



Návrh a optimalizace tepelného oběhu systému pro akumulaci energie

Bakalářská práce

Studijní program:

B2301 Strojní inženýrství

Studijní obor:

Strojní inženýrství

Autor práce:

Václav Pokorný

Vedoucí práce:

Ing. Petr Novotný, CSc.

Katedra energetických zařízení





Zadání bakalářské práce

Návrh a optimalizace tepelného oběhu systému pro akumulaci energie

Jméno a příjmení: **Václav Pokorný**
Osobní číslo: S17000107
Studijní program: B2301 Strojní inženýrství
Studijní obor: Strojní inženýrství
Zadávací katedra: Katedra energetických zařízení
Akademický rok: **2019/2020**

Zásady pro vypracování:

Tato bakalářská práce je zaměřena na návrh tepelného oběhu systému pro akumulaci energie. Zmiňovaný systém je určen ke skladování energie ve formě tepla a jeho konverzi elektrickou energií v době nedostatku energie v rozvodné síti. Tepelný oběh bude navržen s ohledem na jeho realizovatelnost a dostupnost hlavních komponent. Z hlediska ekonomie provozu bude provedena optimalizace parametrů oběhu s cílem dosáhnout co nejvyšší účinnosti.

- proveďte stručnou rešerši problematiky akumulace tepelné energie a tepelných oběhů
- na základě provedené rešerše navrhnete 3 - 5 tepelných oběhů vhodných pro účely akumulace energie
- proveďte tepelný výpočet navržených oběhů, určete jejich hlavní parametry a tepelné účinnosti
- parametry oběhů navzájem porovnejte, vyberte oběh s nejlepšími parametry a pokuste se optimalizovat jeho účinnost změnou hlavních parametrů

Rozsah grafických prací: –
Rozsah pracovní zprávy: 25 stran
Forma zpracování práce: tištěná/elektronická
Jazyk práce: Čeština



Seznam odborné literatury:

- [1] BEČVÁŘ, Josef. *Tepelné turbíny*. Praha: Nakladatelství technické literatury, 1968.
[2] KADRNOŽKA, Jaroslav. *Tepelné elektrárny a teplárny*. Praha: Nakladatelství technické literatury, 1984

Vedoucí práce: Ing. Petr Novotný, CSc.
Katedra energetických zařízení

Datum zadání práce: 1. listopadu 2019
Předpokládaný termín odevzdání: 7. července 2021

L.S.

prof. Dr. Ing. Petr Lenfeld
děkan

doc. Ing. Petra Dančová, Ph.D.
vedoucí katedry

V Liberci dne 14. února 2020

Prohlášení

Prohlašuji, že svou bakalářskou práci jsem vypracoval samostatně jako původní dílo s použitím uvedené literatury a na základě konzultací s vedoucím mé bakalářské práce a konzultantem.

Jsem si vědom toho, že na mou bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci nezasahuje do mých autorských práv užitím mé bakalářské práce pro vnitřní potřebu Technické univerzity v Liberci.

Užiji-li bakalářskou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti Technickou univerzitu v Liberci; v tomto případě má Technická univerzita v Liberci právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Současně čestně prohlašuji, že text elektronické podoby práce vložený do IS/STAG se shoduje s textem tištěné podoby práce.

Beru na vědomí, že má bakalářská práce bude zveřejněna Technickou univerzitou v Liberci v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů.

Jsem si vědom následků, které podle zákona o vysokých školách mohou vyplývat z porušení tohoto prohlášení.

28. června 2021

Václav Pokorný

Abstrakt

Tato bakalářská práce se zaměřuje na návrh a optimalizaci tepelného oběhu pro systém na nepřímou akumulaci elektrické energie. Zmiňovaný systém je určen k ukládání energie ve formě tepla a jeho zpětnou konverzi na elektrickou energii v době nedostatku elektrické energie v rozvodné síti. Tepelné oběhy byly navrženy a optimalizovány s ohledem na jejich ekonomii, účinnost, realizovatelnost a dostupnost hlavních komponent. Pro výpočet a optimalizaci byl použit výpočetní model sestavený v Pythonu. Bylo dosaženo sestavení pěti optimalizovaných tepelných oběhů s různým stupněm složitosti konstrukce, nebo různým sekundárním využitím a různou účinností.

Klíčová slova:

akumulace energie, tepelný oběh, optimalizace účinnosti

Abstract

This bachelor thesis focuses on the design and optimization of thermal circulation for an indirect electricity storage system. The mentioned system is intended for the accumulation of energy in the form of heat and its conversion into electricity in the event of a shortage of electricity in the distribution network. Thermal cycles have been designed and optimized with respect to their economy, efficiency, feasibility and availability of major components. A computational model created in Python was used for calculation and optimization. Five optimized thermal cycles with different degrees of complexity, or different secondary uses and different efficiencies were assembled.

Keywords:

energy storage, thermal circulation, efficiency optimization

Obsah

1	Úvod.....	10
2	TEORETICKÁ ČÁST.....	11
2.1	Přehled způsobů akumulace elektrické energie.....	11
2.1.1	Akumulace do chemické energie	11
2.1.2	Akumulace do kinetické energie	11
2.1.3	Akumulace do potenciální energie	12
2.1.4	Akumulace energie do tlaku.....	12
2.1.5	Akumulace energie zkapalňováním vzduchu.....	13
2.1.6	Akumulace energie do tepla	14
2.2	Oběhy.....	15
2.2.1	Rankine Clausiův (RC) - parní oběh.....	15
2.2.2	Ericsson Braytonův (EB) - plynový oběh	16
2.2.3	Paroplynový oběh.....	17
2.2.4	Carnotův oběh	17
3	PRAKTICKÁ ČÁST	18
3.1	Systém na akumulaci tepelné energie.....	18
3.2	Výchozí parametry	18
3.2.1	Volba druhu oběhu a pracovního média	18
3.2.2	Volba akumulačního média	19
3.2.3	Volba maximální pracovní teploty oběhu.....	19
3.2.4	Volba minimální pracovní teploty	19
3.2.5	Volba rozmezí použitelných tlaků	20
3.3	Opatření pro zlepšení účinnosti	20
3.3.1	Regenerace	20
3.3.2	Dělená komprese	20
3.3.3	Dělená expanze	20
3.4	Výpočetní model v Pythonu	21
3.4.1	Načtení knihoven.....	21
3.4.2	Deklarace vstupních hodnot a definice bodů a křivek	21
3.4.3	Výpočty a sloučení dat a Grafický výstup	21
3.4.4	Citlivostní analýzy.....	21
3.5	Schéma celého zařízení	22
3.6	Výchozí oběh.....	22
3.7	Optimalizace.....	24
3.7.1	Citlivostní analýza teplot.....	24

3.7.2	Citlivostní analýza tlaků.....	24
3.7.3	Citlivostní analýza tlaků s vazbou $p_3 = 2,5 * p_1$	25
3.8	Optimalizované oběhy	26
3.8.1	První optimalizovaný oběh.....	26
3.8.2	Druhý optimalizovaný oběh s využitím odpadního tepla.....	26
3.8.3	Třetí optimalizovaný oběh s dělenou kompresí	27
3.8.4	Čtvrtý optimalizovaný oběh s dělenou expanzí	28
3.8.5	Pátý optimalizovaný oběh s dělenou kompresí a expanzí.....	29
3.9	Souhrnný přehled výsledků	30
4	Závěr.....	31
5	Použitá literatura	33

Seznam Obrázků

Obrázek 1	- Spotřeba a výroba elektřiny solární elektrárnou v čase [14].....	10
Obrázek 2	- Podíl zdrojů energie na její výrobě v čase [12].....	10
Obrázek 3	- Schéma reakcí při nabíjení (vlevo) a vybíjení (vpravo) Pb akumulátoru [11]	11
Obrázek 4	- Schéma PHS [3]	12
Obrázek 5	- Schéma CAES [9]	13
Obrázek 6	- Schéma LAES	13
Obrázek 7	- Schéma jedno nádržové CB [3].....	14
Obrázek 8	- Schéma a Ts diagram pro RC oběh[9].....	15
Obrázek 9	- Schéma a Ts diagram pro EB oběh [10].....	16
Obrázek 10	- Ts diagramy pro reálný a ideální paroplynový oběh [2].....	17
Obrázek 11	- Ts a pv diagramy pro Carnotův oběh [8].....	17
Obrázek 12	- Schéma zařízení systému pro akumulaci energie do tepla	22
Obrázek 13	- Ts diagram výchozího oběhu s regenerací.....	23
Obrázek 14	- Citlivostní analýza teplot.....	24
Obrázek 15	- Citlivostní analýza tlaků.....	24
Obrázek 16	- Citlivostní analýza tlaků s vazbou $2,5x$	25
Obrázek 17	- Závislost hmotnostního průtoku na tlaku p_3	25
Obrázek 18	- Ts diagram prvního optimalizovaného oběhu	26
Obrázek 19	- Ts diagram druhého optimalizovaného oběhu.....	26
Obrázek 20	- Schéma oběhu s dělenou kompresí	27
Obrázek 21	- Ts diagram třetího optimalizovaného oběhu	27
Obrázek 22	- Schéma oběhu s dělenou expanzí.....	28
Obrázek 23	- Ts diagram oběhu s dělenou expanzí.....	28
Obrázek 24	- Schéma oběhu s dělenou kompresí a expanzí	29
Obrázek 25	- Ts diagram oběhu s dělenou kompresí a expanzí	29
Obrázek 26	- Citlivostní analýza tlaků pro oběh s regenerací a bez regenerace.....	30

Seznam tabulek

Tabulka 1	- Přehled několika zvažovaných akumulačních medií [4].....	19
Tabulka 2	- Přehled vstupních parametrů výchozího oběhu	22
Tabulka 3	- Přehled účinností výchozího oběhu	23
Tabulka 4	- Shrnutí parametrů všech oběhů.....	30

Seznam příloh

Příloha 01	Oběh s regenerací
Příloha 02	Oběh s dělenou kompresí
Příloha 03	Oběh s dělenou expanzí
Příloha 04	Oběh s dělenou kompresí a expanzí
Příloha 05	Citlivostní analýza teplot
Příloha 06	Citlivostní analýza tlaků
Příloha 07	Citlivostní analýza tlaků s vazbou 2,5x
Příloha 08	Hmotnostní průtok

Seznam zkratk a použitého značení

Bi		bismut
CAES		Compressed air energy storage – uchovávání energie do stlačeného vzduchu
CB		Carnotova baterie
CO ₂		oxid uhličitý
EB		Erickson Brigtonův oběh
K		draslík
m	[kg]	hmotnost
LAES		Liquid air energy storage - uchovávání energie do zkapalněného vzduchu
N ₂		dusík
Na		sodík
O ₂		kyslík
OZE		Obnovitelné zdroje energie
p	[Pa]	tlak
P	[W]	práce
Pb		olovo
PHS		Pumped hydroelectricity storage – přečerpávací vodní elektrárna
p _i	[MPa]	tlak v bodě i oběhu
q _{ij}	[J]	teplo odvedené nebo přivedené do nebo z oběhu mezi body oběhu i a j
q _{in}	[J]	teplo vstupující do oběhu
q _{out}	[J]	teplo vystupující z oběhu
h	[J·kg ⁻¹]	entalpie
h _i	[J·kg ⁻¹]	entalpie v bodě i oběhu
RC		Rankine Clausiův oběh
s	[J·K ⁻¹]	entropie
s _i	[J·K ⁻¹]	entropie v bodě i oběhu

