

Corso di impiego industriale dell'energia



Esercitazione Gurobi: Ottimizzazione del funzionamento di un sistema trigenerativo

Prof. Rodolfo Taccani

Ing. Pivetta Davide

AA 2020 – 21

Analisi energetica di una cantina di vino

Obiettivi:

- Analizzare le richieste di energia della cantina
- Individuare le unità di conversione e stoccaggio di energia esistenti
- Proposta di altre configurazioni del sistema energetico
- Ottimizzazione del funzionamento delle configurazioni proposte



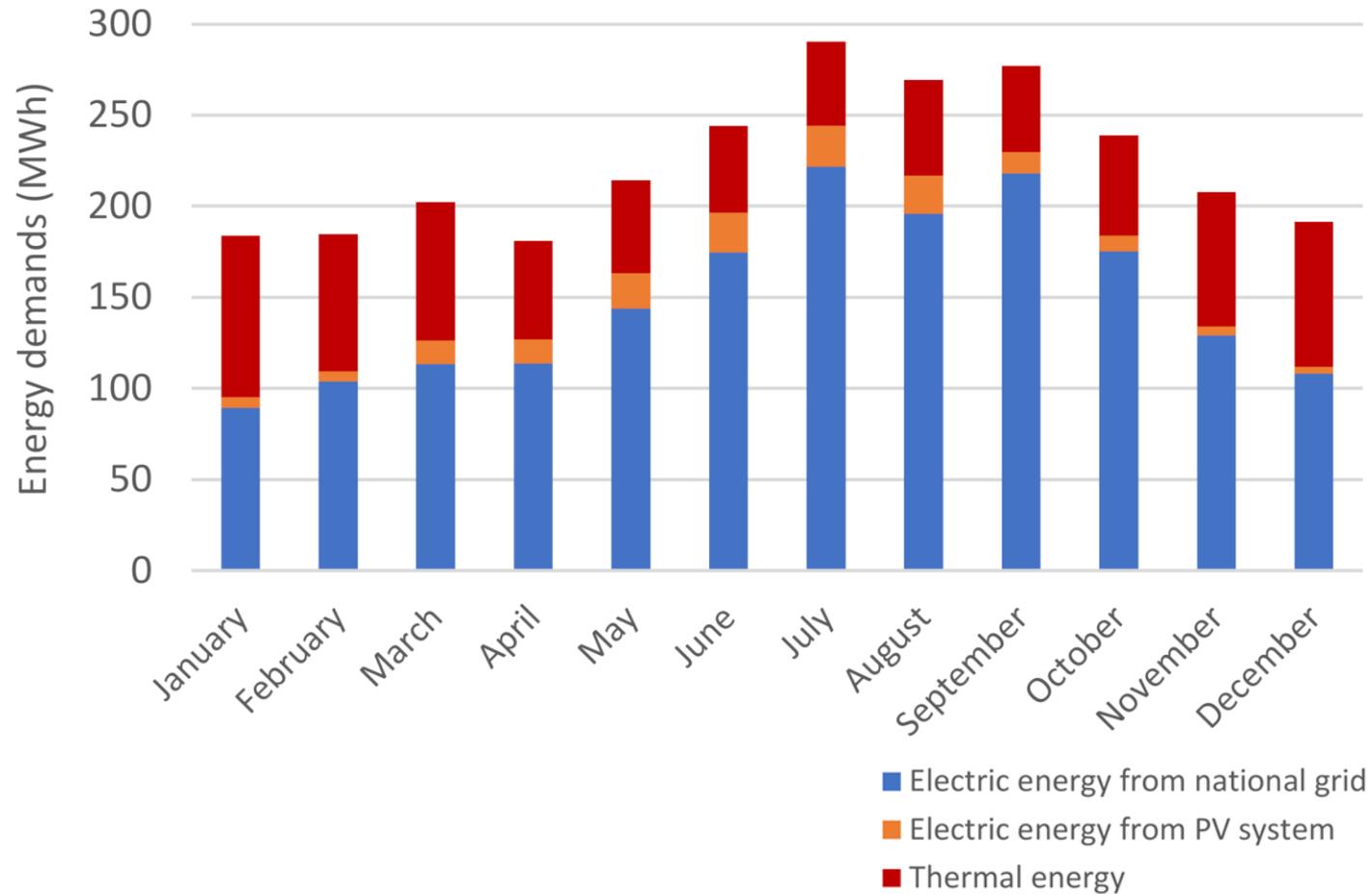
Borgo Molino Vigne&Vini - <http://www.borgomolino.it/>



Roncadelle (TV)

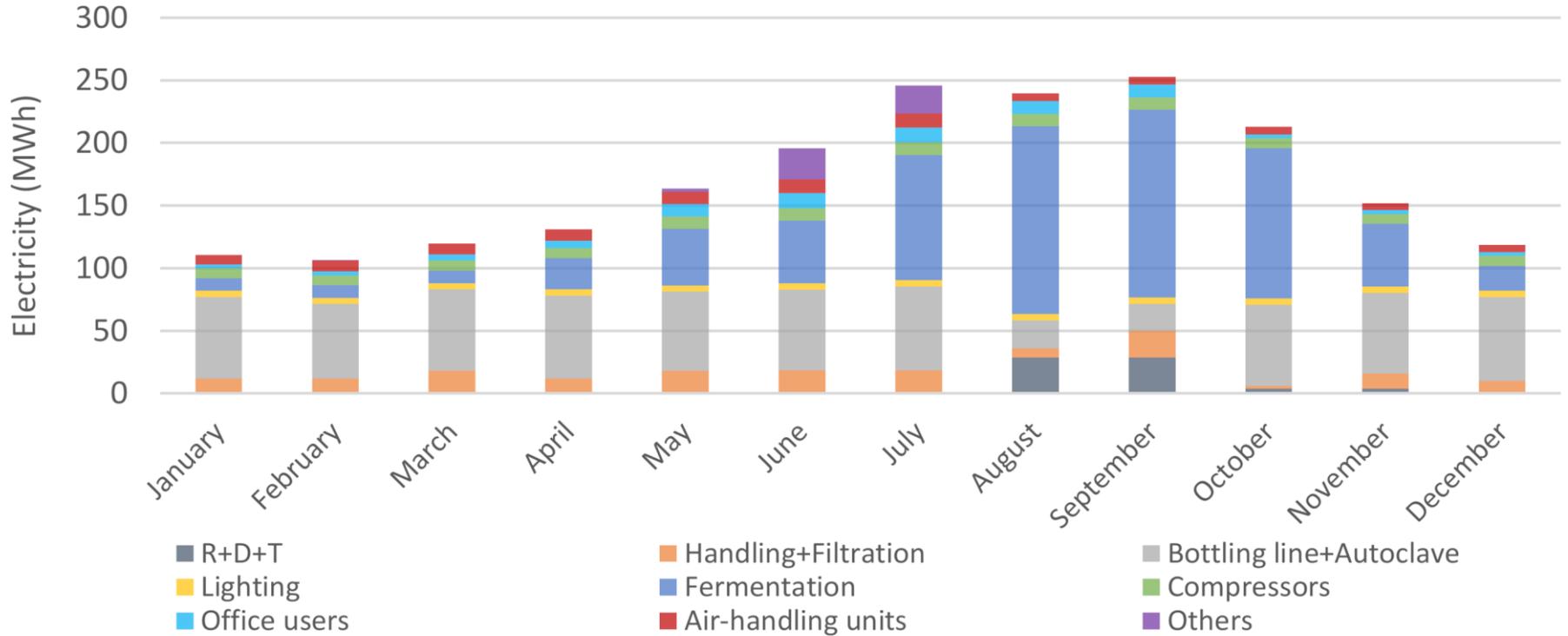


Richiesta energetica complessiva

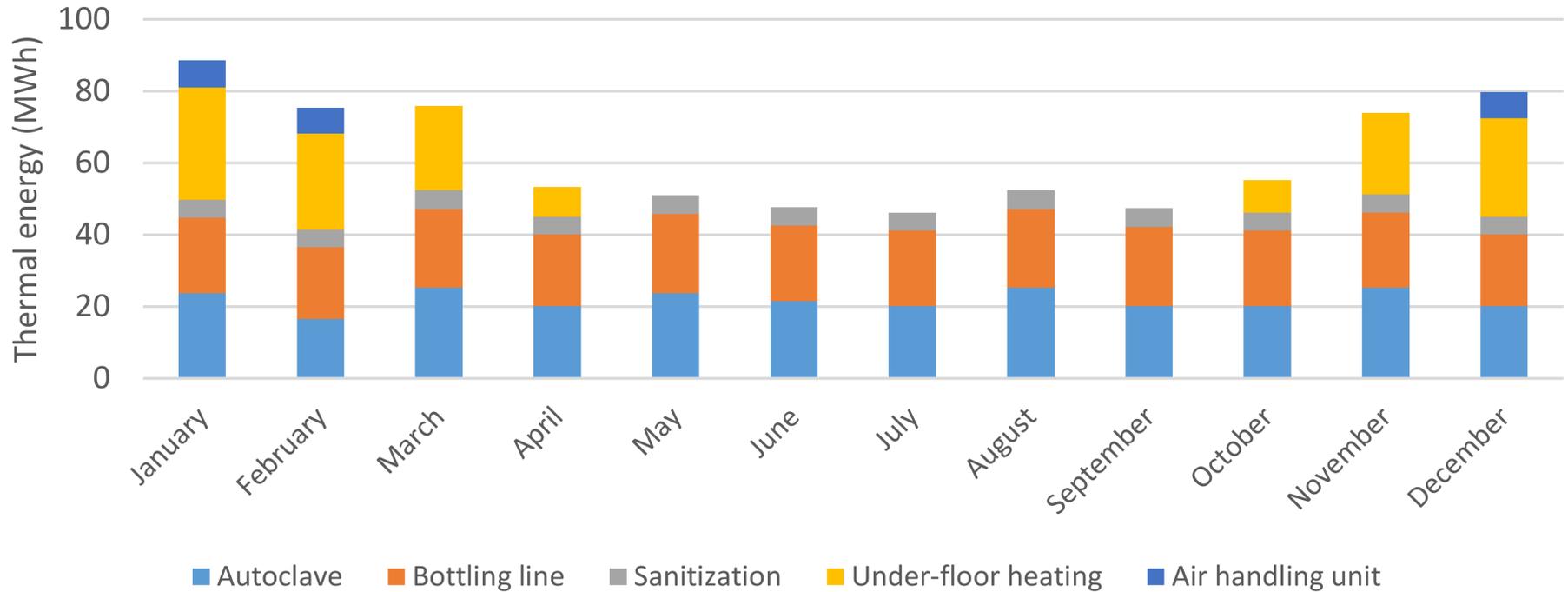


Pivetta et al., Choice of the Optimal Design and Operation of Multi-Energy Conversion Systems in a Prosecco Wine Cellar. Energies 2020

