+
EQL_UCRT	m			+							L	+
EQE_ACTEFF	-			+							E	0
Variable Typ	+	+	+	+	+	+	+	+	+	u		

Variables

Associated bound parameters

- Process oriented: *region* *vintage period* *actual period* *process* *timeslice* *lo,up or fx*
  1. VAR_ACT(r,v,t,p,s): activity of a process — ACT_BND(r,t,p,s,l)
  2. VAR_CAP(r,v,p): installed capacity = previous investments + new investments + past investments still existing — CAP_BND(r,v,p,l)
  3. VAR_NCAP(r,v,p): new investment in period v — NCAP_BND(r,v,p,l)



- Flow oriented:
  1. VAR_FLO(r,v,t,p,c,s): flow level of commodity c linked to process p
  2. VAR_IRE(all_reg,v,t,p,c,s,ie): inter-regional exchange variable (*imp or exp*) — IRE_BND(r,t,c,s,all_reg,ie,l) / IRE_XBND(all_reg,t,c,s,ie,l)
  3. VAR_SIN / VAR_SOUT(r,v,t,p,c,s): flows entering/leaving a process p storing a commodity c — STG_IN/OUTBND(r,t,c,s,all_reg,ie,l)

Variables contd.

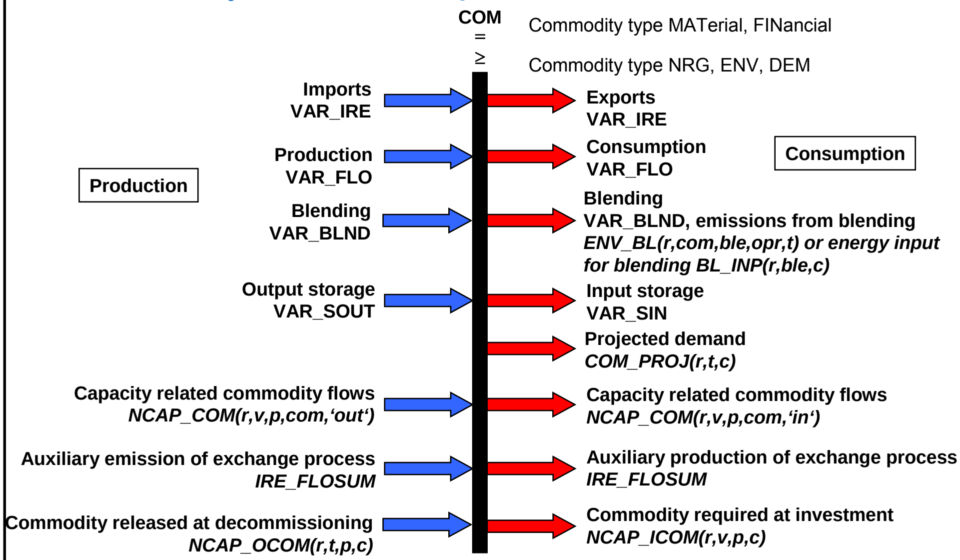
Associated bound parameters

- Commodity oriented (only created if bound provided):
  i. VAR_COMPRD(r,t,c,s): total production of a commodity — COM_BNDPRD(r,t,c,s,l)
  ii. VAR_COMCON(r,t,c,s): total consumption of a commodity — COM_BNDCON(r,t,c,s,l)
  iii. VAR_COMNET(r,t,c,s): net level of a commodity (production – consumption) — COM_BNDNET(r,t,c,s,l)
- Blending variables
  i. VAR_BLND(r,t,ble,opr): amount of blending stock opr needed for the production of blending product ble

Basic equations

EQ(l)_COMBAL $_{r,t,c,s}$	Commodity balance
EQ_ACTFLO $_{r,v,t,p,s}$	Definition of activity variable
EQ_CAPACT $_{r,v,t,p,s}$	Utilization constraint
EQ_PTRANS $_{r,v,t,p,cg1,cg2,s}$	Transformation equation
EQ(l)_INSHR/OUTSHR $_{r,t,p,c,cg,s}$	Share constraints on in/output side of process
EQ_OBJ	Objective function

Commodity balance equation



Commodity balance equation contd.

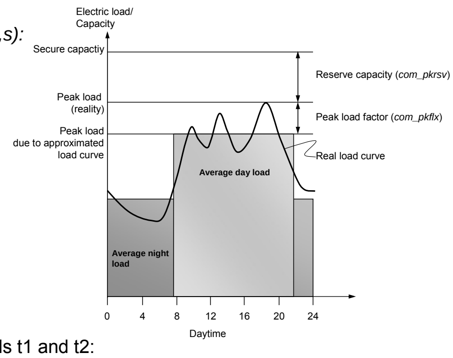
- Commodity balance is created for timeslices on timeslice level specified by *COM_TSL* or individual timeslices given by *COM_TS* (note: then commodity is only available in *COM_TS* timeslices)
- Commodity efficiency *COM_IE*: $Commodity\ production \times COM\_IE \geq Commodity\ consumption$
- Annual demand given by *COM_PROJ(r,t,c)*
- Load curve of demand described by *COM_FR(r,t,c,s):*

  PARAMETER COM_FR

  /

WEU.2000.RH.ID	0.12000000
WEU.2000.RH.IN	0.06000000
WEU.2000.RH.SD	0
WEU.2000.RH.SN	0
WEU.2000.RH.WD	0.54670000
WEU.2000.RH.WN	0.27330000