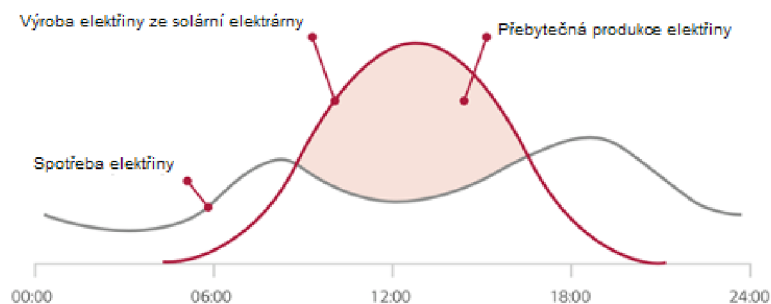
T	[K]	teplota
T_i	[K]	teplota v bodě i oběhu
t_i	[°C]	teplota v bodě i oběhu
ToP		Tower of power – věž na uchovávání potenciální energie
TES		Thermal energy storage - akumulace tepelné energie
FESS		Flywheel energy storage system - setrvačnick
v	[m ³]	objem
v_i	[m ³]	objem v bodě i oběhu
W	[J]	Práce
\$		dolar
η	[-]	účinnost oběhu
η_r	[-]	účinnost oběhu s regenerací
η_c	[-]	účinnost Carnotova oběhu pro dané teploty
η_s	[-]	odhadovaná skutečná účinnost
η_v	[-]	účinnost tepelného výměníku
η_t	[-]	účinnost expanzí turbíny
η_k	[-]	účinnost kompresoru

Poděkování

Chtěl bych poděkovat M.Sc. Magdě Vestfálové, Ph.D. mimo jiné za skvělou výuku Termodynamiky a Mechaniky tekutin. Děkuji také Ing. Ondřeji Burianovi za konzultace, výuku programování v Pythonu a pomoc při psaní této bakalářské práce.

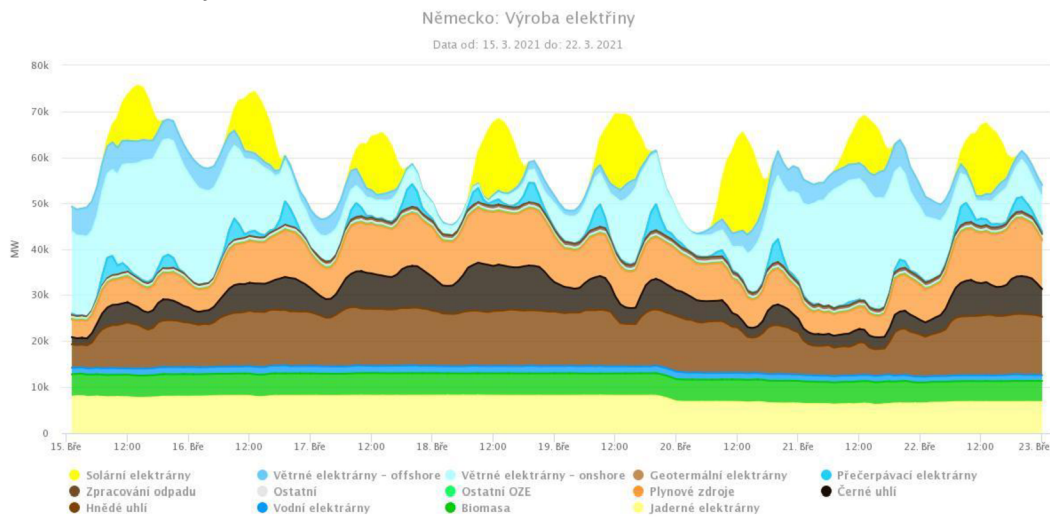
1 Úvod

Cílem této práce je navrhnout a optimalizovat tepelný oběh, který by realizoval přeměnu tepelné energie na energii elektrickou. Tento oběh by měl být součástí většího zařízení, jehož účelem je akumulovat elektrickou energii z rozvodné sítě v době jejího nadbytku a následně tuto energii vrátit do rozvodné sítě v době nedostatku energie. Důvodem pro vznik podobného zařízení je především aktuální situace s přechodem k OZE, jako jsou solární, nebo větrné elektrárny. Tyto OZE nemají stálý výkon. Odběr elektrické energie se během dne také mění. Tyto dva faktory nás přivádějí k problému, který spočívá v tom, že OZE někdy vyrábí elektrické energie víc, než se spotřebovává a jindy se spotřebovává víc, než OZE zrovna vyrábí, viz. Obrázek 1.



Obrázek 1 - Spotřeba a výroba elektřiny solární elektrárnou v čase [20]

Aktuálně je tento problém řešen primárně nárazovým použitím elektráren na fosilní paliva, což spolu s nestálým výkonem OZE můžeme vidět na datech o požitých zdrojích elektrické energie v Německu během vybraného úseku několika dní viz. Obrázek 2.



Obrázek 2 - Podíl zdrojů energie na její výrobě v čase [18]

Tento problém lze udržitelněji řešit právě akumulací energie. Elektrická energie je obtížně skladovatelná. Proto je nutné elektrickou energii přeměnit na jinou formu energie, která už skladovatelná je. Existuje tedy poptávka po zařízení, které by bylo schopné akumulovat velké množství energie a po relativně krátké době tuto energii zase relativně rychle uvolňovat.

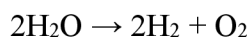
2 Teoretická část

2.1 Přehled způsobů akumulace elektrické energie

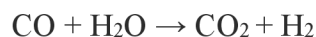
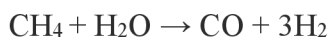
Akumulovat elektrickou energii lze přímo a nepřímo. Přímá akumulace elektrické energie se odehrává v kondenzátorech, což ale není vhodné pro dlouhodobější skladování velkého množství energie, a proto se budeme věnovat nepřímé akumulaci elektrické energie. Způsobů, jak nepřímo akumulovat elektrickou energii, je celá řada. Proto si je v této části stručně představíme a porovnáme.

2.1.1 Akumulace do chemické energie

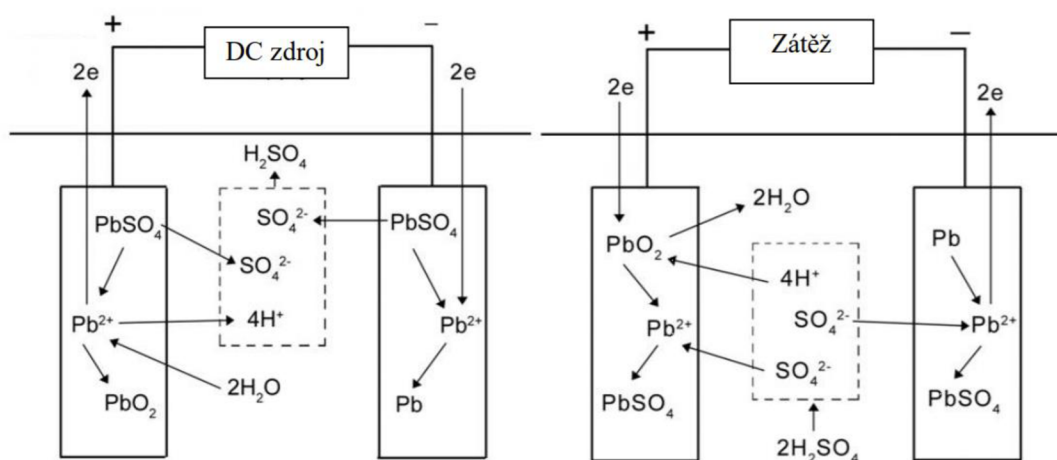
Může spočívat ve výrobě paliva, jako je výroba vodíku. Vodík lze vyrábět například pomocí elektrolýzy, což je asi nejznámější způsob výroby,



nebo také mimo jiné pomocí parního reformingu zemního plynu, což je aktuálně nejpopulárnější způsob výroby vodíku[16], protože je nejlevnější.



Další ze způsobů akumulace v této kategorii spočívá ve vratné chemické reakci, která se uskutečňuje v rámci baterie, jako jsou tužkové baterie, akumulátory, nebo Li-Ion baterie.



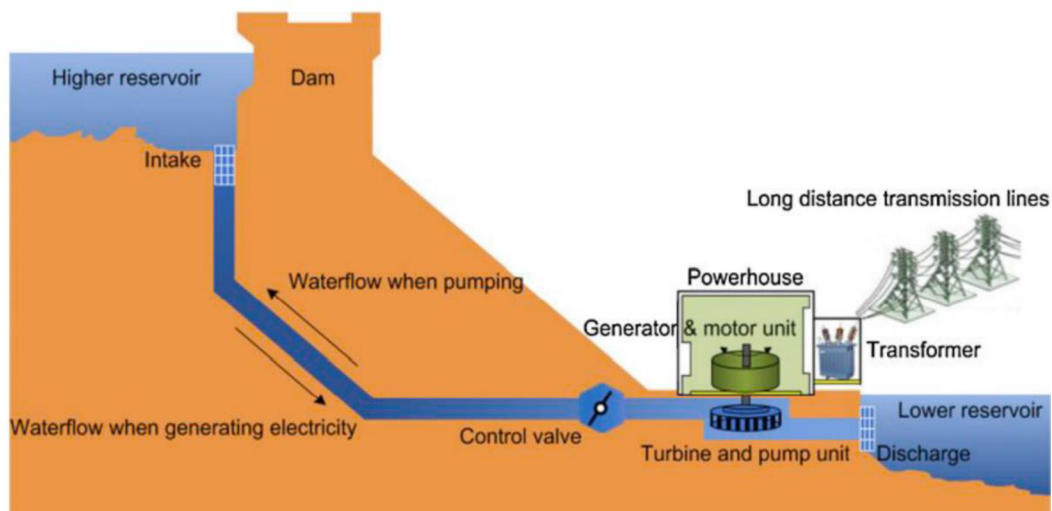
Obrázek 3 - Schéma reakcí při nabíjení (vlevo) a vybití (vpravo) Pb akumulátoru [15]

2.1.2 Akumulace do kinetické energie

Nejčastěji bude tento způsob akumulace realizován pomocí setrvačnicku (FESS). Zjednodušeně si to můžeme představit jako rotující setrvačník, kterému může být dodávána energie nebo odebírána energie, pomocí točivého elektrického stroje, který může plnit zároveň funkci motoru i generátoru. Limitujícím faktorem takového systému je velikost a pevnost onoho setrvačnicku, který nemůže přesáhnout určité otáčky, aby se neroztrhnul působením odstředivé síly.

2.1.3 Akumulace do potenciální energie

Aktuálně nejpoužívanějším systémem, spadajícím do této kategorie, a dost možná i mimo tuto kategorii, jsou přečerpávací vodní elektrárny (PHS). Mají dvě vodní nádrže horizontálně co nejbliž k sobě a vertikálně co nejdál od sebe, což výrazně omezuje lokalitu, kde je možné efektivní PHS postavit. Při přebytku elektřiny je voda hnána z dolní nádrže do nádrže horní a při nedostatku elektřiny naopak.