Richiesta di energia elettrica



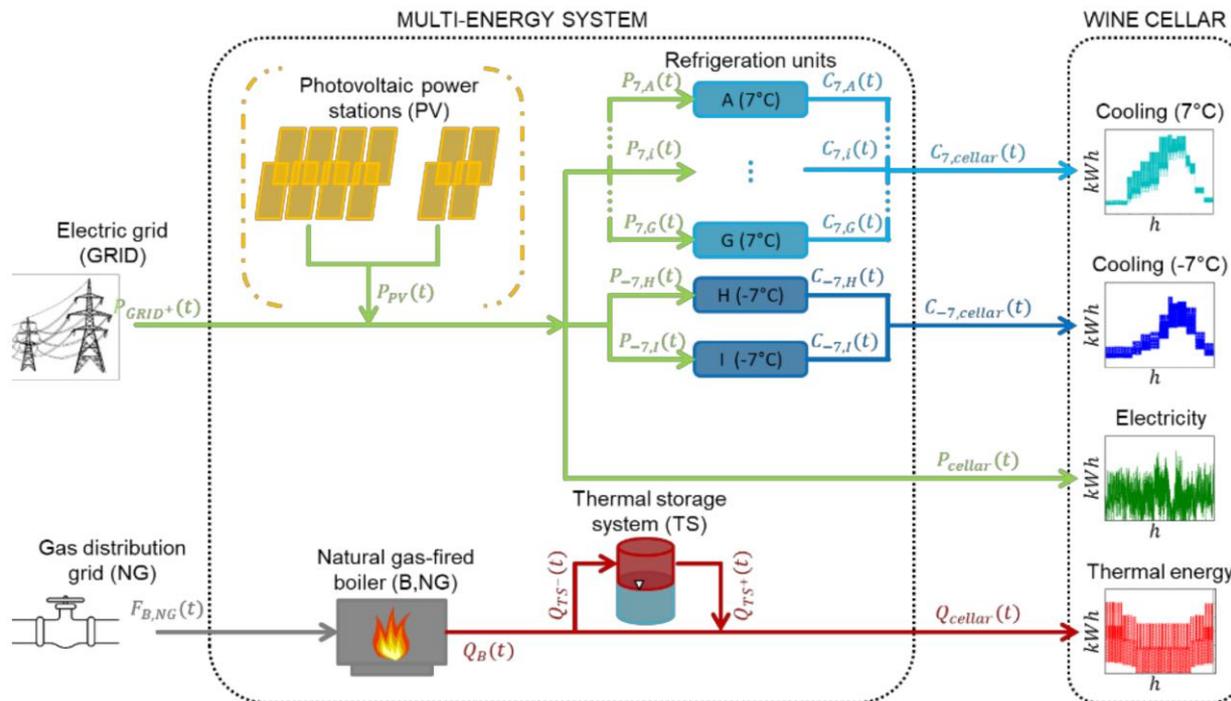
Richiesta di energia termica



Modello dell'impianto esistente

Macchine per la conversione di energia esistenti:

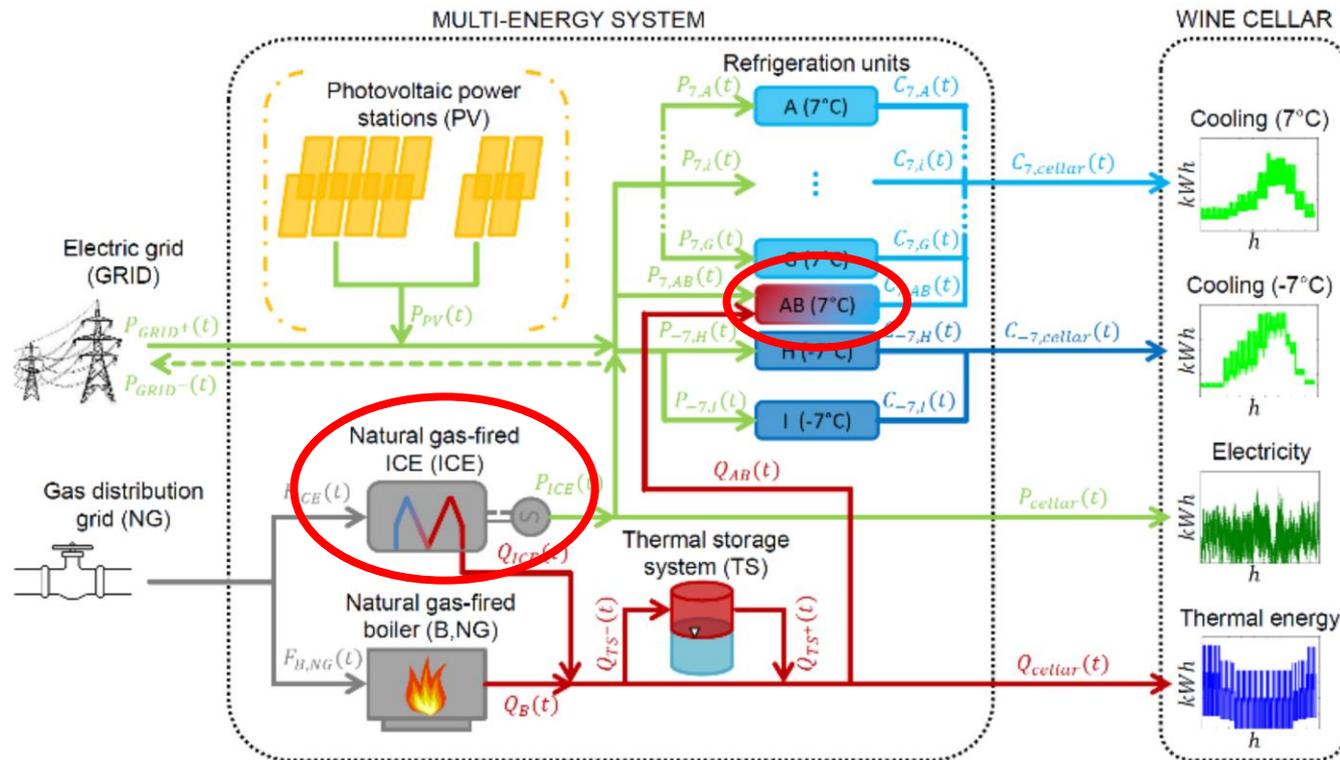
- Caldaia alimentata a gas naturale (275 kW)
- Frigoriferi a compressione (COP \approx 3)
- Impianto fotovoltaico (150 kW_p)



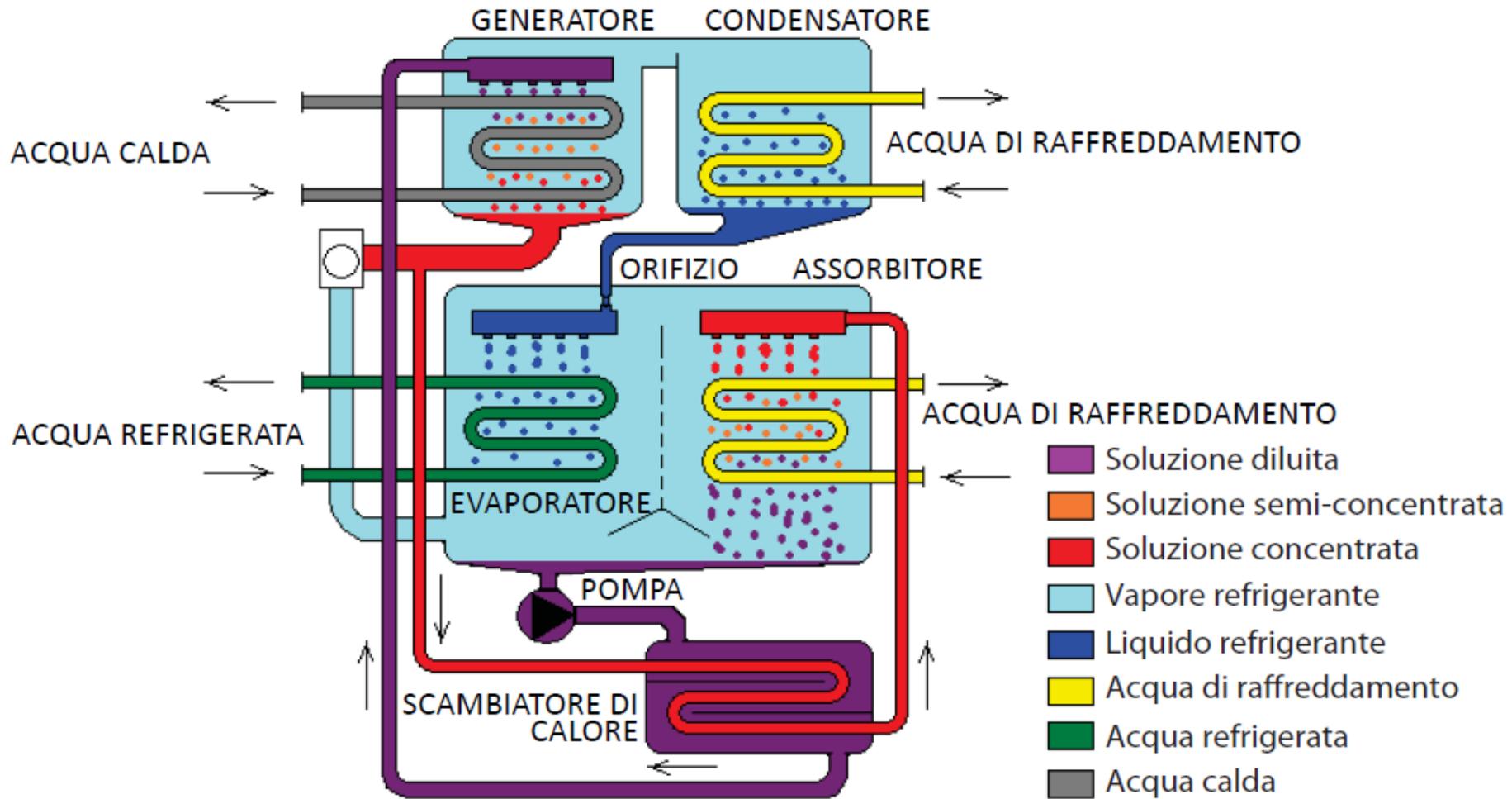
Modello dell'impianto che si vuole proporre

Nuove macchine da installare:

- Motore a combustione interna cogenerativo alimentato a gas naturale (200 kW)
- Assorbitore (176 kW)