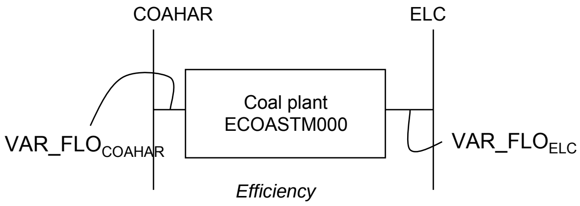
  /



- Cumulative commodity bounds between two periods t1 and t2:
  - *COM_CUMNET(r,t1,t2,c,l)* limit on net amount of commodity
  - *COM_CUMPRD (r,t1,t2,c,l)* limit on production of commodity

Transformation equation

- Transformation equations establishes relationship between the flows of two commodity groups.
- Example 1: Efficiency of coal plant ECOASTM000



Efficiency

$$FLO\_FUNC_{r,t,ECOASTM000,COAHAR,ELC,s} = 0.33$$

Transformation equation

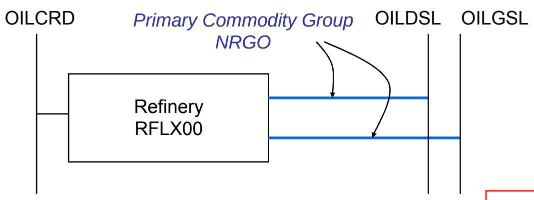
$$EQ\_PTRANS_{r,v,t,ECOASTM000,COAHAR,ELC,s}$$

$$FLO\_FUNC_{r,t,ECOASTM000,COAHAR,ELC,s} \times VAR\_FLO_{r,v,t,ECOASTM000,COAHAR,s} = VAR\_FLO_{r,v,t,ECOASTM000,ELC,s}$$

Process — *1st commodity group* — *2nd commodity group*

Transformation equation contd.

- Example 2: Simple refinery RFLX00



Efficiency $FLO\_FUNC_{r,t,RFLX00,NRGO,OIL,s} = 1$

1st commodity is now on the output side: FLO_FUNC appears always on the side of the process indicated by the first commodity group!!

Transformation equation

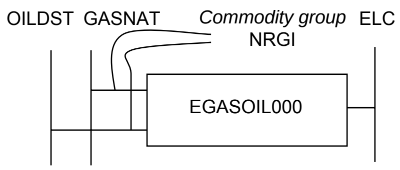
$$EQ\_PTRANS_{r,v,t,RFLX00,NRGO,OILCRD,s}$$

$$FLO\_FUNC_{r,t,RFLX00,NRGO,OILCRD,s} \times \left(VAR\_FLO_{r,v,t,RFLX00,OILDSL,s} + VAR\_FLO_{r,v,t,RFLX00,OILGSL,s}\right) = VAR\_FLO_{r,v,t,RFLX00,OILCRD,s}$$

Process — *1st commodity group* — *2nd commodity group*

Transformation equation contd.

- Example 3: Oil/Gas power plant



Overall Efficiency

$$FLO\_FUNC_{r,t,EGASOIL000,NRGI,ELC,s} = 1$$

Fuel-dependent Efficiency

$$FLO\_SUM_{r,t,EGASOIL000,NRGI,OILDST,ELC,s} = 0.35$$

$$FLO\_SUM_{r,t,EGASOIL000,NRGI,GASNAT,ELC,s} = 0.32$$

Process · *1st commodity group* · *Member of 1st commodity group* · *2nd commodity group*

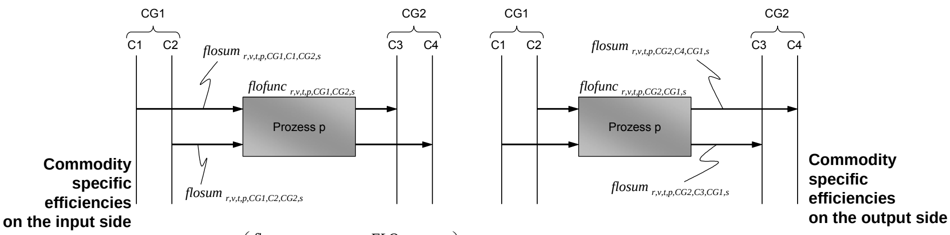
Transformation equation

$EQ\_PTRANS_{r,v,t,EGASOIL000,NRGI,ELC,s}$

$$FLO\_FUNC_{r,t,EGASOIL000,NRGI,ELC,s} \times \begin{pmatrix} VAR\_FLO_{r,v,t,EGASOIL000,ELC,s} \times FLO\_SUM_{r,t,EGASOIL000,NRGI,OILDST,ELC,s} \\ + VAR\_FLO_{r,v,t,EGASOIL000,ELC,s} \times FLO\_SUM_{r,t,EGASOIL000,NRGI,GASNAT,ELC,s} \end{pmatrix} = VAR\_FLO_{r,v,t,EGASOIL000,ELC,s}$$

Transformation equation contd.