Obrázek 4 - Schéma PHS [3]

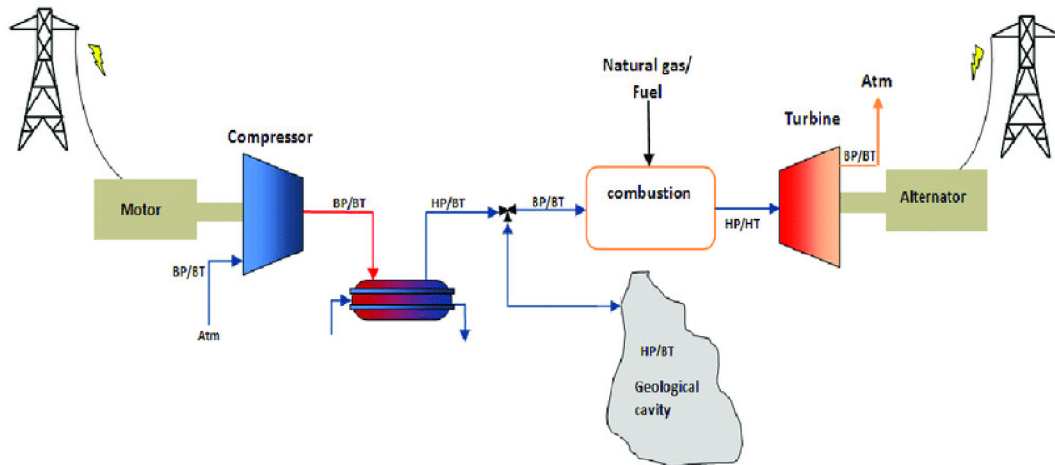
Tímto způsobem je možné rychle akumulovat velké množství energie a zároveň má tento systém relativně vysokou účinnost až 87 % [3]. Kvůli těmto výhodám také připadá přes 90 % instalované kapacity všech zařízení na nepřímou akumulaci elektrické energie právě PHS [3]. Jiné řešení hledáme proto, že tento systém nemůže být postaven všude a jeho velikost je značná. Existuje tedy poptávka po systému, který by bylo možné postavit všude a nemusel být tak velký a drahý.

Existují i další způsoby akumulace energie, které spadají do této kategorie. Spočívají ve zvedání betonových závaží do výšky (ToP) nebo spouštění závaží do opuštěných důlních šachet. Nicméně možnosti těchto dalších technologií jsou omezené.

2.1.4 Akumulace energie do tlaku

Technologie CAES spočívá zjednodušeně v tom, že při přebytku elektřiny v rozvodné síti je tato energie využita k pohonu kompresorů, které tlakují vzduch do pro to určené nádoby, která je typicky v podzemí a při nedostatku elektřiny v síti je tento tlak zase naopak použit k roztáčení turbíny a generování elektřiny.

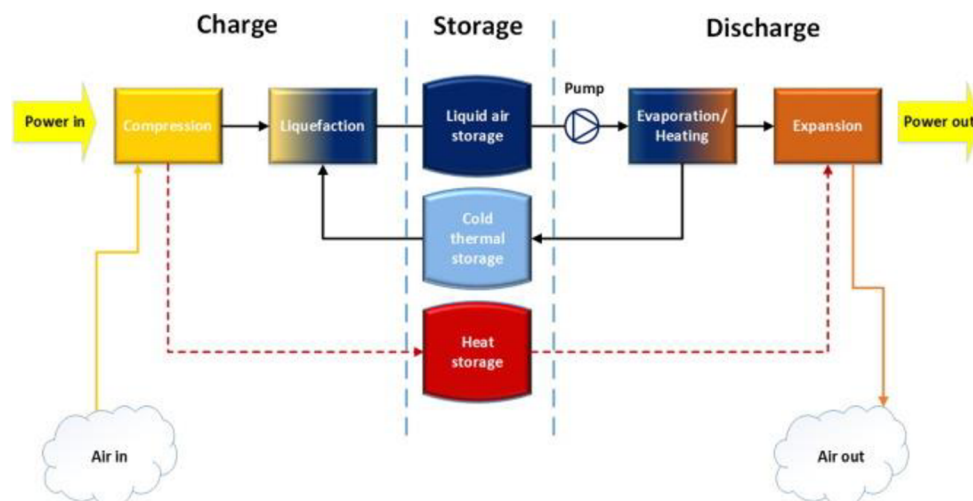
Účinnost CAES se může pohybovat mezi 40 % a 70 % [3], což je v porovnání s PHS výrazně méně. Dalším nedostatkem je to, že se při expanzi plynu musí přehřívat, k čemuž bývají používána fosilní paliva, jako je zemní plyn. Stejně jako PHS je tato technologie vázána na příhodnou lokalitu.



Obrázek 5 - Schéma CAES [11]

2.1.5 Akumulace energie zkapalňováním vzduchu

Technologie LAES funguje tak, že se za využití přebytku elektrické energie chladí a zkapalňuje vzduch, který je následně skladován v nádobách stejným způsobem, jakým je skladován při prevozu například zemní plyn. Pokud je energie naopak v rozvodné síti třeba, tak se zkapalněný vzduch ze zásobníku odebírá a vypařuje, aby následně mohl roztáčet turbínu a generovat elektrickou energii.

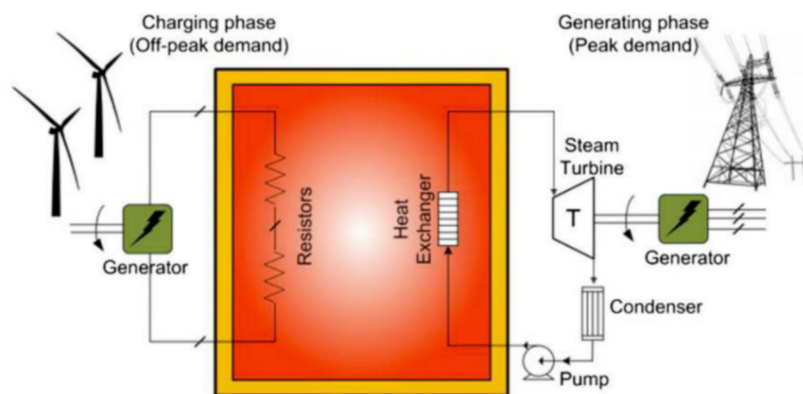


Obrázek 6 - Schéma LAES

Zásadní výhodou ve srovnání s CAES a PHS je to, že LAES není vázán na určitou vhodnou lokalitu a je možné ho realizovat prakticky kdekoli. Účinnost LAES se může pohybovat mezi 50 a 60 % [8] v případě že je použita i regenerace tepla.

2.1.6 Akumulace energie do tepla

V našem případě je tím myšlena Carnotova baterie (CB). Jedná se o systém, pro který budeme optimalizovat tepelný oběh a tím pádem mu budeme věnovat zvýšenou pozornost. Funguje tak, že se pomocí elektrické energie, které je přebytek, ohřívá na teploty co možná nejvzdálenější teplotě okolí. Toto médium je pak skladováno v dobře izolovaném zásobníku. Když je energie nedostatek, tento tepelný rozdíl se postupně vyrovnává pomocí tepelného oběhu, kterému je dodávána energie od akumulárního média. Tepelný oběh díky tomuto rozdílu teplot může pracovat a generovat mechanickou práci, která je dále pomocí generátoru měněna zpět na elektrickou energii.



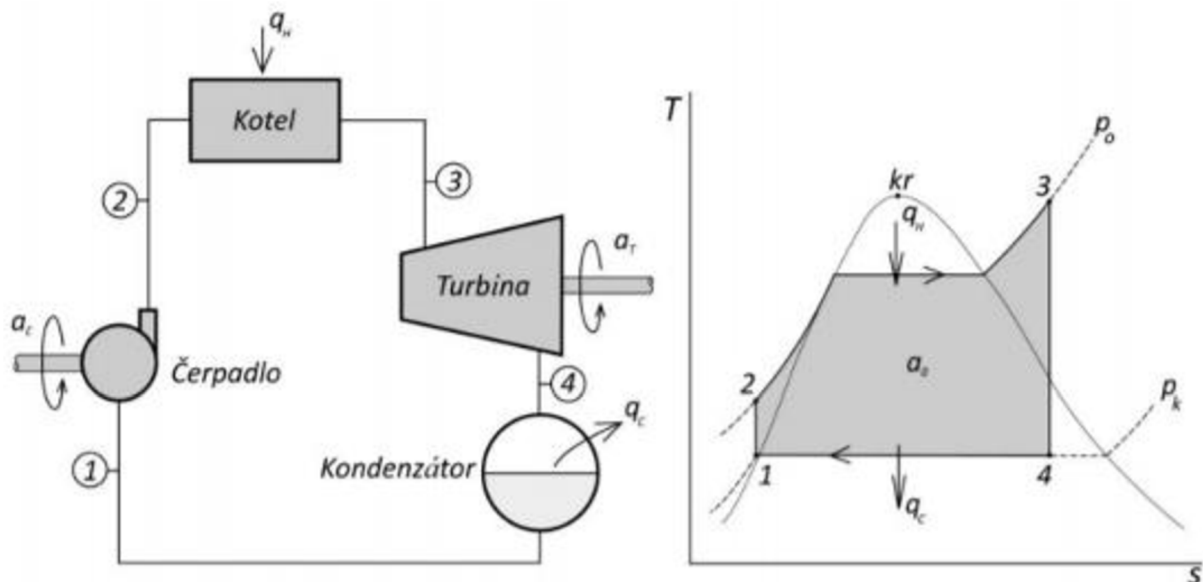
Obrázek 7 - Schéma jedno nádržové CB [3]

Účinnost Carnotovy baterie se může pohybovat mezi 30 a 60 % [3], což je oproti PHS výrazně nižší účinnost, na druhou stranu může být tento systém použit prakticky kdekoliv a může být výrazně menší a levnější než PHS. Mimo to tato technologie dosáhla nejvyššího hodnocení v analýze energetické bezpečnosti [3]. Řešením potřeby nepřímé akumulace elektrické energie pravděpodobně nebude jedna technologie, ale kombinace několika technologií a Carnotova baterie má potenciál být vedle PHS jednou z těchto technologií.

2.2 Oběhy

2.2.1 Rankine Clausiův (RC) - parní oběh

Rankin Clausiův, znám také jako parní, oběh je v energetice asi nejpoužívanější tepelný oběh, díky své relativně vysoké účinnosti. Skládá se ze čtyř základních součástí z čerpadla, kotle, turbíny a kondenzátoru. Čerpadlo zvyšuje tlak. Kotel dodává teplo médiu, a to prochází změnou skupenství z kapaliny na páru. Turbína koná mechanickou práci. Kondenzátor mění skupenství média z páry na kapalinu a odvádí odpadní teplo. Pro tento oběh je charakteristické, že zde médium prochází skupenskou změnou z plynu na kapalinu a zpět z kapaliny na plyn. Nejčastěji se jako médium používá pára, lze ale použít i jiná media, například směsi uhlovodíků. Používá se typicky v tepelných elektrárnách. Účinnost lze vypočítat jako $\eta = \frac{q_{in} - q_{out}}{q_{in}}$

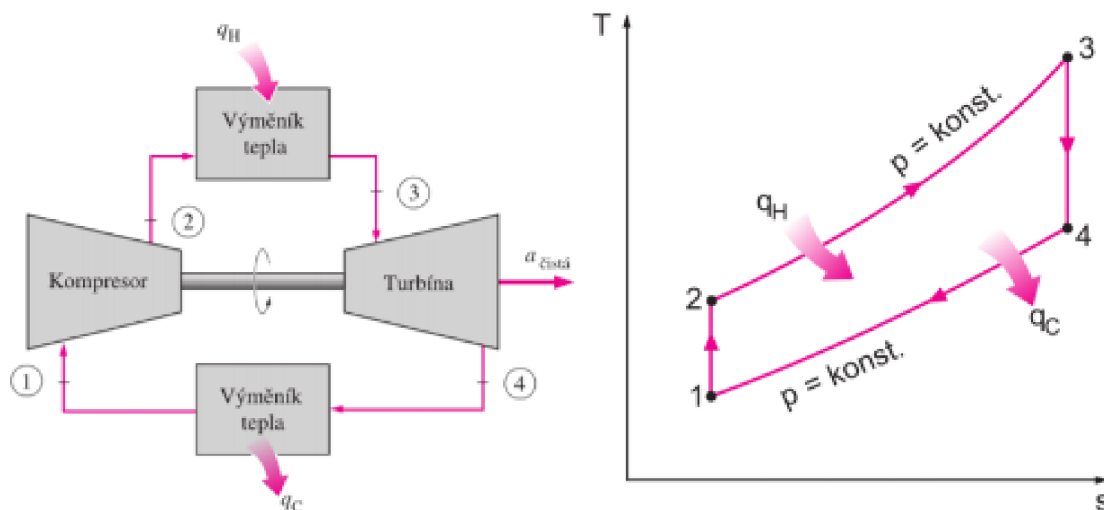


Obrázek 8 - Schéma a Ts diagram pro RC oběh[13]

Hlavní výhodou je, že tento oběh dosahuje vyšších účinností. Další méně podstatnou výhodou může být fakt, že zkondenzované médium zaujímá výrazně menší objem, než by zaujímal plyn a čerpadlo může být výrazně menší, než by musel být kompresor u jiného řešení. Hlavní nevýhodou je zase jeho vyšší složitost, cena a problémy, které vznikají, pokud oběh nepracuje neustále.