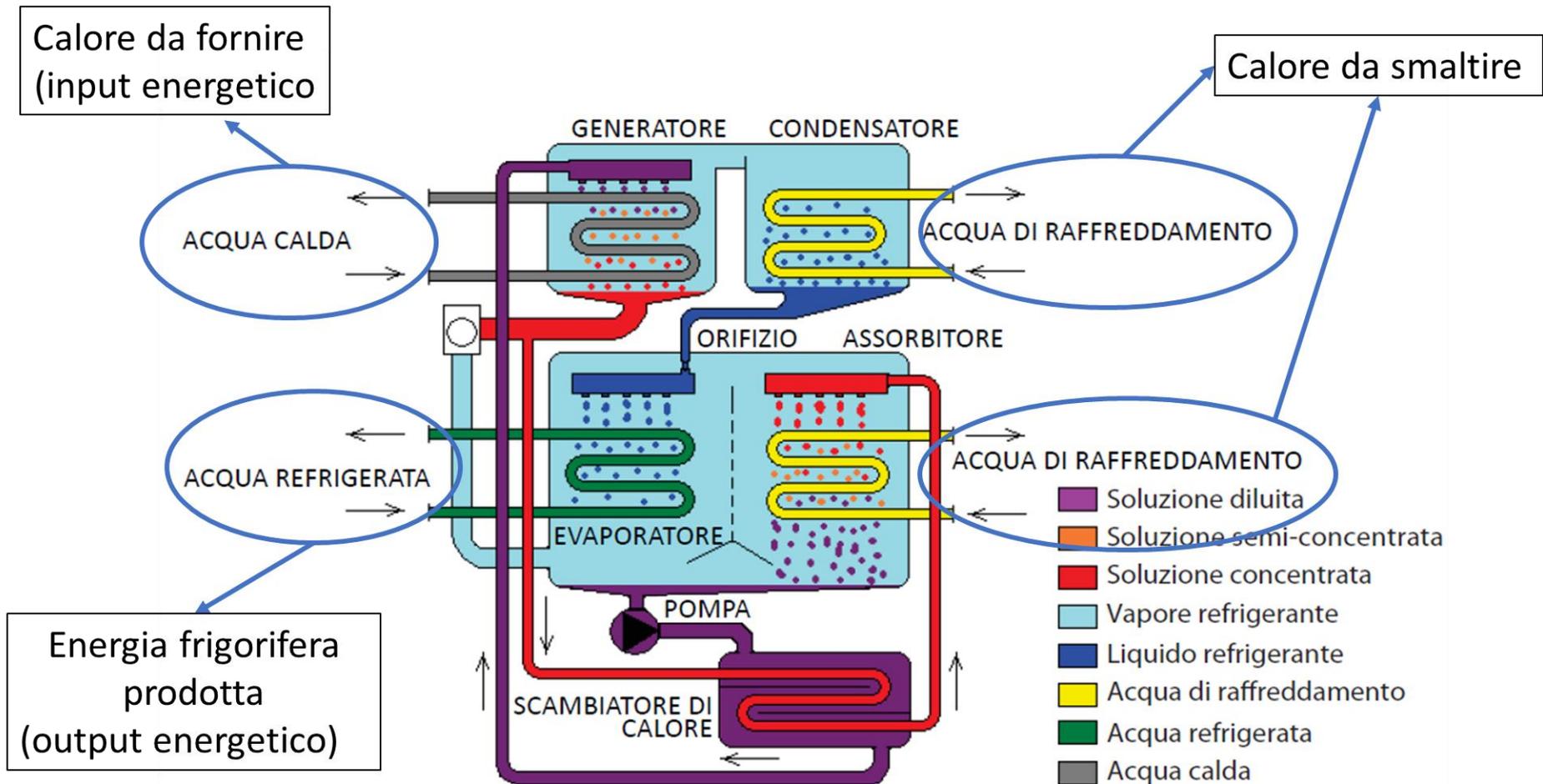


Gruppo frigo ad assorbimento



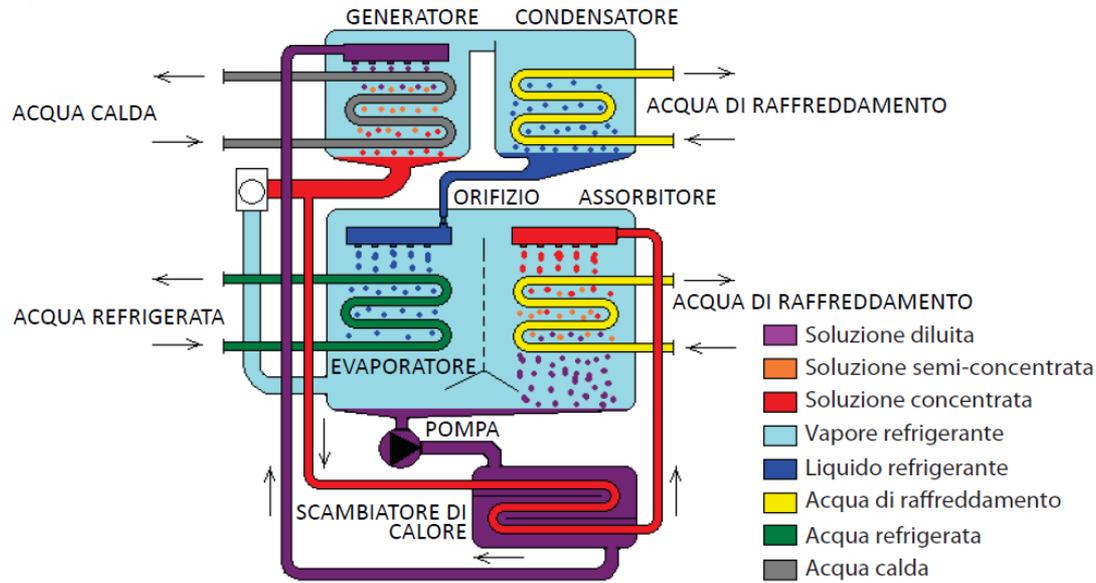
Assorbitori Maya-Yazaki - <https://maya-airconditioning.com/>

Gruppo frigo ad assorbimento



Assorbitori Maya-Yazaki – <https://maya-airconditioning.com/>

Gruppo frigo ad assorbimento



Assorbitori Maya-Yazaki – <https://maya-airconditioning.com/>

NOTE:

- **ASSORBIMENTO:** si utilizzano due fluidi, uno dei quali evapora per produrre energia frigorifera, l'altro è usato come assorbente; ad esempio, acqua (refrigerante) e bromuro di litio (assorbente), o acqua (assorbente) e ammoniaca (refrigerante).
- Come per i normali frigoriferi, si possono operare due o tre livelli di pressione, in modo da aumentare l'efficienza del processo.
- Coefficient Of Performance (COP) = $\frac{E_{OUTPUT}}{E_{INPUT}} = \frac{E_{FRIG}}{E_{TERM} + E_{ELETT}} \approx 0,7$

Problema di ottimizzazione MILP (Mixed-Integer Linear Programming)



Trovare $x^*(t), \beta^*, \delta^*$ che massimizzano/minimizzano:

$$Z = f(x^*(t), \beta^*, \delta^*) \quad (1)$$

s.v. $g(x^*(t), \beta^*, \delta^*) = 0 \quad (2)$

$$h(x^*(t), \beta^*, \delta^*) \leq 0 \quad (3)$$

Dove:

$x^*(t)$ Valore ottimo variabile continua;

β^*, δ^* Valore ottimo variabile intera o binaria;

Z Funzione obiettivo;

$g(x^*(t), \beta^*, \delta^*)$ Vincolo di uguaglianza;

$h(x^*(t), \beta^*, \delta^*)$ Vincolo di disuguaglianza.

Vincoli: $g(x^*(t), \beta^*, \delta^*)$ e $h(x^*(t), \beta^*, \delta^*)$

Vincoli $g(x^*(t), \beta^*, \delta^*)$ e $h(x^*(t), \beta^*, \delta^*)$

Unità di conversione di energia

$$\varphi_{in}(t) = K1_i * \varphi_{out}(t) + K2_i * \delta_i(t)$$

$$\varphi_{out}(t) \leq K3_i * \varphi_{out,max} * \delta_i(t)$$

$$\varphi_{out}(t) \geq K4_i * \varphi_{out,min} * \delta_i(t)$$

Unità di accumulo di energia

$$V_{TS}(t) = V_{TS}(t-1) + \left(\frac{1}{\rho_{TS} * c_{p,TS} * (\theta_{TS,hot} - \theta_{TS,cold})} \right) * \left(\eta_H * F_{TS}(t) - \frac{1}{\eta_H} * QH_{TS}(t) \right) * dt$$

$$0.1 * V_{TS,max} \leq V_{TS}(t) \leq 0.9 * V_{TS,max}$$

ICE cogenerativo

$$F_{ICE}(t) = K1_{ICE} * P_{ICE}(t) + K2_{ICE} * \delta_{ICE}(t)$$

$$QH_{ICE}(t) \leq K3_{ICE} * P_{ICE}(t) + K4_{ICE} * \delta_{ICE}(t)$$

$$P_{ICE,min} * \delta_{ICE}(t) \leq P_{ICE}(t) \leq P_{ICE,max} * \delta_{ICE}(t)$$

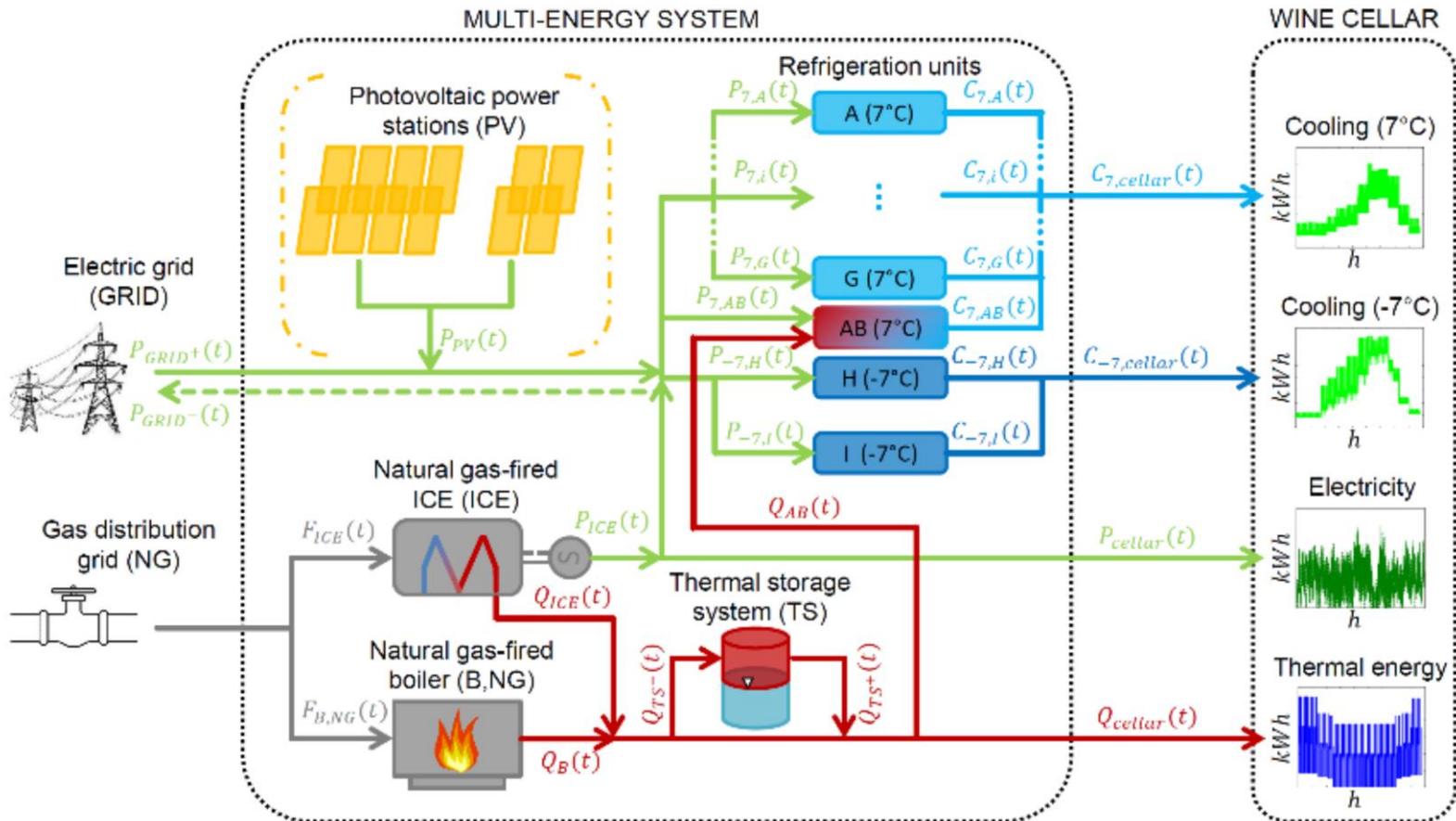
Gruppo frigo ad assorbimento

$$QH_{AB}(t) = K1_{AB} * Q7_{AB}(t) + K2_{AB} * \delta_{AB}(t)$$

$$P_{AB}(t) = K3_{AB} * Q7_{AB}(t) + K4_{AB} * \delta_{AB}(t)$$

$$Q7_{AB,min} * \delta_{AB}(t) \leq Q7_{AB}(t) \leq Q7_{AB,max} * \delta_{AB}(t)$$

Modello del sistema energetico cantina



Funzioni obiettivo: $Z = f(x^*(t), \delta^*)$

Minimization of cost

$$\sum F_{GN}(t) * c_{GN} + \sum P_{GR_{pos}}(t) * c_{GR_{pos}} - \sum P_{GR_{neg}}(t) * c_{GR_{neg}} + \sum F_{ICE}(t) * c_{ICE}$$