- Two possible cases:



$$flofunc_{r,v,t,p,CG1,CG2,s} \begin{pmatrix} flosum_{r,v,t,p,CG1,C1,CG2,s} FLO_{r,v,t,p,C1,s} \\ + flosum_{r,v,t,p,CG1,C2,CG2,s} FLO_{r,v,t,p,C2,s} \end{pmatrix} = FLO_{r,v,t,p,C3,s} + FLO_{r,v,t,p,C4,s}$$

$$FLO_{r,v,t,p,C1,s} + FLO_{r,v,t,p,C2,s} = flofunc_{r,v,t,p,CG2,CG1,s} \begin{pmatrix} flosum_{r,v,t,p,CG1,C3,CG2,s} FLO_{r,v,t,p,C3,s} \\ + flosum_{r,v,t,p,CG1,C4,CG2,s} FLO_{r,v,t,p,C4,s} \end{pmatrix}$$

- It is not possible to use commodity specific efficiencies on both sides of the process at the same time!
- *EQ_PTRANS* is created on the finer timeslice level of SPG or PRC_TSL. SPG (Shadow Primary Commodity Group) is defined as the group of commodities on the opposite side of the PCG having the same commodity type as the PCG.

Efficiency parameters in VEDA-FE

Commodity consumption on input side per unit of activity

Commodity efficiency

Efficiency from SPG to PCG

Existing electricity power plants				~FI_T							
Sets	TechName	TechDesc	Comm-IN	Comm-OUT	Consumption	Input	Share~LO	CEfficiency	CEFF~COAHAR	EFF	S_EFF
				\I:							
	ECOASTM000	EPLT.COA.Steam Turbine.Existing.	COAHAR		3.00						
				ELC							
	EGASOIL000	EPLT.Dual Fuel.Existing.	OILDST				0.20	0.40			
			GASNAT					0.45			
				ELC							
	EHYDDAM000	EPLT.HYD.Dam.Seasonal Reservoir.	RENHYD								
				ELC							
\D_ST: ELE,STGTSS	EHYDSTG00	EPLT: Hydro Storage Dam	ELC	ELC							0.80
				\I:							
	ECOASTM100	EPLT.COA.Steam Turbine.Var 1.	COAHAR	ELC					0.33		
				\I:							
	ECOASTM200	EPLT.COA.Steam Turbine.Var 2.	COAHAR	ELC						0.33	
				\I:							
	EGASOIL100	EPLT.Dual Fuel.Existing.Var 1.	OILDST			3.00	0.20				
			GASNAT			2.00					
				ELC							

Commodity-specific efficiency efficiency

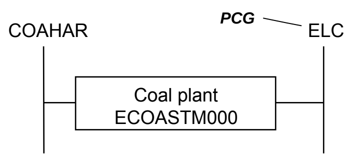
				~FI_ST: TCH, DMD			
				~FI_UT: TCAP=000 units; TACT=MVKms			
Existing Cars				~FI_T			
TechName	TechDesc	Comm-IN	Comm-IN-A	Comm-OUT	CEFF	Cons	Cap2Act
				\I:	MKms/PJ		stock/demand
CARGSL00	Gasoline Car - Dual mode	OILGSL					0.001
				CAR_LD	1.2		
				CAR_SD	0.9		

Process transformation parameter VEDA_FLOP

- Syntax: *region*, *actual period (datayear)*, *process*, *commodity group*, *timeslice*

  VDA_FLOP(r, t, prc, cg, s)

- Example 1: Efficiency of coal plant ECOASTM000



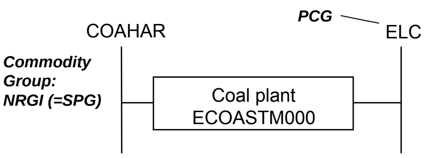
- Primary commodity group (PCG) = Definition of activity (PRC_ACTUNT)
  here: Electricity ELC chosen
- Efficiency of 33%:

$$VDA\_FLOP_{r,t,ECOASTM000,COAHAR,s} = 3$$

- *VDA_FLOP* defined for individual commodities c (input or output side):
  Flow of commodity c per unit of activity (here 3 units coal per 1 unit of electricity)
- Rigid processes can be defined in a similar manner, i.e. relating all commodity flows to the activity by individual *VDA_FLOP*s.

VEDA_FLOP contd.

- Example 2: Efficiency of coal plant ECOASTM000



- Primary commodity group (PCG) = Definition of activity (PRC_ACTUNT)
  here: Electricity ELC chosen
- Shadow primary group (SPG): all commodities of type NRG on the opposite side of the PCG
  here: COAHAR commodity
- Efficiency of 33%: $VDA\_FLOP_{r,t,ECOASTM000,COAHAR,s} = 0.33$

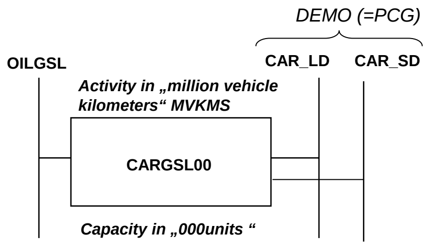
$$VDA\_FLOP_{r,t,ECOASTM000,NRGI,s} = 1$$

- VDA_FLOP for the shadow primary commodity group (SPG) NGRI of 1 needed to indicate that transformation equation works from the input to the output side
  (actually describes: activity units per one unit of SPG; here: one unit of electricity per unit of coal; corresponds to FLO_FUNC)
- VDA_FLOP for coal commodity defines the actual efficiency of 33% (corresponds to FLO_SUM).

$$VDA\_FLOP_{r,t,ECOASTM000,NRGI,s} \times VDA\_FLOP_{r,t,ECOASTM000,COAHAR,s} \times VAR\_FLO_{r,v,t,ECOASTM000,COAHAR,s}$$
$$= VAR\_FLO_{r,v,t,ECOASTM000,ELC,s}$$

VEDA_FLOP contd.