2.2.2 Ericsson Braytonův (EB) - plynový oběh

Ericsson Braytonův, znám také jako plynový, oběh je jedním z nejpoužívanějších tepelných oběhů. Může být otevřený, nebo uzavřený, přičemž obvyklejší je otevřený, nicméně my se budeme věnovat uzavřenému oběhu. Oběh se skládá ze čtyř základních součástí a to kompresoru, ohříváku, turbíny a tepelného výměníku, který odvádí odpadní teplo. Kompresor zvyšuje tlak. Ohřívák dodává plynu teplo. Turbína koná mechanickou práci. Tepelný výměník odvádí odpadní teplo. Pro tento oběh je charakteristické, že médium, které se používá, neprochází změnou skupenství. Typicky se používá pro plynové spalovací, nebo plynové expanzní turbíny. Účinnost lze vypočítat jako $\eta = \frac{q_{in} - q_{out}}{q_{in}}$

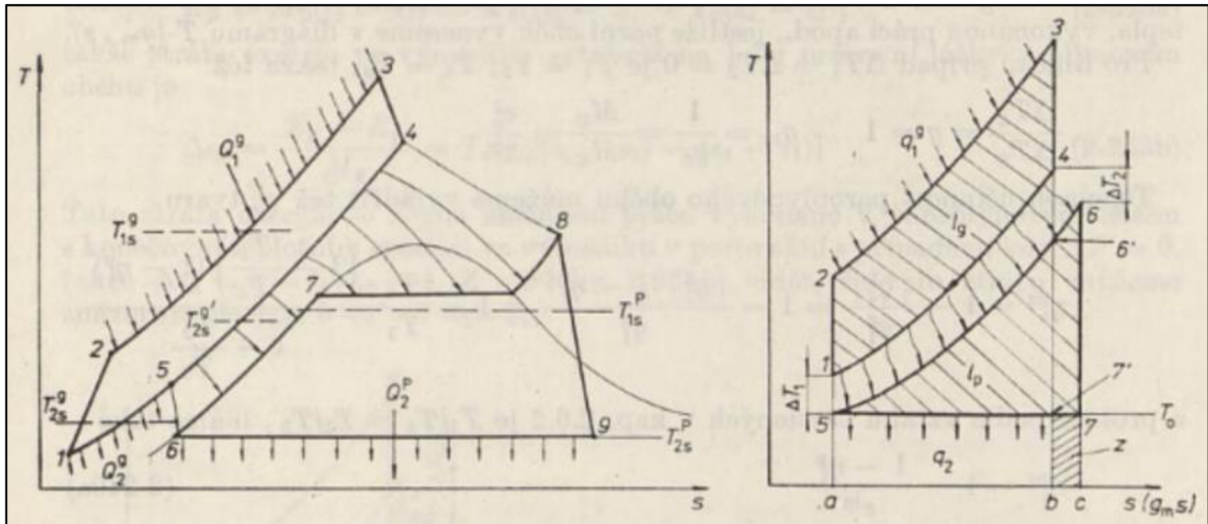


Obrázek 9 - Schéma a Ts diagram pro EB oběh [14]

Hlavní výhodou je jednoduchost konstrukce a možnost bezproblémového přerušování chodu oběhu. Hlavní nevýhodou je nižší účinnost. Další menší nevýhodou je to, že plyn zaujímá větší objem než voda a kompresor musí být výrazně větší než čerpadlo u alternativního řešení.

2.2.3 Paroplynový oběh

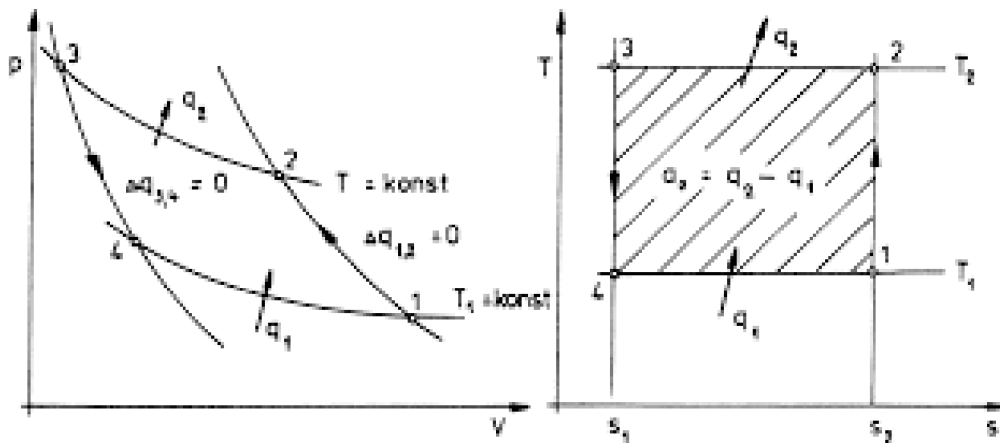
Paroplynový oběh vlastně není jeden oběh, ale dva na sebe navazující oběhy. Odpadní teplo z plynového oběhu (EB) je použito v parním oběhu (RC). Toto řešení umožňuje zvýšení účinnosti, za cenu výrazného zvýšení složitosti. $\eta = \frac{q_{in} - q_{out}}{q_{in}}$



Obrázek 10 - Ts diagramy pro reálný a ideální paroplynový oběh [2]

2.2.4 Carnotův oběh

Carnotův oběh je vratný kruhový děj hypotetického ideálního tepelného stroje, který se skládá ze čtyř fází, a to izotermické expanze, adiabatické expanze, izotermické komprese, adiabatické komprese. Tento hypotetický ideální oběh není v praxi realizovatelný, ale používá se pro porovnání s reálnými oběhy, jelikož žádný reálný oběh pracující mezi stejnými mezními teplotami nemůže dosáhnout větší účinnosti než Carnotův oběh. Účinnost lze vypočítat dle vzorce $\eta_c = 1 - \frac{T_1}{T_2}$



Obrázek 11 - Ts a pv diagramy pro Carnotův oběh [12]

3 Praktická část

3.1 Systém na akumulaci tepelné energie

Celý systém pracuje tak, že v době, kdy je v síti elektrické energie přebytek, využívá elektrické energie k ohřátí média. Toto ohřáté medium je skladováno v dobře izolované nádobě. Tepelná energie media je tedy takto uchována. V době nedostatku elektrické energie v síti je naopak provedená zpětná přeměna tepelné energie media na energii elektrickou, pomocí tepelného oběhu, který navrhujeme. Existují jedno nádržové a dvou nádržové řešení. S jedno nádržovými je ale pro naše potřeby ta potíž, že při vybíjení klesá teplota akumulčního média, a tedy není, mimo jiné problémy, dost dobře možné navrhnout optimální oběh. I proto se v našem případě bude jednat o dvou nádržový systém, který je schopen zajistit přísun tepla za konstantní teploty.

3.2 Výchozí parametry

Abychom mohli navrhnout tepelný oběh, který budeme následně optimalizovat, musíme si nejdřív stanovit určité výchozí parametry, jako je druh oběhu a pracovní medium, akumulční medium, rozmezí teplot a tlaků, ve kterých se můžeme pohybovat apod.

3.2.1 Volba druhu oběhu a pracovního média

Jelikož z principu fungování celého zařízení vyplývá, že oběh nebude pracovat neustále, ale bude se naopak často zapínat a vypínat, pravděpodobně v osmihodinových cyklech, takže použít RC, nebo paroplynový oběh, by bylo problematické, vzhledem k nutnosti udržování systému v tzv. Horké rezervě. Mimo to se při použití RC oběhu musí řešit odvádění kondenzátu a vzniká také problém se zavzdušňováním. Zvolíme tedy raději oběh EB, který sice nedosahuje na tak vysoké účinnosti jako RC, ale jeho konstrukce je jednodušší, menší a levnější. Jako nejvhodnější pracovní medium se jeví CO₂, protože existuje snaha tento, v atmosféře nežádoucí plyn, využít a je tedy i relativně levný.

3.2.2 Volba akumulčního média

Mezi důležité atributy akumulčního média patří maximální a minimální pracovní teplota, tepelná kapacita, vodivost, hustota a nízká cena. Na základě těchto parametrů, především s ohledem na nízkou cenu a vhodný rozsah použitelných teplot, byla z dostupných alternativ zvolena solární sůl viz. možnost č. 2 v tabulce 1.

Tabulka 1 - Přehled několika zvažovaných akumulčních medií [4]

č.	Akumulční medium	Bod tání [°C]	Limit stability [°C]	Tepelná kapacita [KJ/kgK]	Tepelná vodivost [W/mK]	Cena [\$/kg]
1	Minerální olej	-20	300	-	0,1	0,3
2	Solární sůl	220	600	1,1	0,55	0,5
3	sandia mix	95	500	1,3	0,65	0,7
4	Hitec	142	535	1,56	0,2	0,93
5	Hitec XL	120	500	1,45	0,52	1,1
6	Na	98	883	1,25	46	2
7	NaK	-12	785	0,87	26	2
8	Silikonový olej	-20	400	-	0,1	5
9	PbBi	125	1533	0,15	13	13

3.2.3 Volba maximální pracovní teploty oběhu

Na volbu maximální teploty našeho oběhu má vliv několik faktorů. Kvůli účinnosti bychom chtěli maximální teplotu co nejvyšší. Námí zvolené akumulční medium nám ale nedovolí jít nad 600 °C. Dále potřebujeme určitý tepelný spád, aby mohl tepelný výměník správně fungovat. A konečně, protože nechceme kvůli tepelnému namáhání využívat speciální vysoce odolné, ale také drahé materiály a součástky, tak se omezíme na teplotu do 550 °C, což je i maximální teplota, pro které jsme našli dostupné turbíny, konkrétně jde o expanzní turbíny ETG [17].