Minimization of primary energy consumption

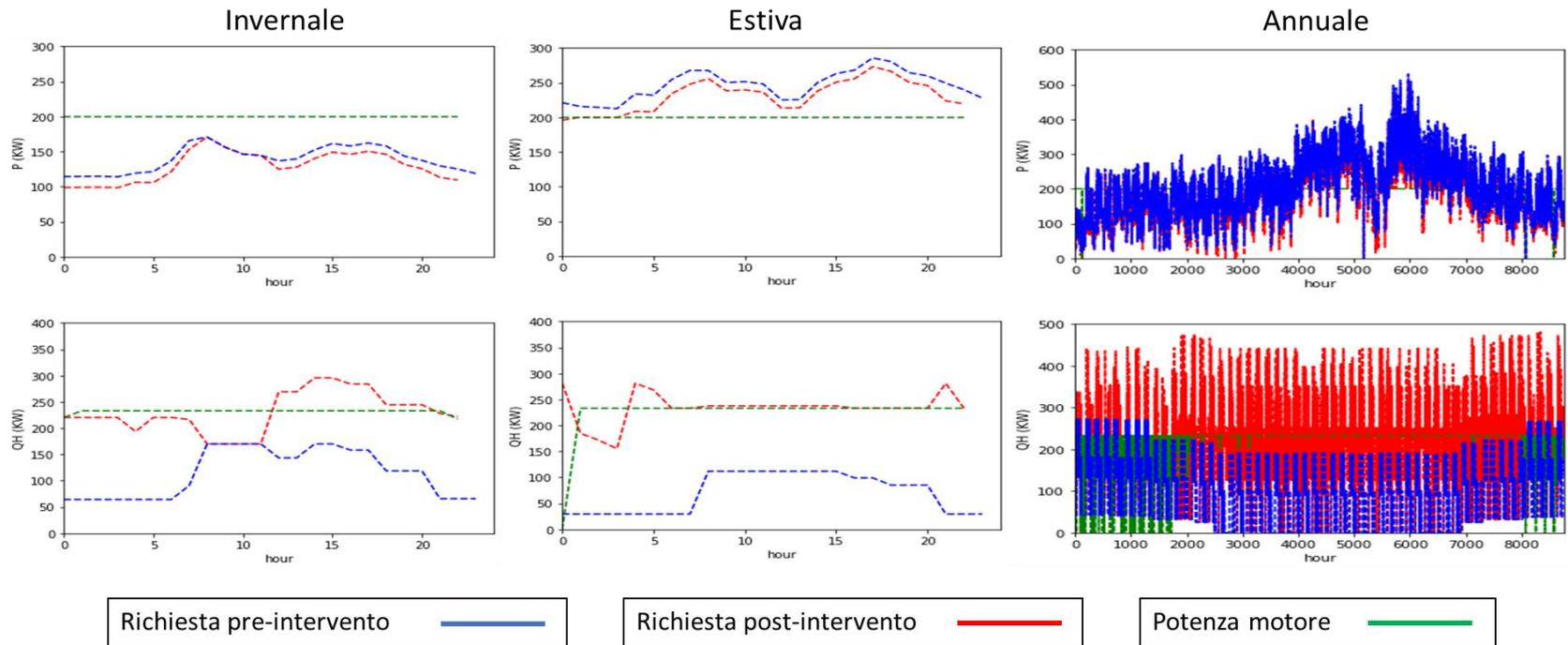
$$\left(\sum F_{GN}(t) + F_{ICE}(t) \right) * e_{GN_t} + \left(\sum P_{GR_{pos}}(t) - \sum P_{GR_{neg}}(t) \right) * e_{GR_t}$$

Minimization of primary energy consumption from fossil fuels

$$\left(\sum F_{GN}(t) + F_{ICE}(t) \right) * e_{GN} + \left(\sum P_{GR_{pos}}(t) - \sum P_{GR_{neg}}(t) \right) * e_{GR}$$

Risultati ottimizzazione

Minimizzazione dei costi d'esercizio



Risultati ottimizzazione

Minimizzazione costo

	Costo operativo (€/anno)	Costi di manutenzione (€/anno)	Costo totale (€/anno)	Risparmio ottenuto (€/anno)	Payback time (anni)	VAN* a 25 anni (€)
1. Rete + Caldaia Gas	288.509	0	288.509	0	/	/
2. Motore	160.360	35.000	195.360	93.149	3,8	962.832
3. Motore + Assorb.	142.046	45.000	187.046	101.463	4,4	980.009

*t.a.=5%

Minimizzazione energia primaria

	Energia primaria consumata (MWh)	Energia primaria risparmiata (MWh)
1. Rete + Caldaia Gas	5201	/
2. Motore	4466	735
3. Motore + Assorb.	4284	917

Minimizzazione energia primaria da fonte fossile

	En. Prim. consumata da fonte fossile (MWh)	En. Prim. risparmiata da fonte fossile (MWh)
1. Rete + Caldaia Gas	4361	/
2. Motore	3935	426
3. Motore + Assorb.	3935	426

Risultati da impianto esistente



Trigeneratore nell'industria vitivinicola

Risparmiare sui costi energetici. Contenere il fabbisogno di energia primaria riducendo l'impatto ambientale.

www.espiu.it

820.000 kWh

Produzione elettrica annua

9,67 TEP

Risparmio annuo energia primaria

90.000,00 €

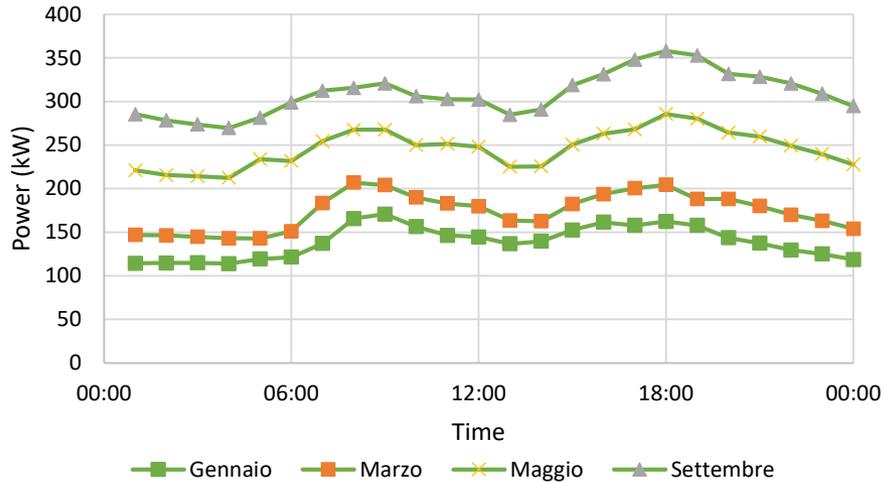
Risparmi annui generati

ESERCITAZIONE

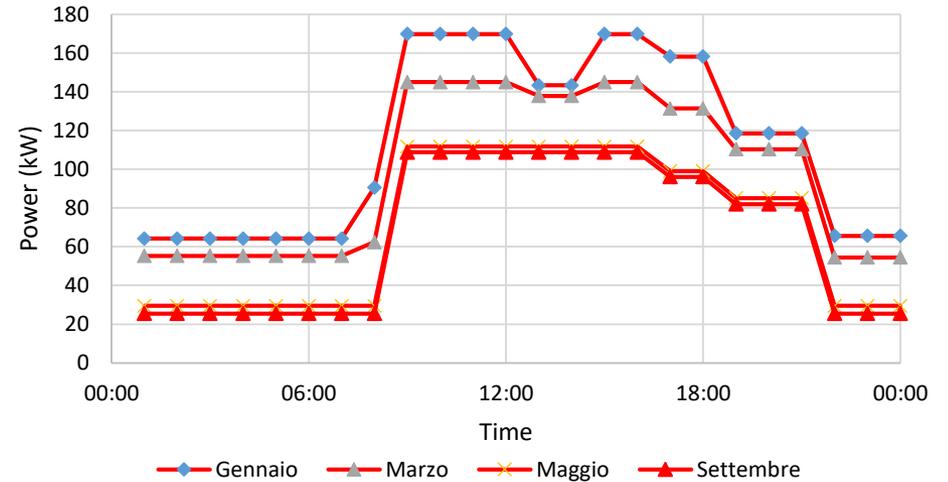


Importiamo i dati

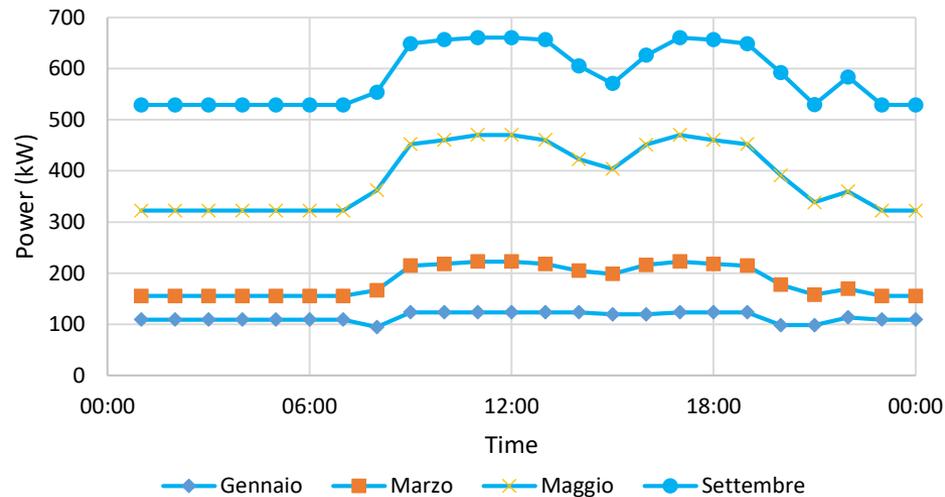
Potenza elettrica



Potenza termica



Potenza cooling



Importiamo i dati

```
1 import openpyxl as xl
2 from openpyxl import Workbook
3
4 wb=xl.load_workbook('Data.xlsx')
5
6 sheet1=wb['Gennaio']
7 sheet2=wb['Marzo']
8 sheet3=wb['Maggio']
9 sheet4=wb['Settembre']
10
11 P_el=[]
12 P_th=[]
13 P_cool=[]
14
15 print('Scegli il mese in cui avviare ottimizzazione')
16 print('')
17 print('Gennaio (1), marzo (2), maggio (3), settembre (4)')
18 print('')
19 mese=int(input())
20
21 A=2
22 B=25
23
24 if mese==1:
25     while A<=B:
26         P_el.append(float((sheet1.cell(A,2)).value))
27         P_th.append(float((sheet1.cell(A,3)).value))
28         P_cool.append(float((sheet1.cell(A,4)).value))
29         A+=1
30
```

•••••

```
44
45 elif mese==4:
46     while A<=B:
47         P_el.append(float((sheet4.cell(A,2)).value))
48         P_th.append(float((sheet4.cell(A,3)).value))
49         P_cool.append(float((sheet4.cell(A,4)).value))
50         A+=1
51
52 else:
53     print('')
54     print('Non hai digitato correttamente il numero corrispondente al mese')
55
56
```

Motore a combustione interna

```
#ICE
K1_ICE=2.2678
K2_ICE=46.443
K3_ICE=0.9356
K4_ICE=45.679
P_ICE_min=100
P_ICE_max=200
Q_ICE_min=140
Q_ICE_max=234
```

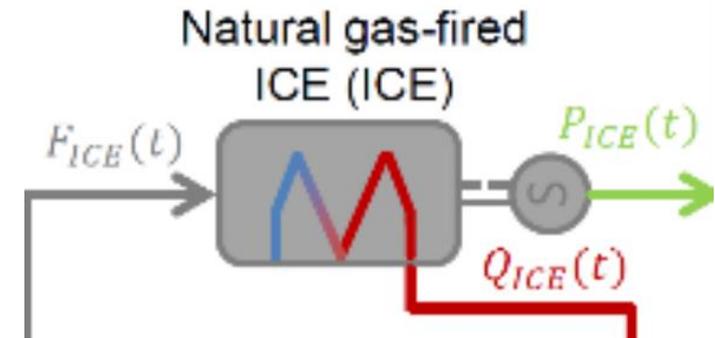
Variabile continua di riferimento per il motore