- Example 3: Dual mode car



- Primary commodity group (PCG) = Definition of activity (PRC_ACTUNT)
  here: group of CAR_LD and CAR_SD; called DEMO
- SPG: here OILGSL
- Modal efficiencies:

$$VDA\_FLOP_{r,t,CARGSL,CAR\_LD,s} = 0.8$$

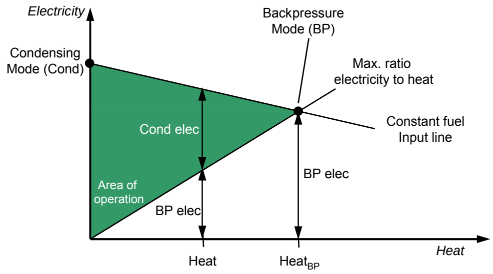
$$VDA\_FLOP_{r,t,CARGSL,CAR\_SD,s} = 1.11$$

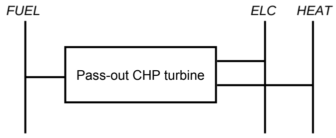
- Overall efficiency from PCG to SPG(optional):

$$VDA\_FLOP_{r,t,CARGSL,DEMO,s} = 1$$

$$VAR\_FLO_{r,v,t,CARGSL00,OILGSL,s} = VDA\_FLOP_{r,t,CARGSL,DEMO,s} \times \begin{pmatrix} VDA\_FLOP_{r,t,CARGSL,CAR\_LD,s} \times VAR\_FLO_{r,v,t,CARGSL,CAR\_LD,s} + \\ VDA\_FLOP_{r,t,CARGSL,CAR\_SD,s} \times VAR\_FLO_{r,v,t,CARGSL,CAR\_SD,s} \end{pmatrix}$$

CHP plants





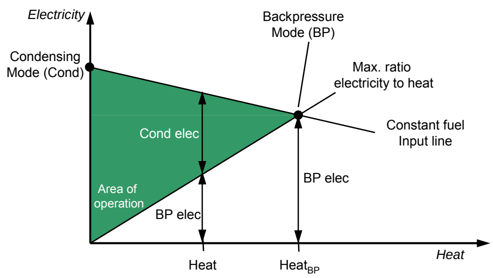
Description of constant fuel input line

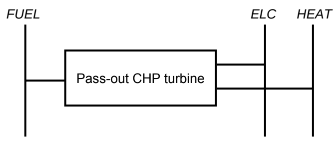
$$VAR\_FLO_{fuel} = \frac{1}{\eta_{cond}}\left(VAR\_FLO_{ELC} + elp \cdot VAR\_FLO_{HEAT}\right)$$

$$reh \geq \frac{VAR\_FLO_{ELC}}{VAR\_FLO_{HEAT}}$$

η_{cond} : Condensing mode efficiency
reh : Max. Ratio of electricity to heat
elp : Electricity loss per heat unit

CHP plants – Input parameters

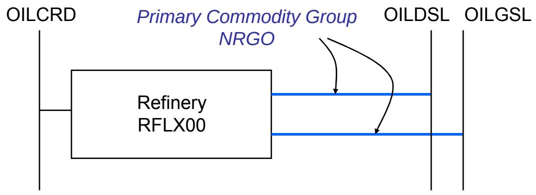




- Pass-out or extraction turbine:
  - i. Parameter VDA_CEH(r,t,prc):
    - a) <= 1: electricity loss per unit of heat gained (moving from condensing to backpressure mode; indicates activity is measured in terms of electricity
    - b) >= 1: heat loss per unit of electricity gained (moving from backpressure to condensing mode); indicates activity is measured in terms of total ouptut (electricity plus heat)
  - ii. Efficiencies:
    Case a): efficiencies must be specified for condensing point; costs must be specified based on condensing mode (e.g. investment costs)
    Case b): efficiencies must be specified for backpressure point; costs must be specified based on total electricity and heat output at backpressure point (e.g. investment costs)
  - iii. NCAP_CHPR(r,t,prc,bd): Ratio of heat to power at backpressure point; at least a maximum value is required, but in addition also a minimu value may be specified

Definition of activity variable

- Activity of a process equals the sum of the flows specified in the Primary Commodity Group (PRC_ACTUNT).
- Activity variable is created on the timeslice level specified by PRC_TSL.



$$EQ\_ACTFLO_{r,v,t,RFLX00,s}$$
$$=$$
$$VAR\_ACT_{r,v,t,RFLX00,s} = VAR\_FLO_{r,v,t,RFLX00,OILDSL,s} + VAR\_FLO_{r,v,t,RFLX00,OILGSL,s}$$

Universität Stuttgart
Institut für Energiewirtschaft und Rationelle Energieanwendung IER

Capacity utilization constraint

EQ(l)_CAPACT $_{r,v,t,p,s}$

VAR_ACT(r,v,t,p,s) ≤ = Available capacity of process p in period t and timeslice s

timeslice s member of PRC_TSL

Capacity:

- Past investments before model horizon: NCAP_PASTI
- New investments from previous periods
- New investments in actual period t