3.2.4 Volba minimální pracovní teploty

Volba minimální teploty našeho oběhu je výrazně jednodušší než volba maximální teploty, jelikož zde nehraje roli tolik faktorů. V podstatě jediné faktory, ke kterým budeme přihlížet jsou teplota okolí, do kterého budeme z oběhu odvádět odpadní teplo a tepelný spád, který bude třeba, aby tepelný výměník fungoval správně. Teplotu okolí budeme uvažovat 30 °C a potřebný tepelný spád pro chlazení budeme uvažovat 20 °C. Minimální teplotu oběhu tedy stanovíme na 50 °C.

3.2.5 Volba rozmezí použitelných tlaků

Naše pracovní médium jsme si zvolili pod-kritické CO₂, protože nadkritické CO₂ by sice mohlo dosahovat lepší účinnosti, ale nároky na komponenty a složitost celého návrhu by rapidně stoupla. Kritický tlak CO₂ je 7,38 MPa, budeme se tedy nutně pohybovat pod touto hodnotou. Dále se musíme ohlížet na namáhání použitých součástí tlakem a obecně technické možnosti, ideálně s ohledem na dostupnost komponentů. Již dříve zmíněné expanzní turbíny ETG[17] mají, dle výrobce, maximální povolený tlak na vstupu 6,5 MPa a 2,5 MPa na výstupu, čehož se tedy budeme držet.

3.3 Opatření pro zlepšení účinnosti

Pokud chceme zvednout účinnost tepelného stroje můžeme toho dosáhnout snížením minimální teploty, nebo zvýšením maximální teploty, dále pak optimalizací pracovních tlaků, využitím regenerace, dělenou kompresí, nebo dělenou expanzí.

3.3.1 Regenerace

Regenerace je označení ohřevu média pomocí části odpadního tepla tam, kde to tepelný spád umožňuje. Technická realizace je není až tak komplikovaná, mezi potrubím za turbínou a za kompresorem vhodně umístíme teplený výměník. Jedná se tedy o zlepšení účinnosti za cenu ne až tak závratného navýšení složitosti.

3.3.2 Dělená komprese

Dělená, nebo také více (2, 4, 6) stupňová, komprese znamená přidání dalších stupňů kompresoru a mezichlazení do oběhu. První stupeň komprese zvýší tlak jen částečně, poté je pracovní médium ochlazen na nižší teplotu a teprve další stupeň komprese (v případě dvou stupňové komprese) zvedne tlak až na požadovaný tlak po kompresi.

3.3.3 Dělená expanze

Dělená, nebo také více stupňová, komprese znamená přidání dalších turbín a přihřívání do oběhu. První turbína sníží tlak jen částečně, poté je pracovní médium ohřáto na vyšší teplotu a teprve potom další turbína (v případě dvoustupňové expanze) sníží tlak až na požadovaný tlak po expanzi.

3.4 Výpočetní model v Pythonu

Některé grafické výstupy a naprostá většina výpočtů pro tuto práci byla provedena v programovacím jazyce Python za použití rozhraní, jako je Anaconda a Spyder. Anaconda byla použita ke stažení potřebných knihoven a pracovního rozhraní Spyder. Spyder byl následně použit k samotné tvorbě výpočetního modelu, který se skládá z 8 skriptů o celkové délce přibližně 1600 řádků kódu.

3.4.1 Načtení knihoven

Nejprve je vždy třeba načíst knihovny funkcí, které budeme používat. V této práci byla použita k matematickým funkcím a práci s maticemi, knihovna NumPy, k práci s grafickým výstupem knihovnu Matplotlib a k získání dalších veličin popisujících stav média na základě dvou známých veličin knihovnu CoolProp.

3.4.2 Deklarace vstupních hodnot a definice bodů a křivek

Po načtení knihoven, následuje sekce, kde deklarujeme všechny vstupní hodnoty, jak ty základní, kterými jsou teploty a tlaky na vstupech do kompresoru a turbíny, tak i ty vedlejší jako jsou různé koeficienty, například pro udání jemnosti kroků všech cyklů používajících nějaký krok. Poté už nic nevolíme, pouze odvozujeme další body na základě zvolených vstupů do výpočetního modelu. K tomu hojně používáme knihovnu CoolProp, která nám jedním příkazem umožňuje najít například entropii na základě známé teploty a tlaku, nebo naopak teplotu na základě známé entropie a tlaku. Například pro body 2 a 4 známe entropie a tlaky, protože, za předpokladu dokonalé komprese a dokonalé expanze, odpovídají v případě bodu 2 entropie bodu 1 a tlak bodu 3 a u bodu 4 naopak. Křivky tyto body spojující pak lze definovat obdobně, ale za využití matic a cyklů. Jednoduše naplňujeme matici řadou bodů pomocí For cyklu s krokem jemnosti odpovídající na začátku kódu zvolenému koeficientu.

3.4.3 Výpočty a sloučení dat a Grafický výstup

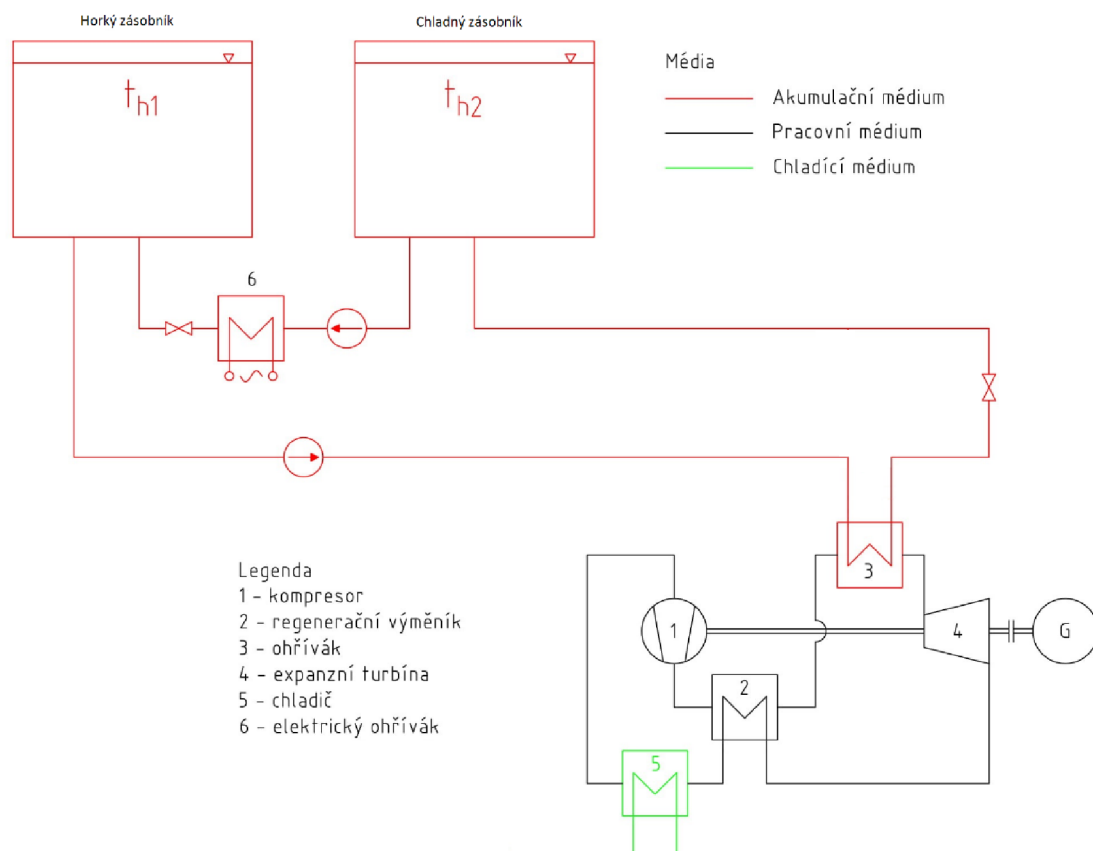
Když už máme všechny hodnoty, body a křivky provedeme výpočty, jako například výpočty účinností. Dále je ale ještě třeba jednotlivé části dat pro celý oběh sloučit do jedné matice, aby se pak snáze pracovalo s grafickým výstupem. Když už máme vše připravené stačí použít několik příkazů z knihovny Matplotlib a vygenerovat grafické výstupy Ts diagramů.

3.4.4 Citlivostní analýzy

Celý kód pro jeden oběh pak definujeme jako funkci a pomocí For cyklu budeme do této funkce postupně dosazovat různé hodnoty a ukládat výstupní hodnoty účinností, z čehož pak můžeme sestavit grafické výstupy citlivostních analýz.

3.5 Schéma celého zařízení

Celé zařízení má 3 okruhy, akumulční okruh znázorněný červeně, pracovní oběh znázorněný černě a chladicí okruh znázorněný zeleně. Tato bakalářská práce se primárně zaměřuje na návrh a optimalizaci pracovního oběhu. Jako pracovní oběh jsme zvolili EB s využitím pod-kritického CO₂. Co je ale ještě třeba dodat je to, že jsme se rozhodli v každém případě využít konstrukční řešení s využitím regenerace, protože se jedná o největší zlepšení účinnosti v porovnání s dostupnými alternativami a má i nejlepší poměr zlepšení účinnosti ku navýšení složitosti.



Obrázek 12 - Schéma zařízení systému pro akumulaci energie do tepla

3.6 Výchozí oběh

Základní zvolené vstupní parametry do výpočetního modelu můžeme vidět v tabulce 2.

Tabulka 2 - Přehled vstupních parametrů výchozího oběhu

t_1	Teplota na vstupu do kompresoru	50 °C
t_3	Teplota na vstupu do turbíny	550 °C
p_1	Tlak na vstupu do kompresoru	2,5 MPa
p_3	Tlak na vstupu do turbíny	6,5 MPa

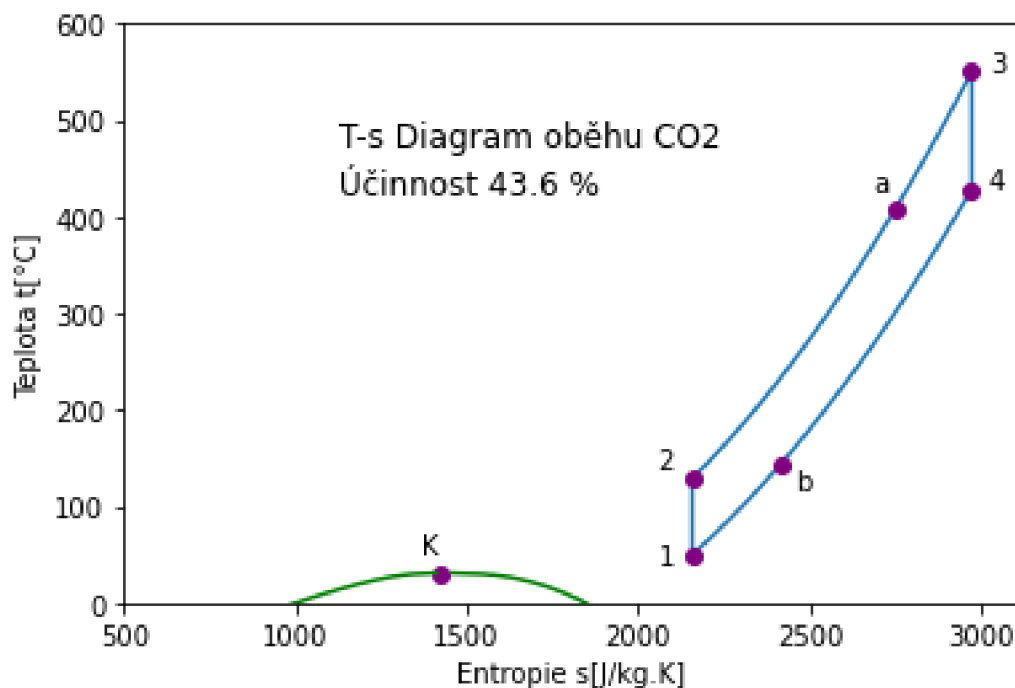
Dále do diagramu zaneseme pomocné body A a B, které nám poslouží při zanesení regenerace do našeho modelu a při výpočtu účinnosti oběhu s regenerací. A vypočteme účinnosti pro případ oběhu s regenerací, bez regenerace a Carnotova cyklu mezi stejnými mezními teplotami. Počítat budeme s tepelným spádem pro regenerační výměník $T_m = 15 \text{ °C}$ a s účinností regeneračního výměníku $\eta_v = 98 \%$.