Variabile binaria che definisce stato on/off (1 o 0)

$$F_{ICE}(t) = K1_{ICE} * P_{ICE}(t) + K2_{ICE} * \delta_{ICE}(t)$$

$$QH_{ICE}(t) \leq K3_{ICE} * P_{ICE}(t) + K4_{ICE} * \delta_{ICE}(t)$$

$$P_{ICE,min} * \delta_{ICE}(t) \leq P_{ICE}(t) \leq P_{ICE,max} * \delta_{ICE}(t)$$

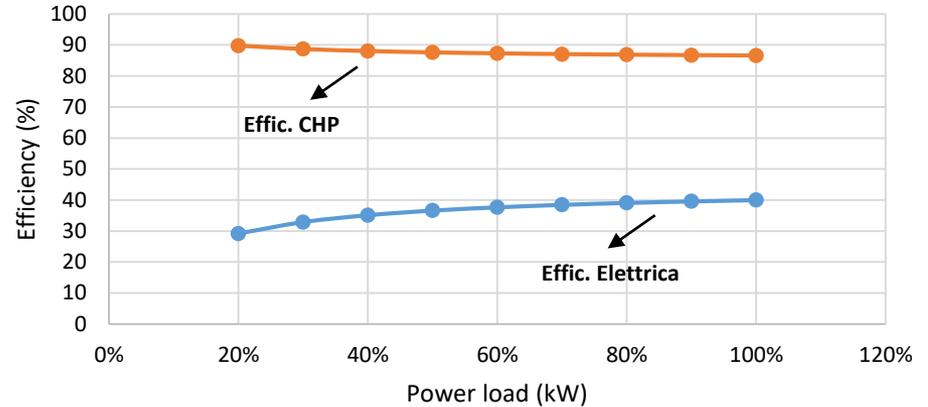


```
#Variabili per il motore
P_ICE=m.addVars(len(P_e1),lb=P_ICE_min,ub=P_ICE_max,vtype=GRB.CONTINUOUS,name=("P_ICE")) #0
QH_ICE=m.addVars(len(P_e1),vtype=GRB.CONTINUOUS,name=("QH_ICE")) #1
D_ICE=m.addVars(len(P_e1),vtype=GRB.BINARY,name="D_ICE") #2
F_ICE=m.addVars(len(P_e1),vtype=GRB.CONTINUOUS,name=("F_ICE")) #3
```

```
#Motore a combustione interna
m.addConstrs(F_ICE[temp]==(K1_ICE*P_ICE[temp]+K2_ICE)*D_ICE[temp] for temp in range(0,len(P_e1)))
m.addConstrs(QH_ICE[temp]<=(K3_ICE*P_ICE[temp]+K4_ICE)*D_ICE[temp] for temp in range(0,len(P_e1)))
```

Motore a combustione interna

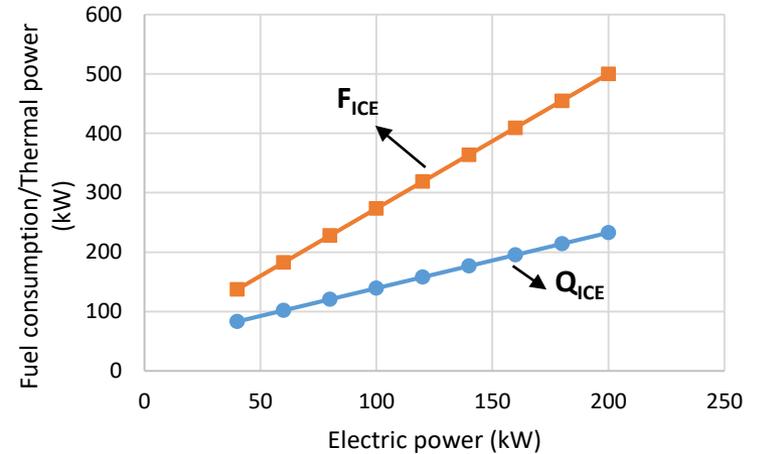
```
#ICE
K1_ICE=2.2678
K2_ICE=46.443
K3_ICE=0.9356
K4_ICE=45.679
P_ICE_min=100
P_ICE_max=200
Q_ICE_min=140
Q_ICE_max=234
```



$$F_{ICE}(t) = K1_{ICE} * P_{ICE}(t) + K2_{ICE} * \delta_{ICE}(t)$$

$$QH_{ICE}(t) \leq K3_{ICE} * P_{ICE}(t) + K4_{ICE} * \delta_{ICE}(t)$$

$$P_{ICE,min} * \delta_{ICE}(t) \leq P_{ICE}(t) \leq P_{ICE,max} * \delta_{ICE}(t)$$

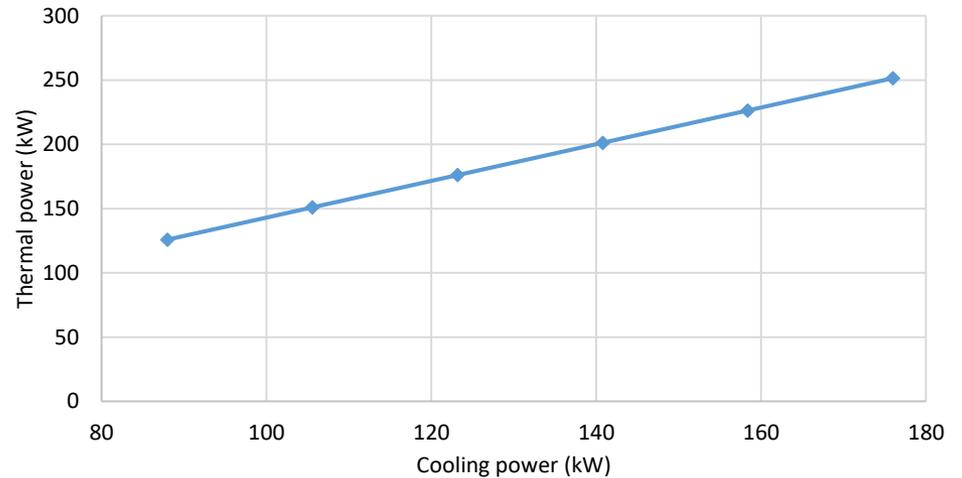


```
#Variabili per il motore
P_ICE=m.addVars(len(P_e1),lb=P_ICE_min,ub=P_ICE_max,vtype=GRB.CONTINUOUS,name=("P_ICE")) #0
QH_ICE=m.addVars(len(P_e1),vtype=GRB.CONTINUOUS,name=("QH_ICE")) #1
D_ICE=m.addVars(len(P_e1),vtype=GRB.BINARY, name="D_ICE") #2
F_ICE=m.addVars(len(P_e1),vtype=GRB.CONTINUOUS,name=("F_ICE")) #3
```

```
#Motore a combustione interna
m.addConstrs(F_ICE[temp]==(K1_ICE*P_ICE[temp]+K2_ICE)*D_ICE[temp] for temp in range(0,len(P_e1)))
m.addConstrs(QH_ICE[temp]<=(K3_ICE*P_ICE[temp]+K4_ICE)*D_ICE[temp] for temp in range(0,len(P_e1)))
```

Gruppo frigo ad assorbimento

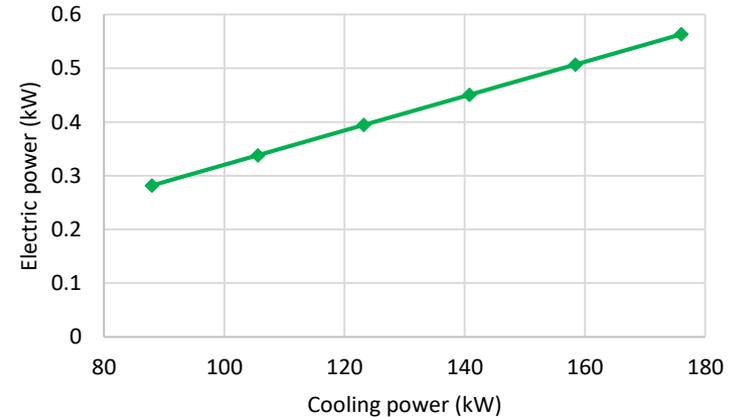
```
#ABSORPTION CHILLER
K1_AB=1.42857
K2_AB=0
K3_AB=0.0032
K4_AB=0
Q7_AB_min=88
Q7_AB_max=176
```



$$QH_{AB}(t) = K1_{AB} * Q7_{AB}(t) + K2_{AB} * \delta_{AB}(t)$$

$$P_{AB}(t) = K3_{AB} * Q7_{AB}(t) + K4_{AB} * \delta_{AB}(t)$$

$$Q7_{AB,min} * \delta_{AB}(t) \leq Q7_{AB}(t) \leq Q7_{AB,max} * \delta_{AB}(t)$$



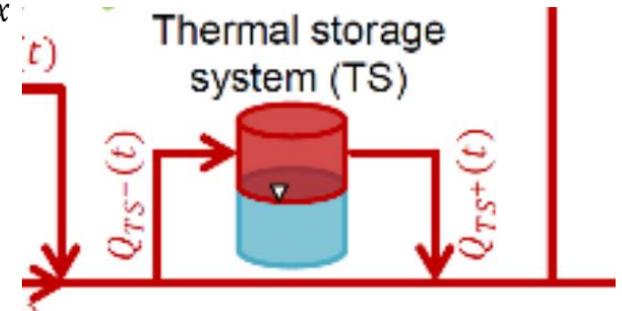
```
#Variabili per il chiller ad assorbimento
Qc_AB=m.addVars(len(P_el),lb=Q7_AB_min,ub=Q7_AB_max,vtype=GRB.CONTINUOUS,name=("Qc_AB")) #10
D_AB=m.addVars(len(P_el),vtype=GRB.BINARY, name="D_AB") #11
QH_AB=m.addVars(len(P_el),vtype=GRB.CONTINUOUS,name=("QH_AB")) #12
P_AB=m.addVars(len(P_el),vtype=GRB.CONTINUOUS,name=("P_AB")) #13
```

```
#Chiller ad assorbimento
m.addConstrs(QH_AB[temp]==(K1_AB*Qc_AB[temp]+K2_AB)*D_AB[temp] for temp in range(0,len(P_el)))
m.addConstrs(P_AB[temp]==(K3_AB*Qc_AB[temp]+K4_AB)*D_AB[temp] for temp in range(0,len(P_el)))
```

Accumulo termico (acqua calda)

$$V_{TS}(t) = V_{TS}(t-1) + \left(\frac{1}{\rho_{TS} * c_{p,TS} * (\theta_{TS,hot} - \theta_{TS,cold})} \right) * \left(\eta_H * F_{TS}(t) - \frac{1}{\eta_H} * Q_{H_{TS}}(t) \right) * dt$$

$$0.1 * V_{TS,max} \leq V_{TS}(t) \leq 0.9 * V_{TS,max}$$



```
#STORAGE
V_TS_max=10.0
rho=1000 #kg/m3
cp=4.186 #kJ/kgK
Thot=90 #°C
Tcold=50 #°C
Coef_Storage=1/(rho*cp*(Thot-Tcold)/3600)
```

```
#Variabili per l'accumulo termico
```