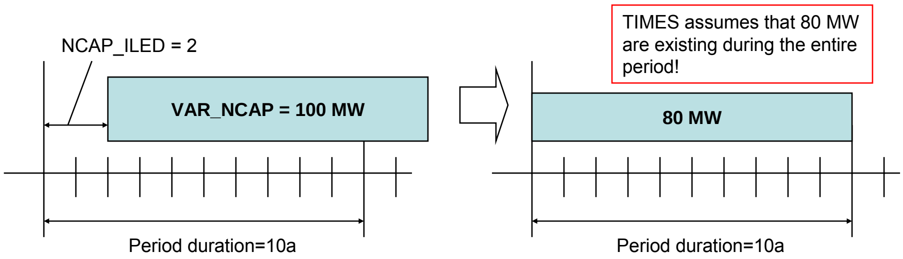
Three availabilities:

- NCAP_AFA(r,v,p,l) : Annual availability
- NCAP_AFS(r,v,p,s,l) : Seasonal availability
- NCAP_AF(r,v,p,s,l) : Availability in timeslices

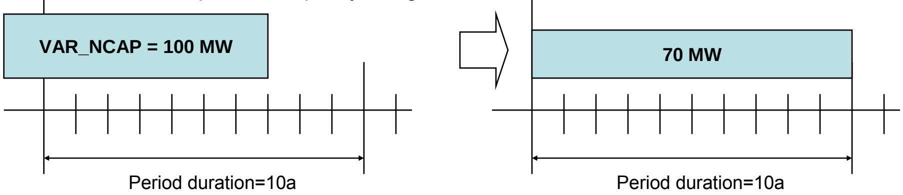
which can be combined.

Capacity utilization constraint contd.

- Effective capacity of a new investment with construction time in construction period:

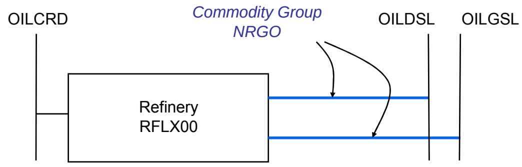


- Similar assumption for capacity being decommissioned:



Share constraints on the input/output of a process

- Possibility to limit the share of a commodity flow within a commodity group on the input or output side of a process.
- Example



$$FLO\_SHAR_{r,t,FRLX00,NRGO,OILDSL,s,UP} = 0.3$$

Process · *Commodity group defining the total flow* · *Commodity*

- Fixed, upper or lower bounds may be specified.
- Commodity group must not necessarily compromise all output/input flows, one can identify a subgroup as commodity group.

EQ(l)_IN/OUTSHR

$$\frac{\sum_{v} VAR\_FLO_{r,v,t,RFLX00,OILDSL,s}}{\sum_{v} \left(VAR\_FLO_{r,v,t,RFLX00,OILDSL,s} + VAR\_FLO_{r,v,t,RFLX00,OILGSL,s} \right)} \leq FLO\_SHAR_{r,t,RFLX00,NRGO,OILDSL,s,UP}$$

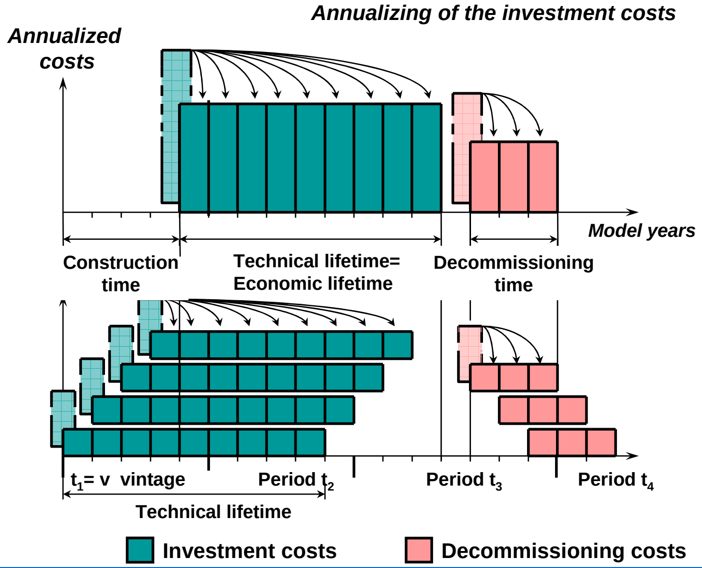
Objective function

- Discounted sum of the annual costs minus revenues:
  - \+ Investment costs
  - \+ Costs for sunk material during construction time
  - \+ Variable costs
  - \+ Fix operating and maintenance costs
  - \+ Surveillance costs
  - \+ Decommissioning costs
  - \+ Taxes

  - \- Subsidies
  - \- Recuperation of sunk material
  - \- Salvage value

- Cost documentation of investments made before the first model year
- Technical and economic lifetime
- Investment and decommissioning lead-times
- General and technology specific discount rate

Objective function



Further model equations

- Description of exchange processes in multi-regional models
- Elastic demands
- Product/market share constraints, e.g. share of hydrogen cars in total person kilometer demand
- Peaking equation: Ensures enough available secure capacity during peak demand
- Storage equation: Modeling of storage between timeslices (e.g. pump storage) or between periods (e.g. stockpiling)
- Commodity-specific availabilities, e.g. full load hours of CHP plant in backpressure and condensing mode
- User constraints: Flexible framework to formulate constraints being not part of the standard portfolio of TIMES equations, e.g. growth constraints, renewable quota

Universität Stuttgart
Institut für Energiewirtschaft und Rationelle Energieanwendung IER

Further TIMES features

- Lumpy investments
- Endogenous technological learning (MIP formulation)
- Linkage with a one-sectoral macroeconomic model
- Climate module
- Multi-stage stochastic programming