- Bod B najdeme na izobaře p1 a jeho teplota bude rovna $t_b = t_2 + T_m$
- Bod A najdeme na izobaře p3 a jeho entalpie bude rovna $h_a = h_2 - (h_b - h_4) * \eta_v^{-1}$
- Účinnost Carnotova oběhu pro stejné mezní teploty bude rovna $\eta_c = 1 - \frac{t_1}{t_3}$
- Účinnost oběhu s regenerací bude rovna $\eta_r = \frac{q_{a3} + q_{b1}}{q_{a3}} = \frac{h_3 - h_a + h_1 - h_b}{h_3 - h_a}$
- Účinnost oběhu bez regenerace bude rovna $\eta = \frac{q_{23} + q_{41}}{q_{23}} = \frac{h_3 - h_2 + h_1 - h_4}{h_3 - h_2}$

Tabulka 3 - Přehled účinností výchozího oběhu

η	Účinnost oběhu	16,59 %
η_r	Účinnost s regenerací	43,62 %
η_c	Účinnost Carnotova oběhu	60,74 %

Náš výchozí oběh s regenerací bude tedy v Ts diagramu vypadat takto.



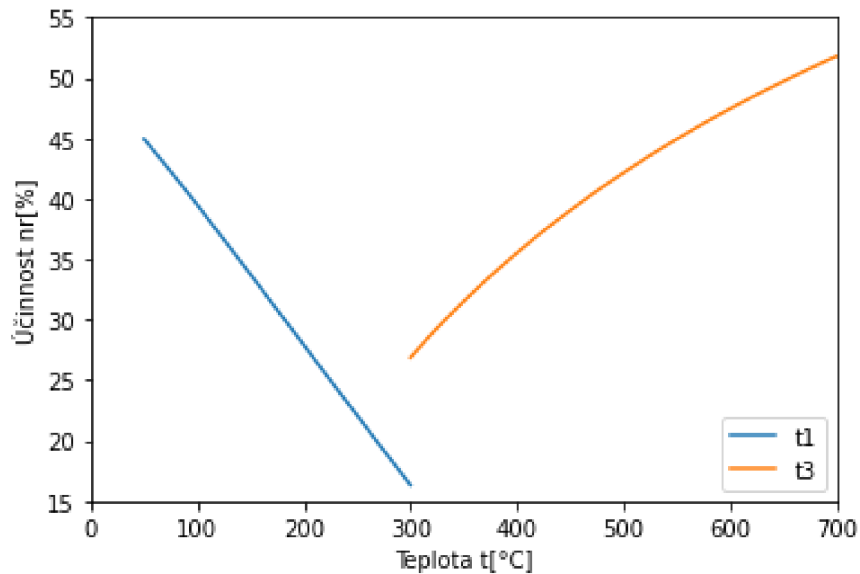
Obrázek 13 - Ts diagram výchozího oběhu s regenerací

Stejný postup budeme během optimalizace opakovat pro jiné vstupní hodnoty t_1 , p_1 , t_3 a p_3 .

3.7 Optimalizace

3.7.1 Citlivostní analýza teplot

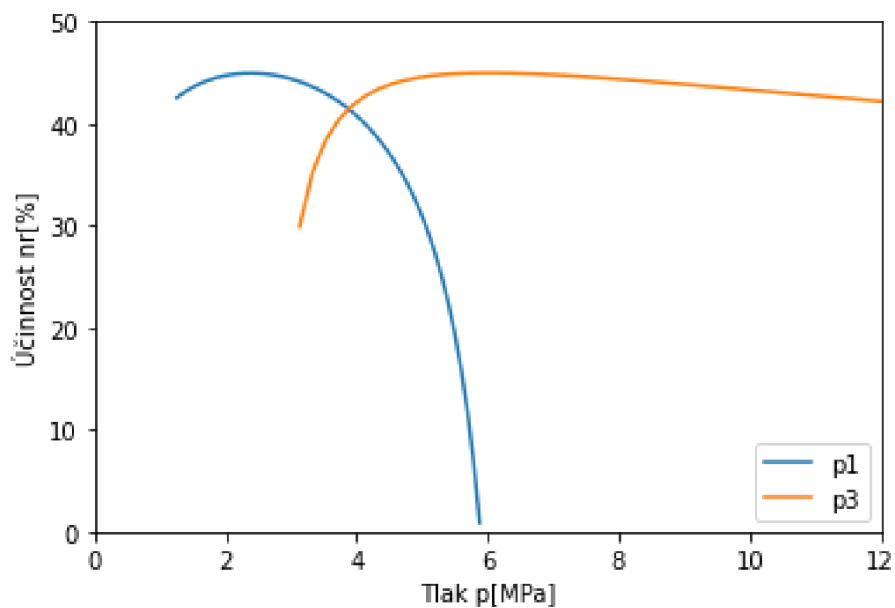
Abychom si ověřili náš předpoklad, že čím nižší je minimální teplota t_1 a čím vyšší je maximální teplota t_3 , tím vyšší účinnosti dosáhneme, a jak moc to účinnost ovlivní, provedeme citlivostní analýzu pro teploty viz. 14.



Obrázek 14 - Citlivostní analýza teplot

3.7.2 Citlivostní analýza tlaků

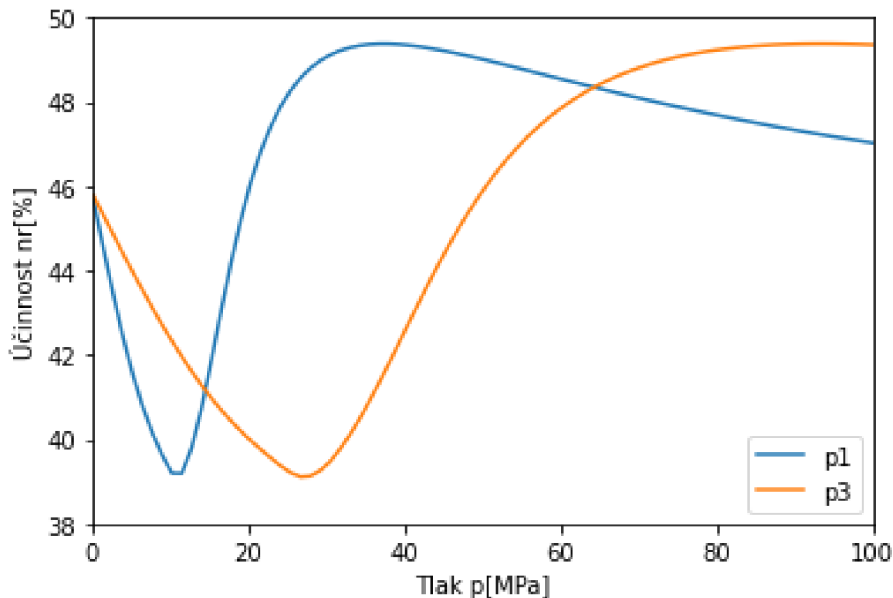
Provedeme citlivostní analýzu tlaků tak, že 3 ze 4 vstupních parametrů teplot a tlaků necháme konstantní a budeme měnit jeden z tlaků. Nejprve p_1 a poté i p_3 a budeme sledovat vliv těchto změn na účinnost oběhu s regenerací.



Obrázek 15 - Citlivostní analýza tlaků

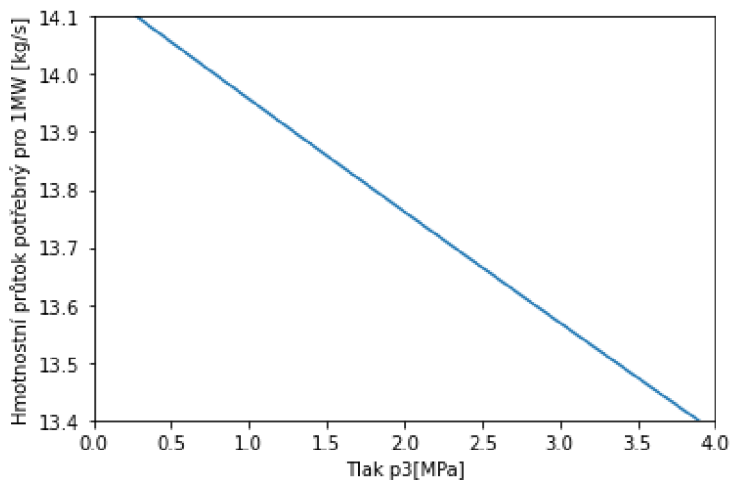
3.7.3 Citlivostní analýza tlaků s vazbou $p_3 = 2,5 * p_1$

Po provedení několika citlivostních analýz tlaků, pro různé oblasti hodnot, byl zpozorován opakující se vzor toho, že nejlepší účinnosti je dosahováno, pokud je p_3 určitým násobkem p_1 . Jako nejvhodnější násobek se jeví 2,5. Zkusíme tedy měnit tlaky a sledovat účinnost při dodržení vazby $p_3 = 2,5 * p_1$.



Obrázek 16 - Citlivostní analýza tlaků s vazbou 2,5x

Z této analýzy vyplývá, že z hlediska účinnosti by byl optimálním maximální tlak okolo 90 MPa, což je ale mimo naše stanovené praktické maximum pro technickou proveditelnost 6,5 MPa a pravděpodobně i mimo technické možnosti jako takové. S tímto omezením maximálního tlaku nám tato analýza ukazuje, že pokud chceme zvýšit účinnost, musíme tlaky naopak snížit, čímž se sníží i namáhání součástí našeho oběhu, ale také se nám bohužel zvedne potřebný hmotnostní průtok pro dosažení stejného výkonu viz. obrázek 17.