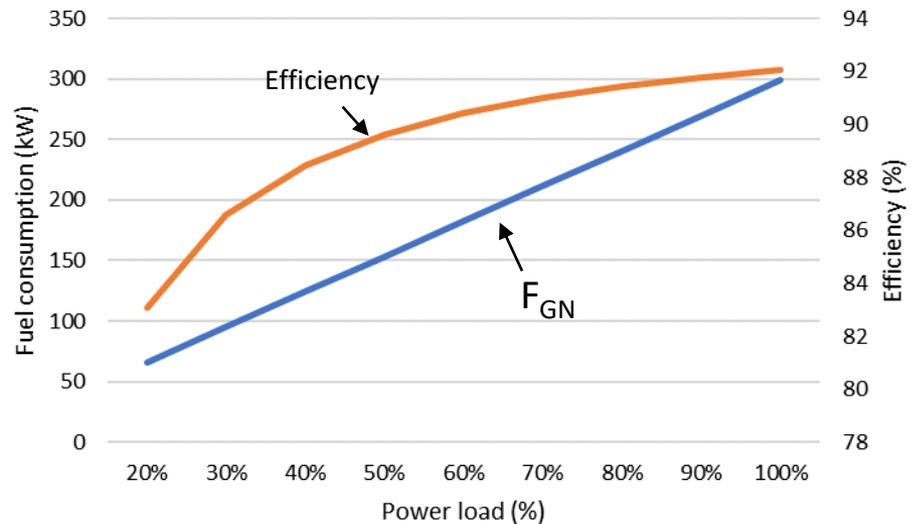
```
V_TS=m.addVars(len(P_el),lb=1.0,ub=9.0,vtype=GRB.CONTINUOUS,name=("V_TS")) #16
F_TS=m.addVars(len(P_el),lb=0,ub=1000,vtype=GRB.CONTINUOUS,name=("F_TS")) #17
Q_TS=m.addVars(len(P_el),lb=0,ub=1000,vtype=GRB.CONTINUOUS,name=("Q_TS")) #18
D_TS=m.addVars(len(P_el),vtype=GRB.BINARY,name=("D_TS")) #19
```

```
#Accumulo termico
```

```
m.addConstrs(V_TS[temp]==V_TS[temp-1]+Coef_Storage*(0.95*F_TS[temp]-1/0.95*Q_TS[temp]) for temp in range(1,len(P_el)))
m.addConstr(V_TS[0]<=V_TS[len(P_el)-1], "c0")
m.addConstrs(F_TS[temp]==F_TS[temp]*(D_TS[temp]) for temp in range(0,len(P_el)))
m.addConstrs(Q_TS[temp]==Q_TS[temp]*(1-D_TS[temp]) for temp in range(0,len(P_el)))
```

Caldia a gas naturale

```
#BOILER NATURAL GAS  
K1_GN=1.0571  
K2_GN=8.0563  
QH_GN_min=67.5  
QH_GN_max=275
```



```
#Variabili per la caldaia  
QH_GN=m.addVars(len(P_el),lb=QH_GN_min,ub=QH_GN_max,vtype=GRB.CONTINUOUS,name=("QH_GN")) #7  
D_GN=m.addVars(len(P_el),vtype=GRB.BINARY, name="D_GN") #8  
F_GN=m.addVars(len(P_el),vtype=GRB.CONTINUOUS,name=("F_GN")) #9
```

```
#Boiler a gas  
m.addConstrs(F_GN[temp]==(K1_GN*QH_GN[temp]+K2_GN)*D_GN[temp] for temp in range(0,len(P_el)))
```

Chillers a compressione

```
#GENERIC CHILLER  
COP_x=3
```

$$Qc_x(t) = COP_x * P_x(t)$$

```
#Variabili per un chiller a compressione generico  
Qc_x=m.addVars(len(P_el),vtype=GRB.CONTINUOUS,name=("Qc_x")) #14  
P_x=m.addVars(len(P_el),lb=0,vtype=GRB.CONTINUOUS,name=("P_x")) #15
```

```
#Chiller generico a compressione  
m.addConstrs(Qc_x[temp]==COP_x*P_x[temp] for temp in range(0,len(P_el)))
```

Rete elettrica

$$\text{if } P_{GR_{pos}} > 0 \rightarrow P_{GR_{neg}} = 0$$

$$\text{if } P_{GR_{neg}} > 0 \rightarrow P_{GR_{pos}} = 0$$

```
#Variabili per la rete elettrica  
P_GR_pos=m.addVars(len(P_el),lb=0,ub=1000,vtype=GRB.CONTINUOUS,name=("P_GR_pos")) #4  
P_GR_neg=m.addVars(len(P_el),lb=0,ub=1000,vtype=GRB.CONTINUOUS,name=("P_GR_neg")) #5  
D_GR=m.addVars(len(P_el),vtype=GRB.BINARY, name="D_GR") #6
```

```
#Rete elettrica  
m.addConstrs(P_GR_pos[temp]*D_GR[temp]==P_GR_pos[temp] for temp in range(0,len(P_el)))  
m.addConstrs(P_GR_neg[temp]*(1-D_GR[temp])==P_GR_neg[temp] for temp in range(0,len(P_el)))
```

Bilanci di energia

```
#Potenza elettrica
m.addConstrs(P_GR_pos[temp]*D_GR[temp]-P_GR_neg[temp]*(1-D_GR[temp])+P_ICE[temp]*D_ICE[temp]==P_el[temp]+P_x[temp]+P_AB[temp]*D_AB[temp] for temp in range(0,len(P_el)))

#Potenza termica
m.addConstrs(QH_ICE[temp]*D_ICE[temp]+Q_TS[temp]+QH_GN[temp]*D_GN[temp]==P_th[temp]+QH_AB[temp]*D_AB[temp]+F_TS[temp] for temp in range(0,len(P_el)))

#Potenza cooling
m.addConstrs(Qc_AB[temp]*D_AB[temp]+Qc_x[temp]==P_cool[temp] for temp in range(0,len(P_el)))
```

Funzioni obiettivo

```
#Minimizzazione del costo operativo dell'intero impianto
m.setObjective(P_GR_pos.sum()*c_GR_pos-P_GR_neg.sum()*c_GR_neg+F_ICE.sum()*c_ICE+F_GN.sum()*c_GN,GRB.MINIMIZE)

#Minimizzazione del consumo di energia primaria
m.setObjective((P_GR_pos.sum()-P_GR_neg.sum())*e_P_GRt+(F_ICE.sum()+F_GN.sum())*e_GNt,GRB.MINIMIZE)

#Minimizzazione del consumo di energia primaria da fonte fossile
m.setObjective((P_GR_pos.sum()-P_GR_neg.sum())*e_P_GR+(F_ICE.sum()+F_GN.sum())*e_GN,GRB.MINIMIZE)
```

```
#Prices
c_GR_pos=0.140
c_GR_neg=0.075
c_ICE=0.03107
c_GN=0.0374
```

```
#Primary energy
e_P_GRt=2.42
e_GNt=1.05
```

↓

Considero un'efficienza media di produzione elettrica in Italia pari a circa 42%

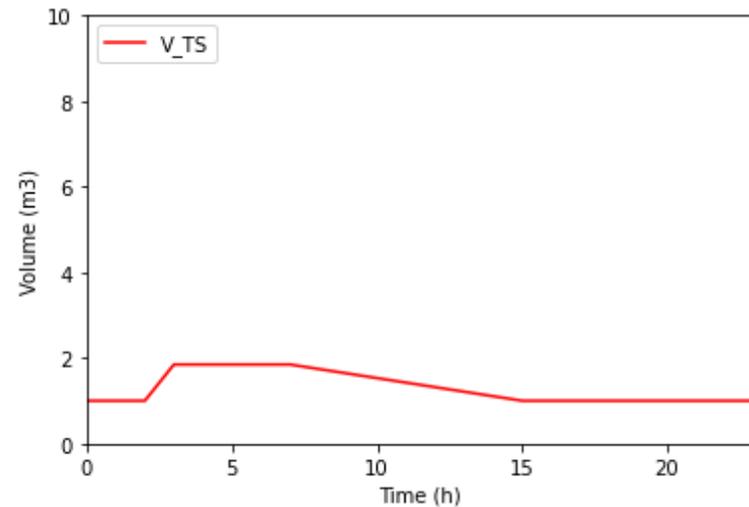
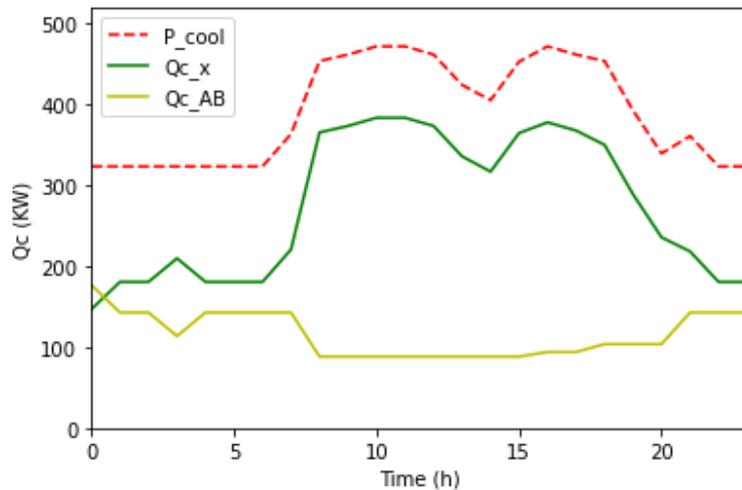
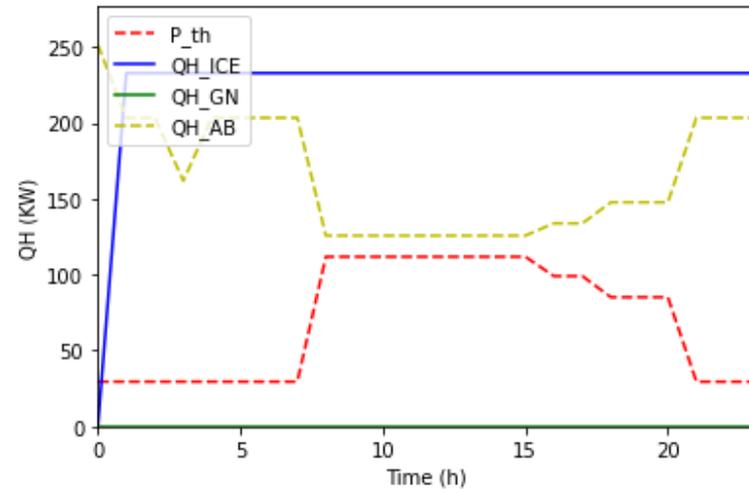
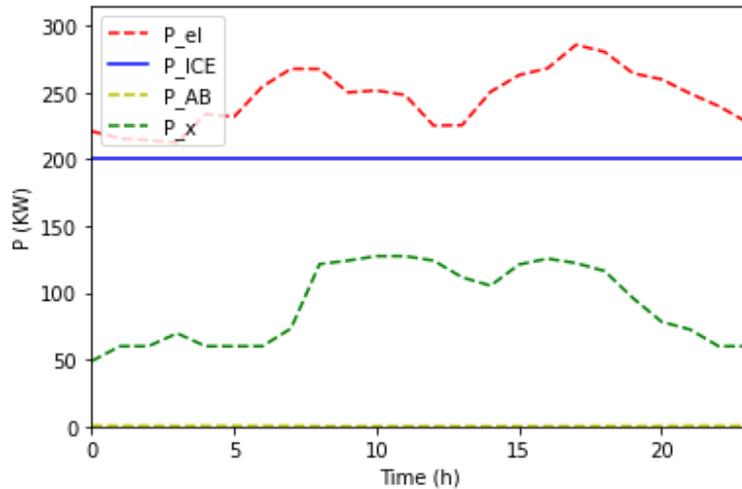
```
#Primary energy from fossil source
e_P_GR=1.95
e_GN=1.05
```