Improvements over MARKAL

- Data decoupling
- Flexible period duration
- Timeslices within a year:
  - Flexibility in number of timeslices
  - Free choice of processes or commodities modeled on subannual timeslice levels
- Process description:
  - Three basic processes: standard/generic process, storage process and exchange process
  - Flexible process formulation in terms of input and output flows
- Investment and dismantling lead-times and costs
- Vintaged processes and age-dependent parameters

Objective function

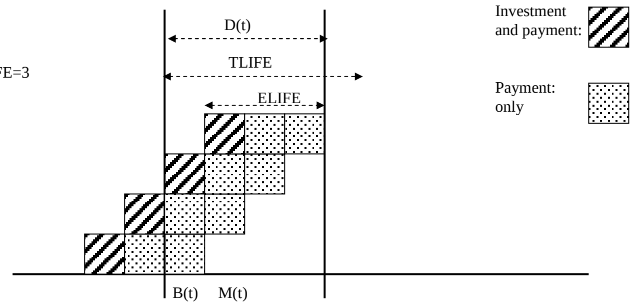
- Four cases considered four investment decisions to make the timing of payments and revenues more realistic:
  - Small vs. large projects (1 and 2)
  - Techncal lifetime greater or smaller than period length (a and b)
  - Cases 1.a, 1.b, 2.a, 2.b
- These four cases are used for:
  - Investment costs ⇒ Cases I.1.a, I.1.b, I.2.a, I.2.b
  - Taxes, Subsidies on investments ⇒ Cases II.1.a, II.1.b, II.2.a, II.2.b
  - Decommissioning costs ⇒ Cases III.1.a, III.1.b, III.2.a, III.2.b
  - Fixed annual costs (FOM and surveillance) ⇒ Cases IV.1.a, IV.1.b, IV.2.a, IV.2.b
  - Taxes, Subsidies on capacity ⇒ Cases V.1.a, V.1.b, V.2.a, V.2.b
  - Salvaging ⇒ Cases IX.1.a, IX.1.b, IX.2.a, IX.2.b
- Investments costs, decommissioning costs, surveillance costs are considered also after end of model horizon ⇒ Salvaging of these cost components needed

Objective function - Investment

Small, divisible projects, unrepeated investment in period

Example I.1.a:

D(t)=4, TLIFE=5,ELIFE=3
M(t)=B(t)+1

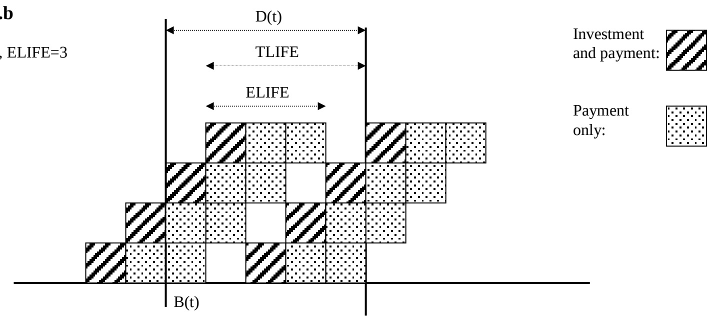


Objective function - Investment

Small projects, repeated investment in period

Example I.1.b

D=5, TLIFE=4, ELIFE=3

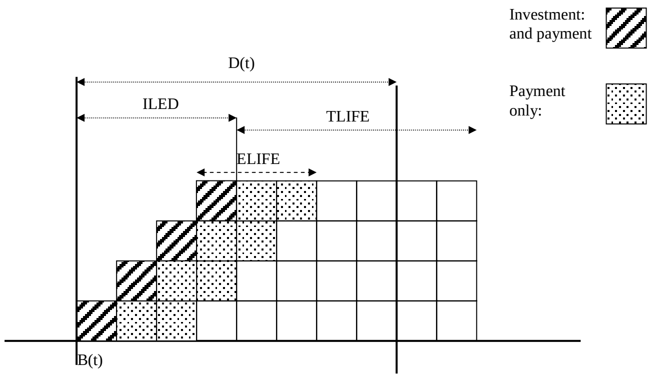


Objective function - Investment

Large, indivisible projects, unrepeated investment in period

Example I.2.a:

D(t)=8, ILED=4
TLIFE=6,ELIFE=3



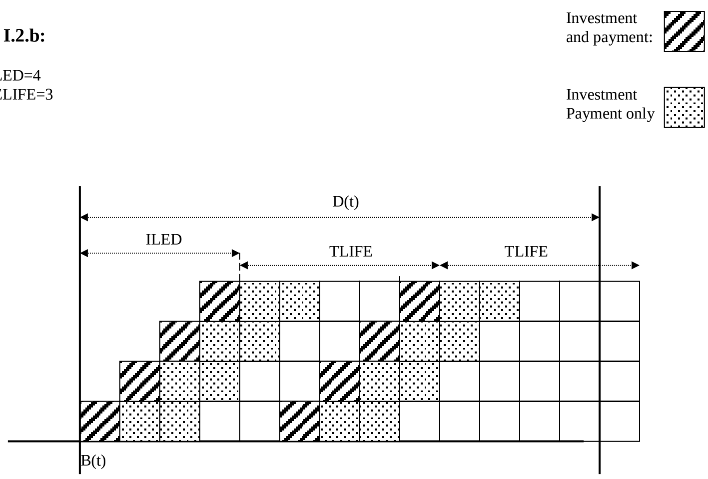
Objective function - Investment

Large, indivisible projects, repeated investment in period

Example I.2.b:

D(t)=13, ILED=4
TLIFE=5,ELIFE=3

C=2



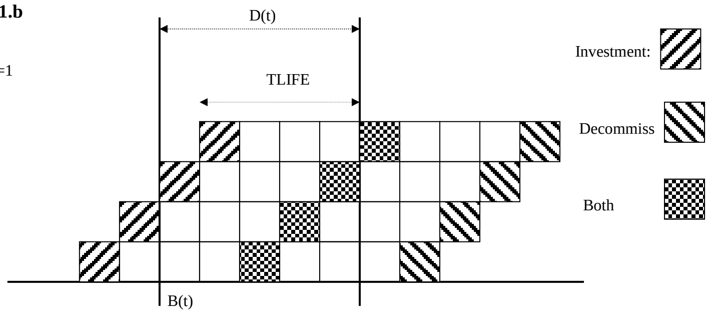
Objective function - Decommissioning

Small projects, repeated investment in period

Example III.1.b

D=5, TLIFE=4
DLIFE=DELIF=1

C=2

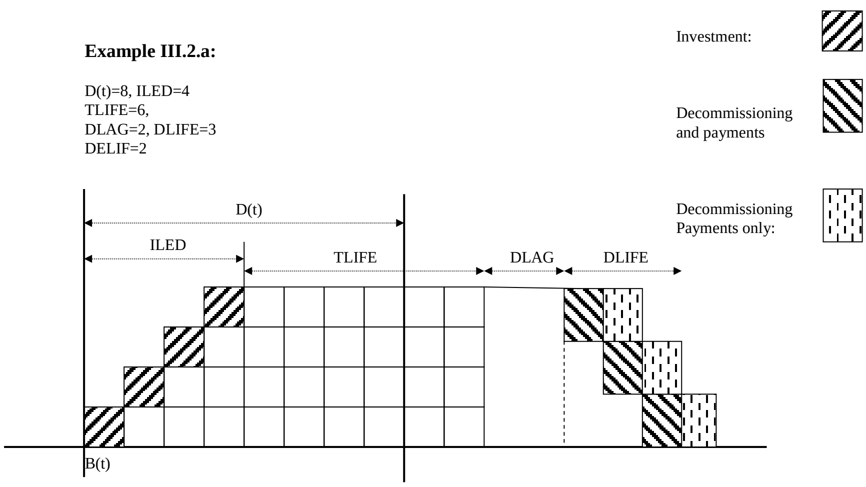


Objective function - Decommissioning

Large, indivisible projects, unrepeated investment in period

Example III.2.a:

D(t)=8, ILED=4
TLIFE=6,
DLAG=2, DLIFE=3
DELIF=2



Objective function - Decommissioning

Large, indivisible projects, repeated investment in period

Example III.2.b:

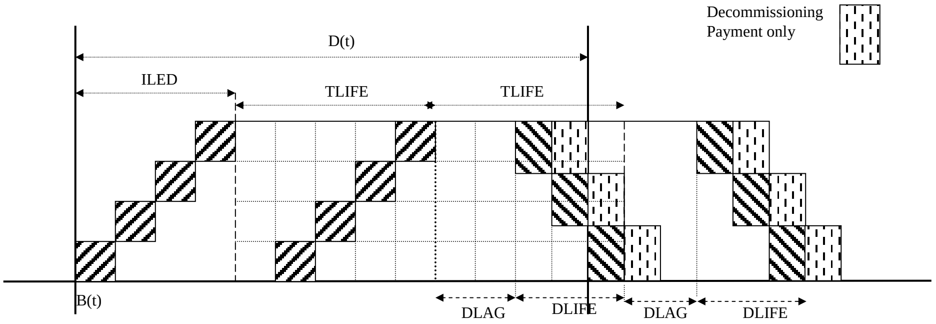
D(t)=13, ILED=4
TLIFE=5,DLAG=2
DLIFE=3, DELIF=2

C=2

Construction

Decommissioning and Payment

Decommissioning Payment only



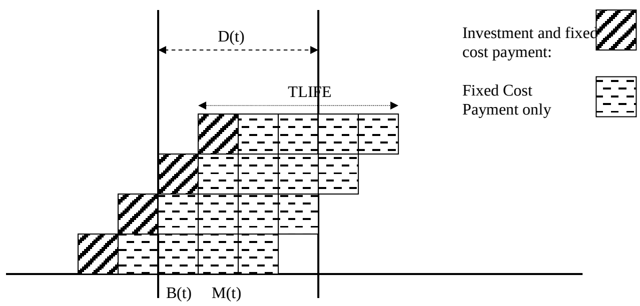
Universität Stuttgart
Institut für Energiewirtschaft und Rationelle Energieanwendung IER

Objective function - Decommissioning

Small, divisible projects, unrepeated investment in period

Example IV.1.a:

D(t)=4, TLIFE=5
M(t)=B(t)+1

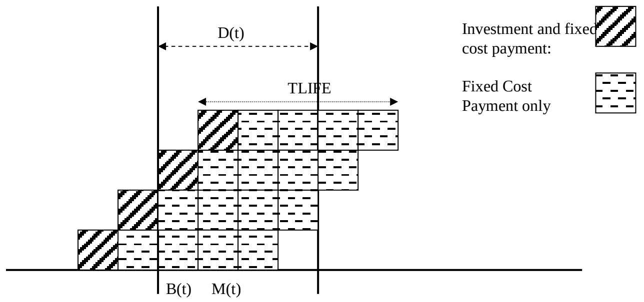


Objective function - Decommissioning

Small, divisible projects, unrepeated investment in period

Example IV.1.a:

D(t)=4, TLIFE=5
M(t)=B(t)+1



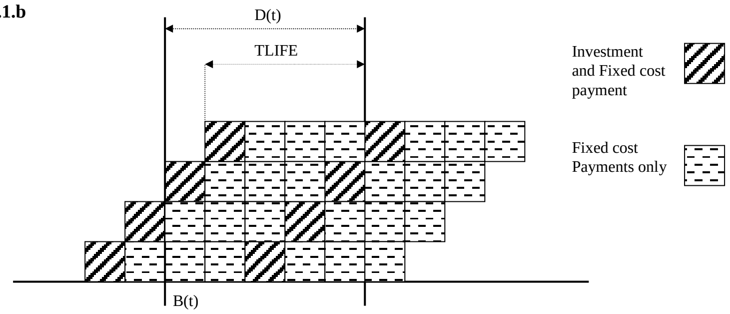
Objective function – Fixed costs

Small projects, repeated investment in period

Example IV.1.b

D=5, TLIFE=4

C=2

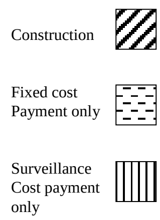


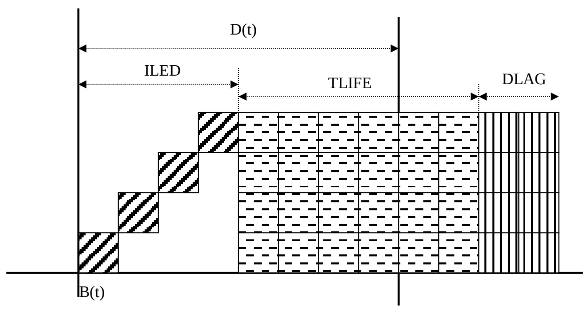
Objective function – Fixed costs

Large, indivisible projects, unrepeated investment in period

Example IV.2.a and IV.2.a':

D(t)=8, ILED=4
TLIFE=6, DLAG=2





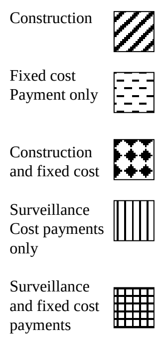
Objective function – Fixed costs

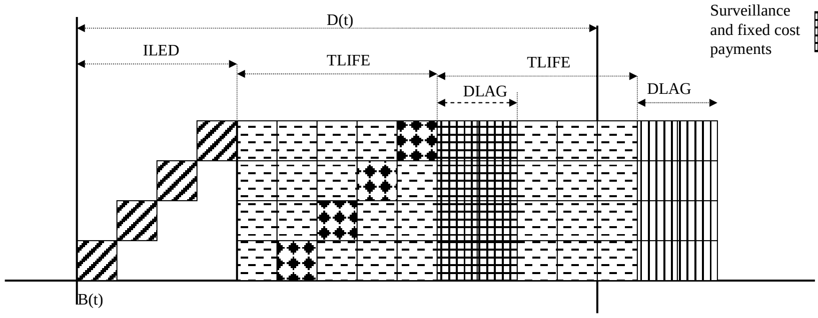
Large, indivisible projects, repeated investment in period

Example for IV.2.b and IV.2.b':

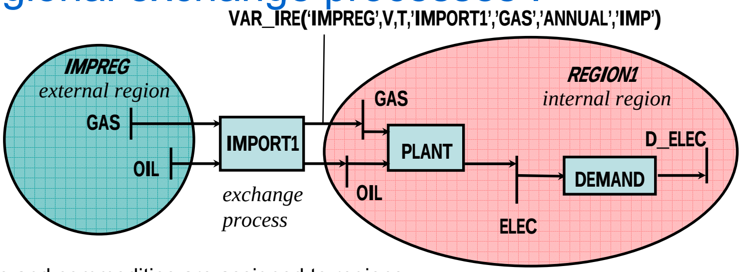
D(t)=13, ILED=4
TLIFE=5, DLAG=2

C=2





Inter-regional exchange processes I



- Processes and commodities are assigned to regions
- Import and export via exchange process between external and internal region
- Sets and parameters for exchange process:

Import Process *Topology*

```
SET PRC /IMPORT1/;  SET TOP_IRE(ALL_REG,COM,ALL_REG,C,PRC)
                        /IMPREG.GAS.REGION1.GAS.IMPORT1
                         IMPREG.OIL.REGION1.OIL.IMPORT1 /;
Import price
PARAMETER IRE_PRICE(REG,ALLYEAR,PRC,COM,TS,ALL_REG,IE,CUR);
IRE_PRICE('REGION1',MILESTONYR,'IMPORT1','GAS','ANNUAL','IMPREG','IMP','DM') = 50000;
IRE_PRICE('REGION1',MILESTONYR,'IMPORT1','OIL','ANNUAL','IMPREG','IMP','DM') = 50000;
Limit on exchange between two regions
PARAMETER IRE_BND(REG,ALLYEAR,COM,TS,ALL_REG,IE,BD);
IRE_BND('REGION1',T,'GAS','ANNUAL','IMPREG','IMP','FX') = 50;
```


Universität Stuttgart
Institut für Energiewirtschaft und Rationelle Energieanwendung IER