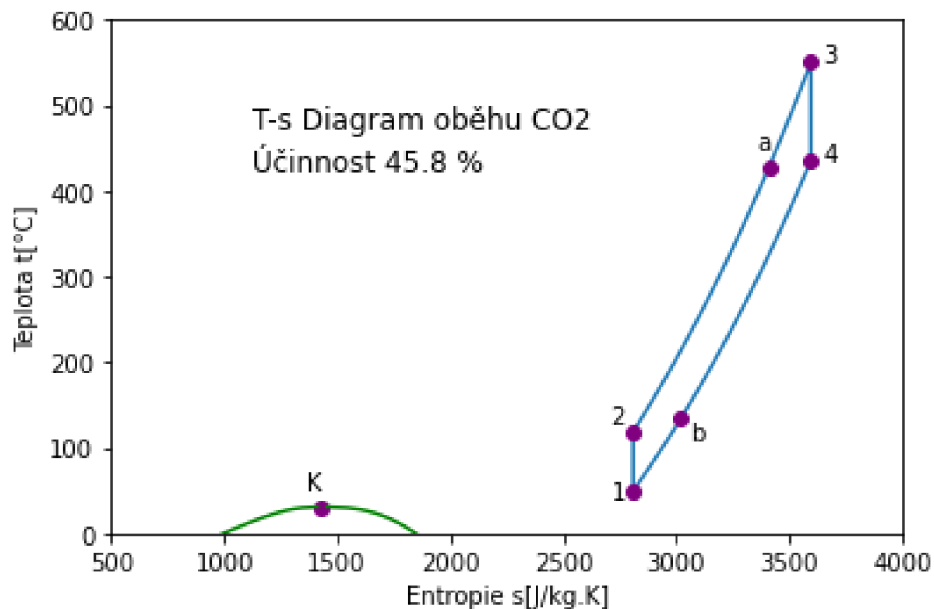


Obrázek 17 - Závislost hmotnostního průtoku na tlaku p_3

3.8 Optimalizované oběhy

3.8.1 První optimalizovaný oběh

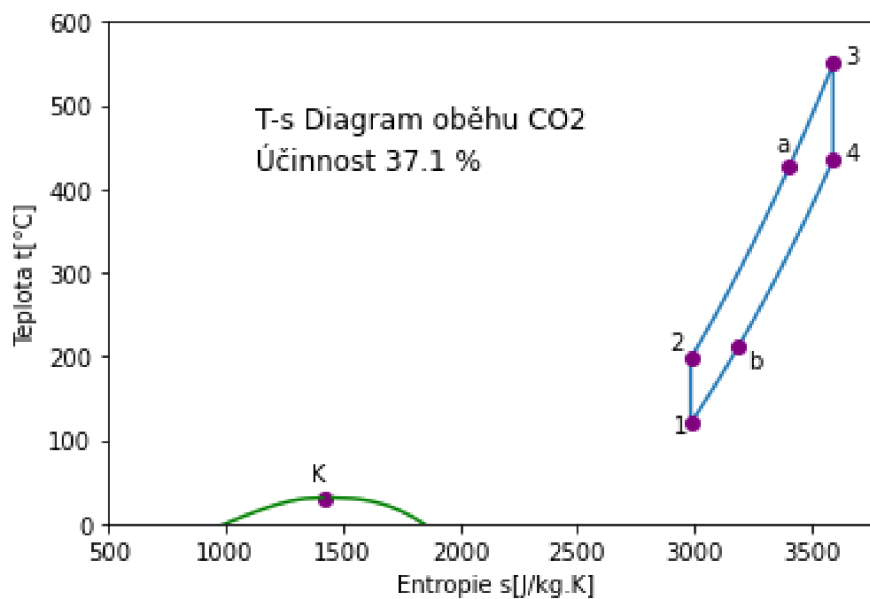
Při daných teplotách $t_1 = 50\text{ °C}$ a $t_3 = 550\text{ °C}$, vychází u oběhu CO_2 s regenerací nejlepší účinnost s ohledem na technickou proveditelnost při tlacích $p_1 = 0,10\text{ MPa}$ a $p_3 = 0,25\text{ MPa}$.



Obrázek 18 - Ts diagram prvního optimalizovaného oběhu

3.8.2 Druhý optimalizovaný oběh s využitím odpadního tepla

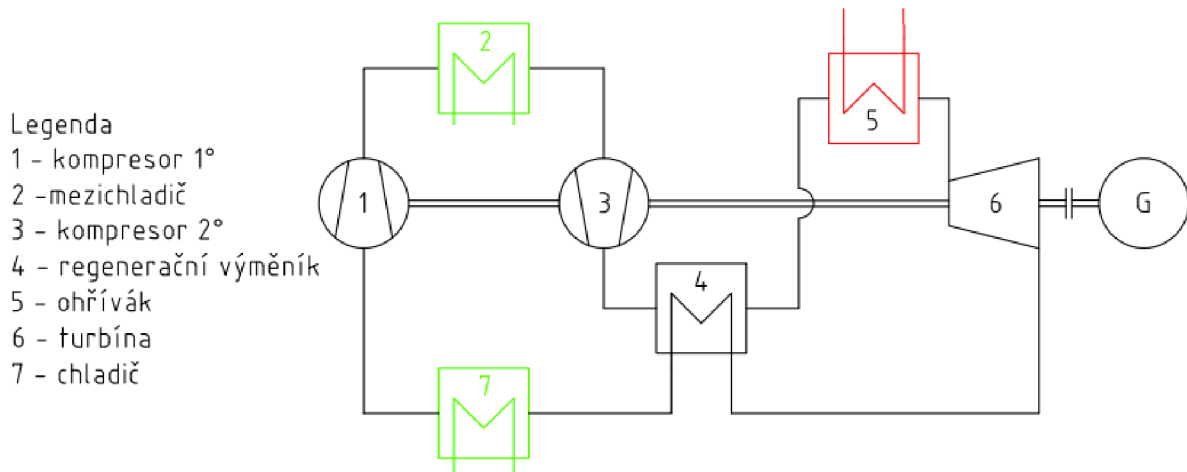
Pokud zvedneme minimální teplotu na $t_1 = 120\text{ °C}$ tak, abychom mohli využít odpadního tepla k vytápění, tak při zachování ostatních parametrů dostaneme následující oběh.



Obrázek 19 - Ts diagram druhého optimalizovaného oběhu

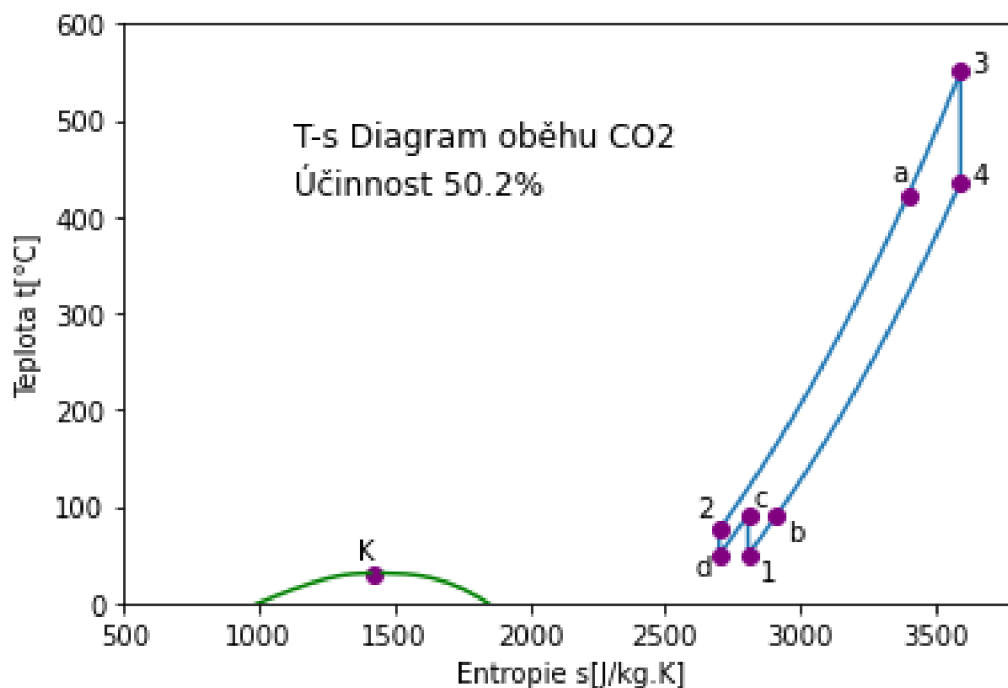
3.8.3 Třetí optimalizovaný oběh s dělenou kompresí

K oběhu přidáme mezichlazení a druhý kompresor viz. schéma oběhu s dělenou kompresí na obrázku 20.



Obrázek 20 - Schéma oběhu s dělenou kompresí

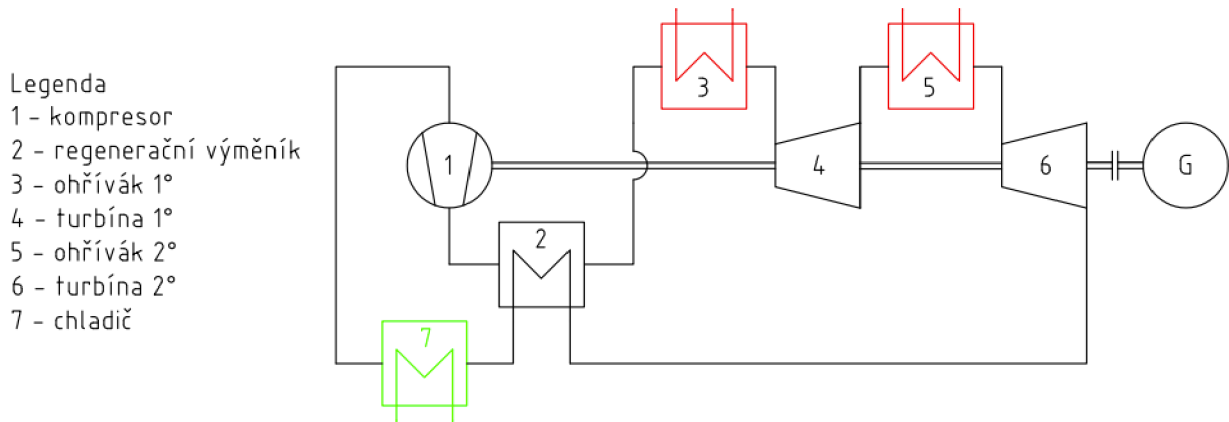
Zachováním parametrů z prvního optimalizovaného oběhu a přidáním dvoustupňové dělené komprese získáme oběh na obrázku 21. Meztlak p_c byl zvolen jako průměr tlaků p_1 a p_3 .



Obrázek 21 - Ts diagram třetího optimalizovaného oběhu

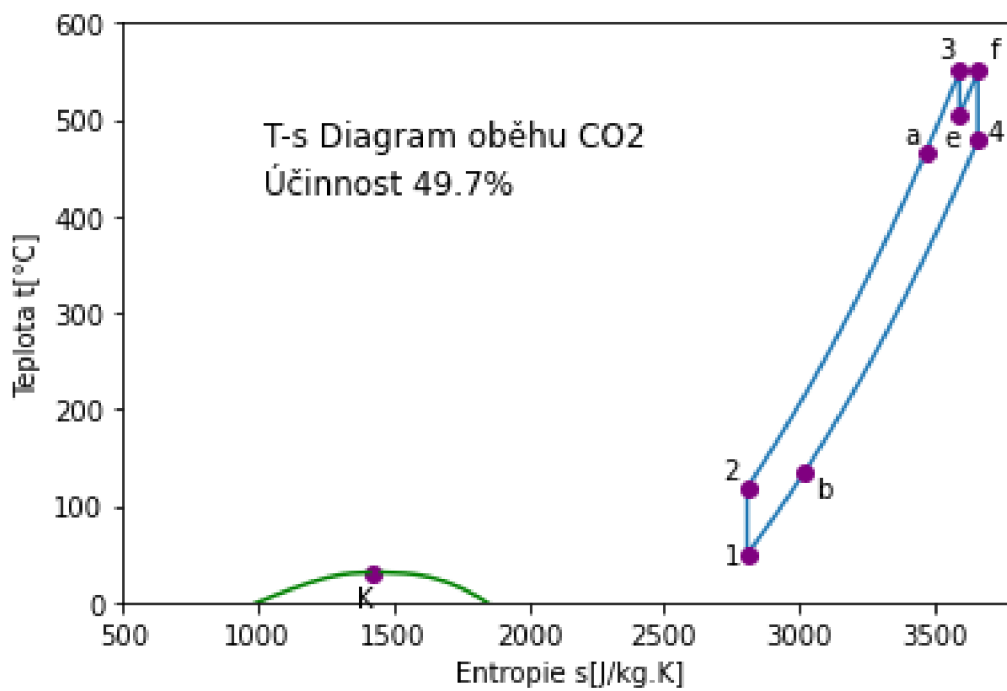
3.8.4 Čtvrtý optimalizovaný oběh s dělenou expanzí

K oběhu přidáme mezichlazení a druhou turbínu viz. schéma oběhu s dělenou expanzí na obrázku 22.