↓

Considero uno share di rinnovabili pari a circa il 20%

Risultati ottimizzazione

Es. minimizzazione costo con ICE + AB: caso maggio



Risultati ottimizzazione

Es. minimizzazione costo con ICE + AB: caso maggio

> Il valore della funzione obiettivo è = 835

> Costo = 835.02 euro

> Energia primaria = 20589 kWh

> Energia primaria da fonte fossile = 19038 kWh

> Fattore di utilizzo del motore = 100.0 %

> Fattore di utilizzo del assorbitore = 65.6 %

> Fattore di utilizzo della caldaia = 0.0 %

> Efficienza media = 51.91 %

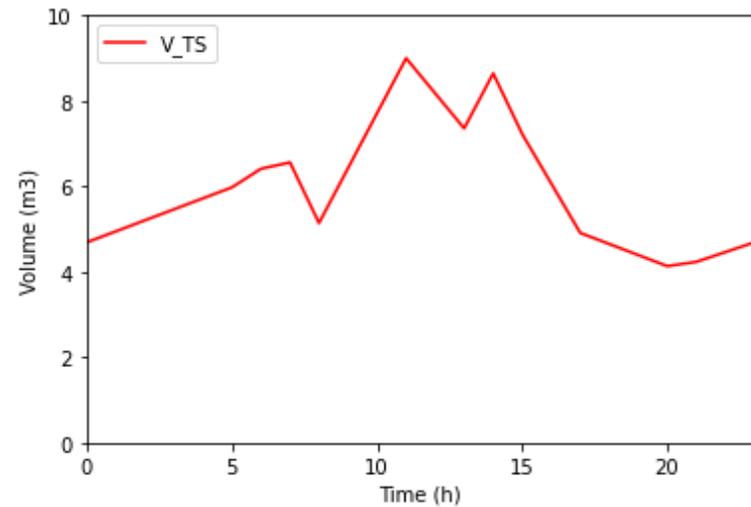
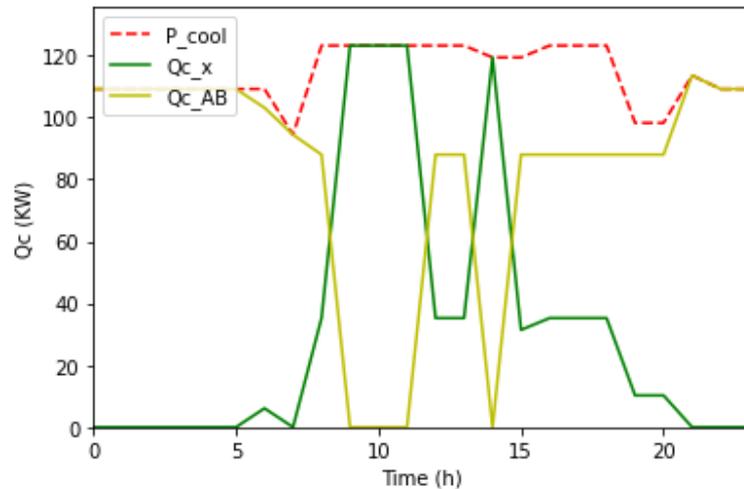
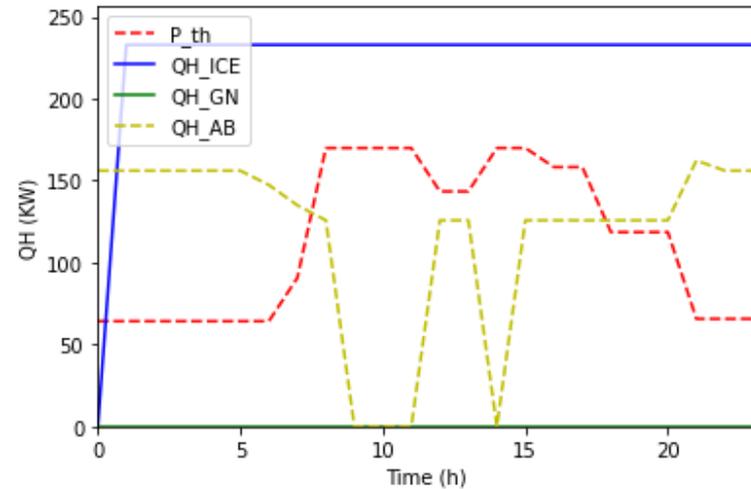
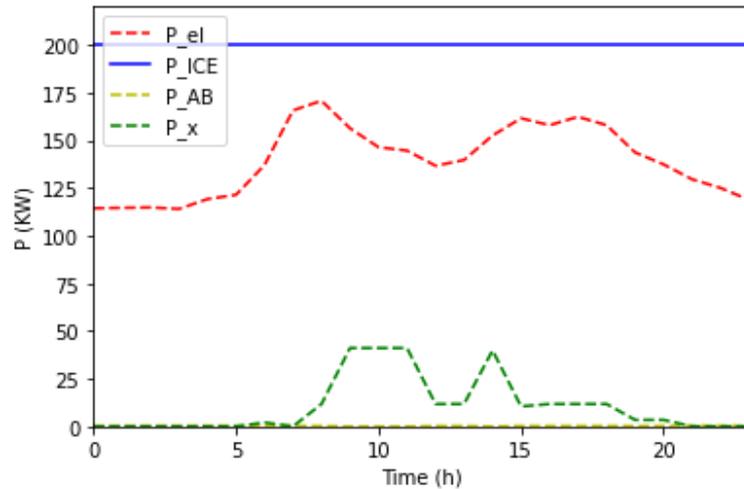
> Efficienza media caso base = 45.34 %

$$f.u.i = \frac{\int_0^{24} \varphi_i(t) dt}{24 * \varphi_{i,max}} * 100$$

$$\eta = \frac{\int_0^{24} \left(P_{el}(t) + P_{th}(t) + \frac{P_{cool}(t)}{COP_x} \right) dt}{En_{primaria}} * 100$$

Risultati ottimizzazione

Es. minimizzazione costo con ICE + AB: caso **gennaio**



Risultati ottimizzazione

Es. minimizzazione costo con ICE + AB: caso **gennaio**

> Il valore della funzione obiettivo è = 283

> Costo = 282.84 euro

> Energia primaria = 9696 kWh

> Energia primaria da fonte fossile = 10260 kWh

> Fattore di utilizzo del motore = 100.0 %

> Fattore di utilizzo del assorbitore = 46.8 %

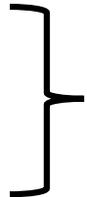


-18 % f.u. assorbitore

> Fattore di utilizzo della caldaia = 0.0 %

> Efficienza media = 71.85 %

> Efficienza media caso base = 53.02 %



+19 % efficienza complessiva

Esercitazione

```
eta_ref_el=0.525  
eta_ref_th=0.9  
p=0.925
```

```
PES=(1-sum(array_F_ICE)/(sum(array_P_ICE)/p/eta_ref_el+sum(array_QH_ICE)/eta_ref_th))
```

$$PES = \left(1 - \frac{E_{fuel}}{\frac{E_{el}}{\eta_{el,ref} \times p} + \frac{Q_{rec}}{\eta_{th,ref}}}\right) \times 100$$

Primary energy saving index

Esempio di valori di riferimento	
Rendimento elettrico $\eta_{el,ref}$	0.525
Rendimento termico $\eta_{th,ref}$	0.90
Efficienza di rete p (es. BT)	0.925

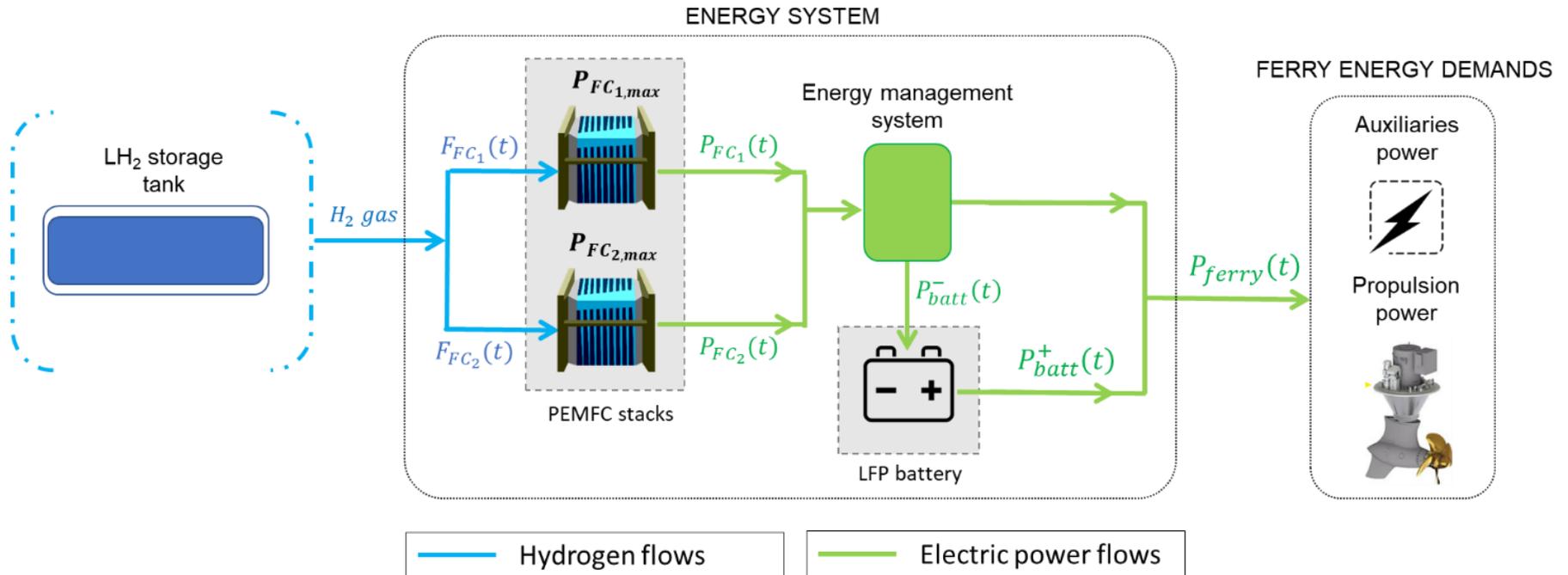
Esercitazione

1. Ottimizzare il sistema energetico con e senza assorbitore per tutte le tre funzioni obiettivo impostate
2. Calcolare valori annuali di costo, energia primaria ed energia primaria da fonte fossile (valore medio moltiplicato x 365 giorni) e valore medio PES, fattore utilizzo del motore e del frigorifero ad assorbimento
3. Commentare risultati ottenuti

	Gennaio	Marzo	Maggio	Settembre	Annuale	
Costo						
En prim						
En prim ff						
PES						
f.u. ICE						
f.u. AB						

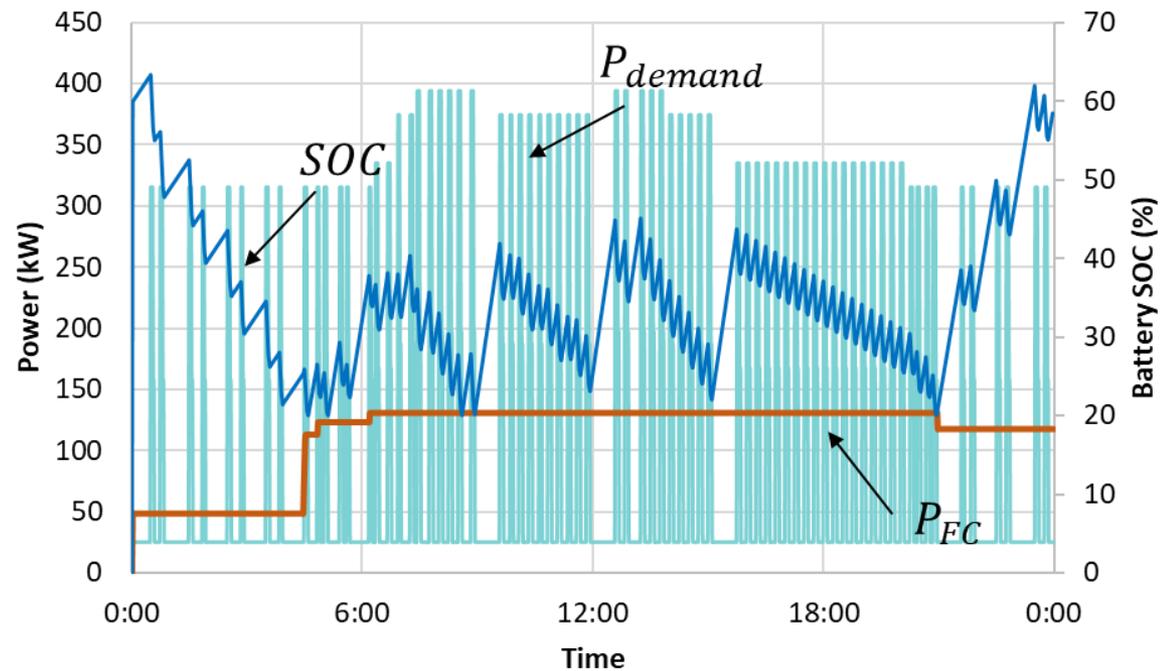
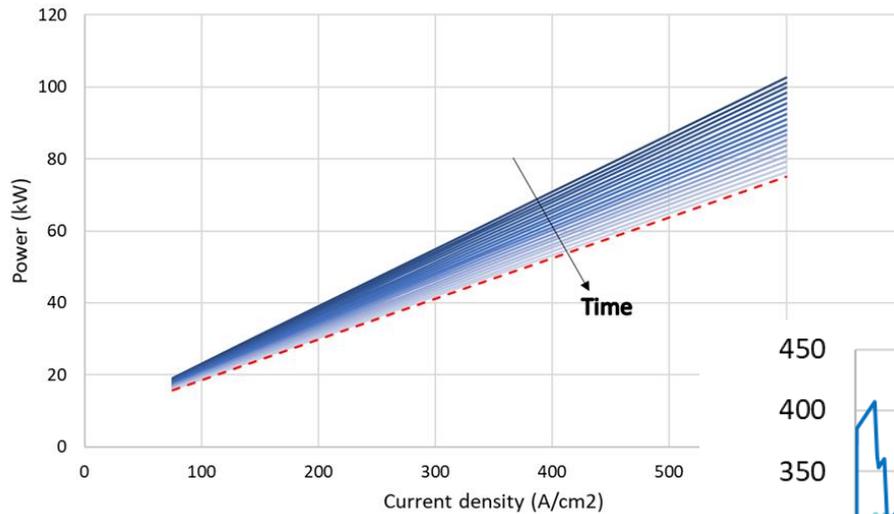
Attività ENESYS Lab

Sistemi di propulsione ibridi (PEMFC + Li-ion battery) per navi

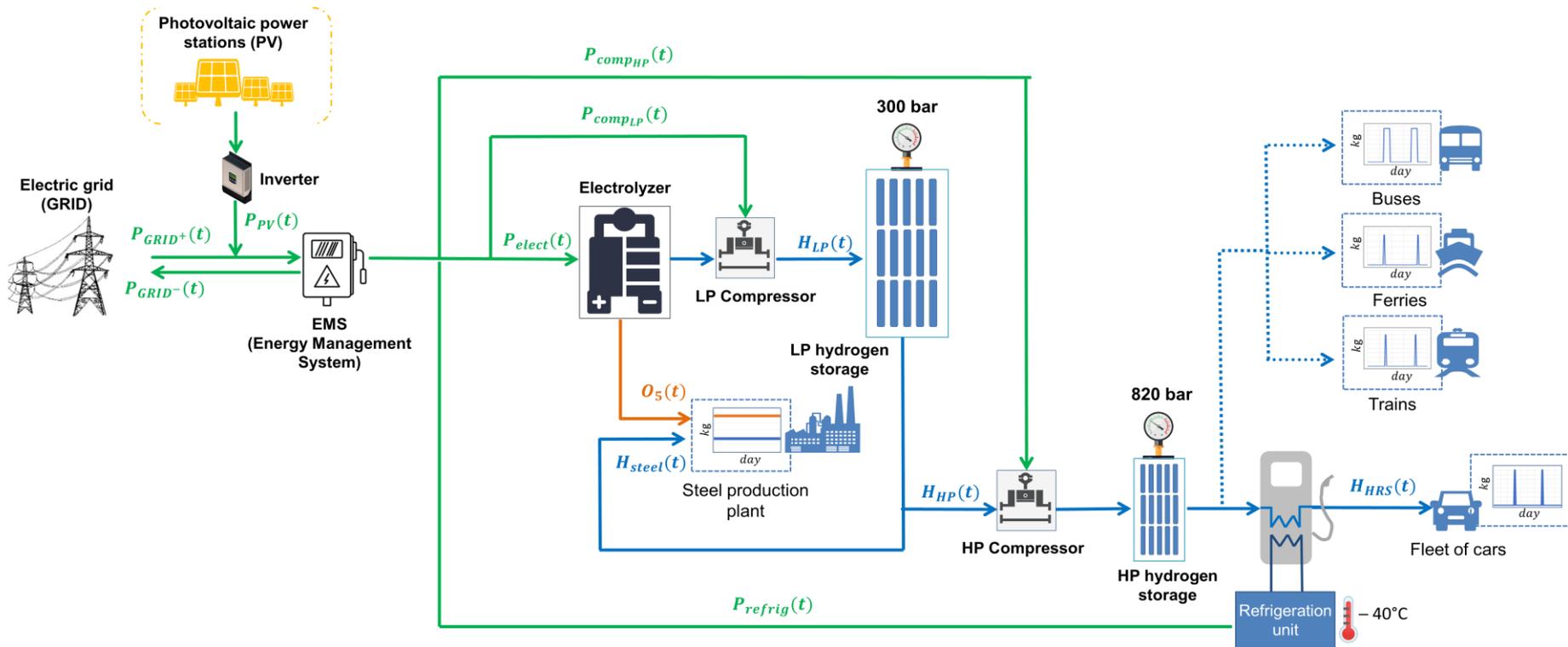


D. Pivetta, C. Dall'Armi, R. Tacani, Multi-objective optimization of hybrid PEMFC/Li-ion battery propulsion systems for small and medium size ferries, International Journal of Hydrogen Energy, 2021, <https://doi.org/10.1016/j.ijhydene.2021.02.124>.

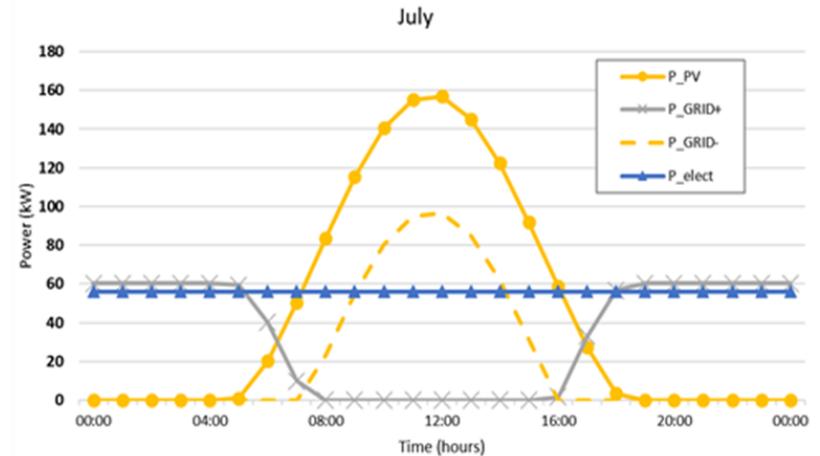
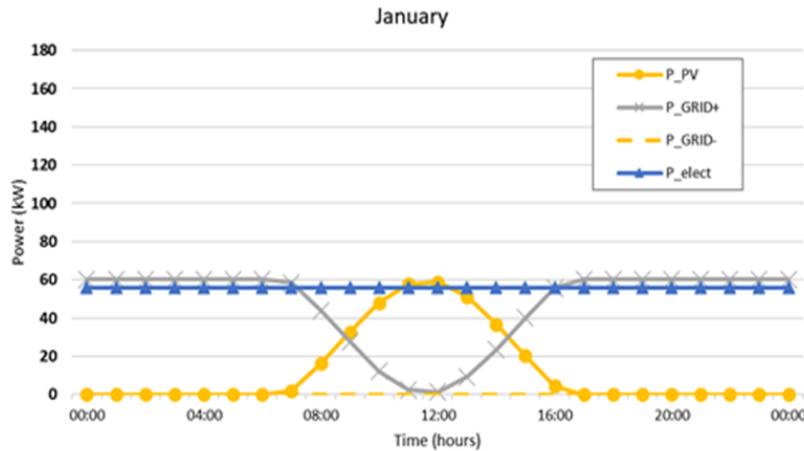
Sistemi di propulsione ibridi (PEMFC + Li-ion battery) per navi



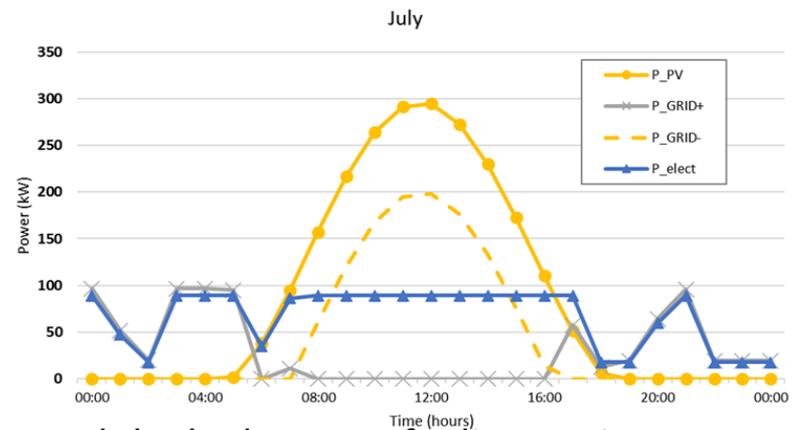
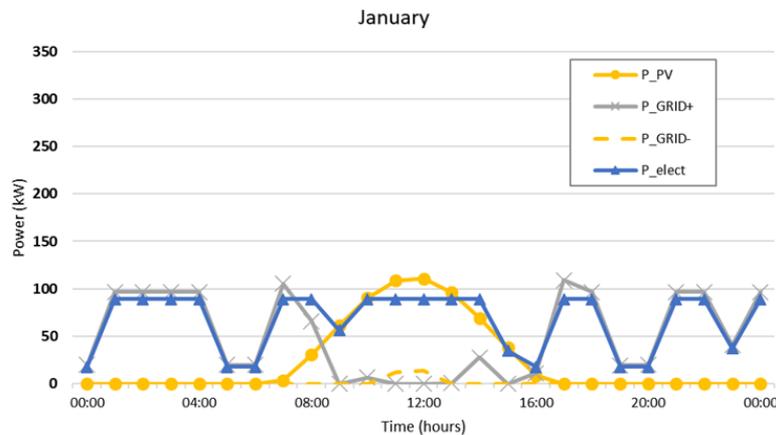
Sistemi di produzione, stoccaggio e distribuzione di idrogeno green



Sistemi di produzione, stoccaggio e distribuzione di idrogeno green



Hydrogen for the steel production plant



Hydrogen for the steel production plant and the hydrogen refueling station



Gurobi Optimization:

- <https://www.gurobi.com/documentation/9.1/refman/index.html>
(Manuale con tutte le istruzioni sulla definizione delle funzioni)
- <https://www.gurobi.com/documentation/9.1/examples/index.html>
(Esempi di modelli di ottimizzazione)
- https://www.gurobi.com/documentation/9.1/refman/multiple_objective_s.html#sec:MultipleObjectives
(Vedi per approfondire ottimizzazioni multi-obiettivo)

Riferimenti:

- Vedi cartella «Riferimenti», guida ed esempi per utilizzo approccio MILP per l'ottimizzazione di sistemi energetici

Per eventuali domande: Davide Pivetta – davide.pivetta@phd.units.it

Grazie per l'attenzione!