Obrázek 22 - Schéma oběhu s dělenou expanzí

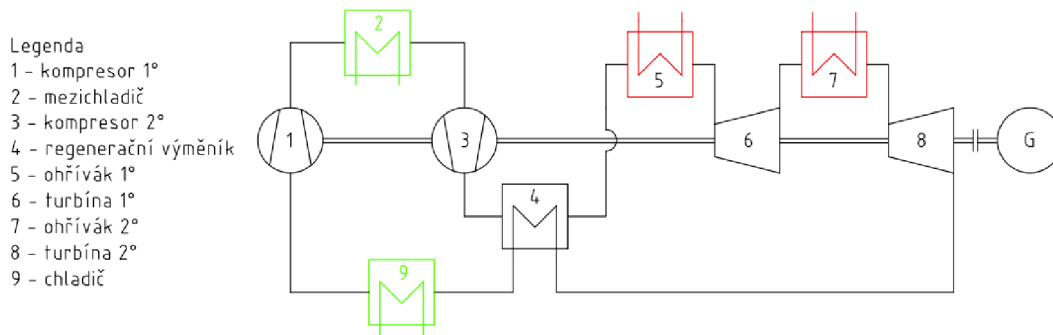
Zachováním parametrů z prvního optimalizovaného oběhu a přidáním dvoustupňové dělené expanze získáme oběh na obrázku 23. Meztlak p_e byl zvolen jako průměr tlaků p_1 a p_3 .



Obrázek 23 - Ts diagram oběhu s dělenou expanzí

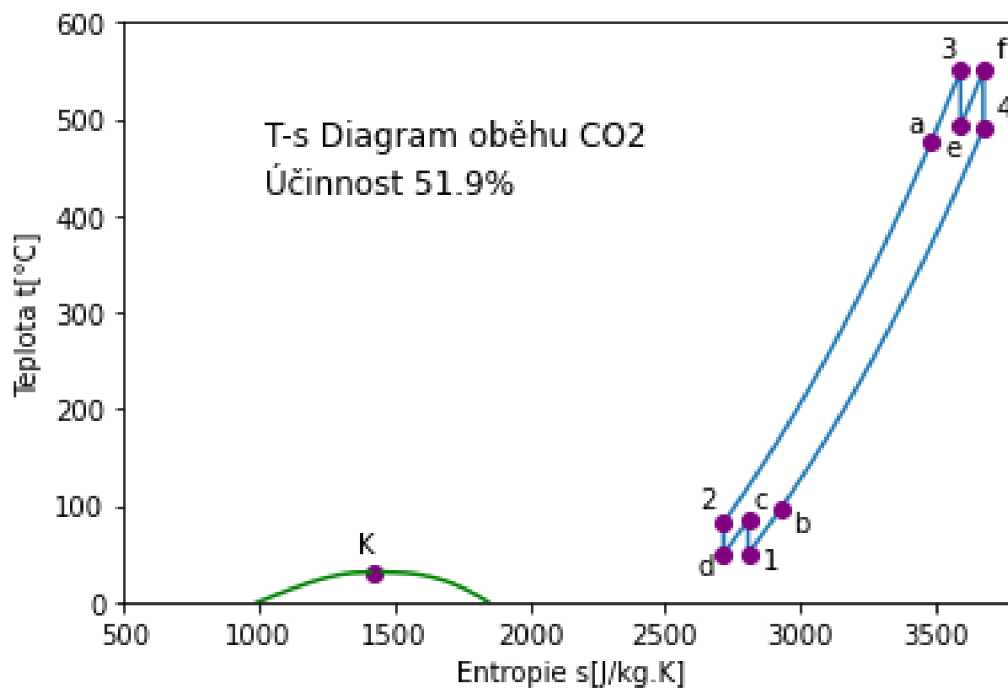
3.8.5 Pátý optimalizovaný oběh s dělenou kompresí a expanzí

Když zkombinujeme technická řešení, použitá u třetího a čtvrtého optimalizovaného oběhu, získáme oběh viz. schéma oběhu s dělenou kompresí a expanzí na obrázku 24.



Obrázek 24 - Schéma oběhu s dělenou kompresí a expanzí

Stejně jako u třetího a čtvrtého optimalizovaného oběhu jsou tlaky mezichlazení p_c a přihřívání p_e průměrem tlaků p_1 a p_3 . Ts diagram tohoto oběhu můžeme vidět na obrázku 25.



Obrázek 25 - Ts diagram oběhu s dělenou kompresí a expanzí

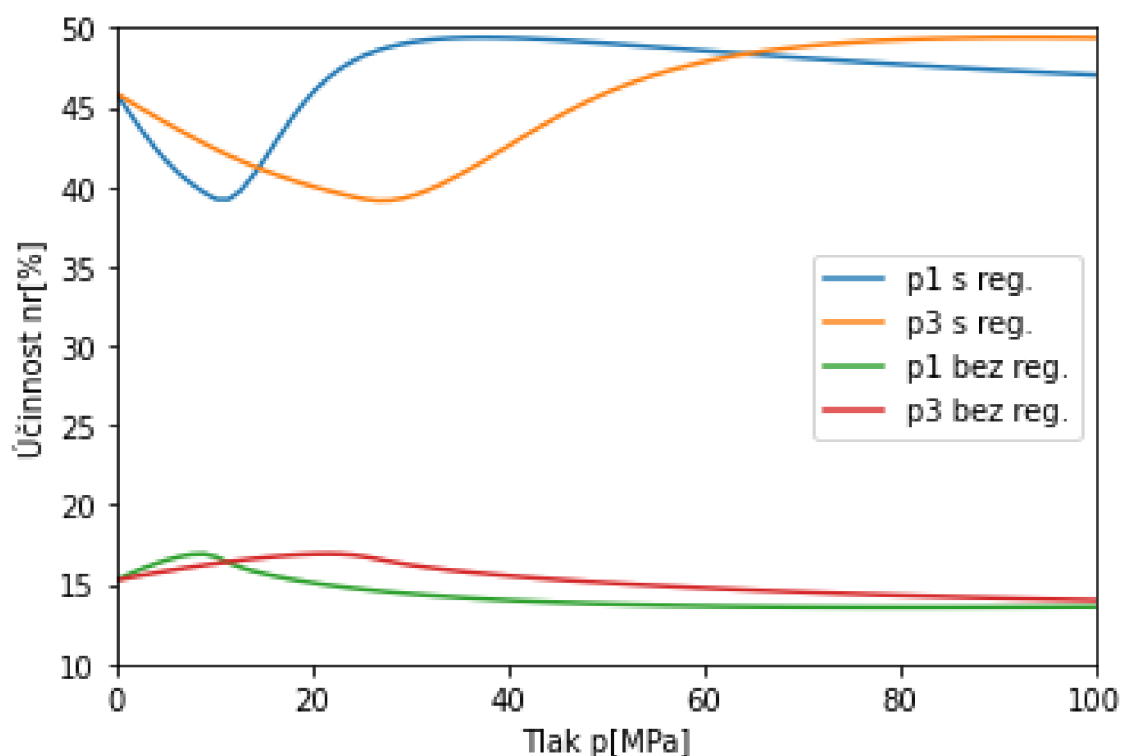
3.9 Souhrnný přehled výsledků

V tabulce 4 níže můžeme vidět souhrnný přehled základních vstupních parametrů do výpočetního modelu a výsledných vypočtených účinností pro všechny v práci počítané oběhy.

Tabulka 4 - Shrnutí parametrů všech oběhů

Oběh	t_1 [°C]	t_3 [°C]	p_1 [MPa]	p_3 [MPa]	η_c [%]	η [%]	η_r [%]	η_s [%]
Výchozí	50	550	2,5	6,5	60,74	16,59	43,62	26,17
První optimalizovaný	50	550	0,1	0,25	60,74	15,31	45,76	27,46
S využitím odpadního tepla	120	550	0,1	0,25	52,24	14,98	37,07	22,24
S dělenou kompresí	50	550	0,1	0,25	60,74	14,88	50,27	30,16
S dělenou expanzí	50	550	0,1	0,25	60,74	14,62	49,74	29,84
S dělenou exp. a komp.	50	550	0,1	0,25	60,74	13,99	51,9	31,14

Dále na obrázku 26 můžeme vidět v citlivostní analýze tlaků graficky znázorněný rozdíl mezi účinnostmi oběhů bez použití regenerace a účinností u oběhů za použití regenerace, a to při různých hodnotách tlaků.



Obrázek 26 - Citlivostní analýza tlaků pro oběh s regenerací a bez regenerace

4 Závěr

Všechny dosavadní výpočty předpokládaly ideální oběhy s dokonalou kompresí a expanzí. Jediné, v čem už je zahrnuta účinnost komponentů je regenerace, u které předpokládáme účinnost regeneračního tepelného výměníku 98 %. Pokud bychom se chtěli přiblížit k účinnostem, kterých bychom dosahovali u reálných oběhů, pak bychom měli zahrnout i účinnosti komponentů, jako jsou kompresory a turbíny, u kterých lze předpokládat účinnosti okolo 80 %. Pokud zahrneme účinnosti komponent a další ztráty, jako jsou tlakové ztráty v potrubí a vlastní spotřeba zařízení, do celkové účinnosti oběhu, dostaneme se přibližně na 60 % původní vypočtené účinnosti. A tyto odhady ztrát jsme použili i pro výpočet odhadované skutečné účinnosti η_s .

První, čeho si můžeme na výsledcích shrnutých v tabulce 4 na straně 28 všimnout, je to že se nám u optimalizovaných oběhů, oproti výchozímu oběhu, zvedá účinnost oběhu s regenerací, ale klesá účinnost bez regenerace. Je to dáno tím, že v rámci citlivostní analýzy tlaků vychází minimum pro oběh s regenerací přibližně pro stejný tlak jako maximum pro oběh bez regenerace, viz. obrázek 26 na straně 28. Další, čeho si můžeme všimnout, je obrovský vliv použití regenerace na účinnost. U všech oběhů se účinnost zvedla přibližně trojnásobně díky použití regenerace. Další opatření pro zvýšení účinnosti, jako je použití dělené komprese a dělené expanze, navýší účinnost pořád výrazně, ale už se jedná o navýšení přibližně jen o desetinu. Při použití obou, jak dělené komprese, tak i dělené expanze je nárůst účinnosti pořád znatelný, ale už se v rámci ideálního oběhu začínáme přibližovat účinnosti Carnotova oběhu pro dané teploty, a tedy už není moc kam účinnost dál navyšovat a další navýšení už tedy není ani zdaleka tak velké.

Kdybychom měli dospět k doporučení jednoho z našich pěti optimalizovaných oběhů, asi bychom volili třetí optimalizovaný oběh s dělenou kompresí, protože má velice dobrou účinnost v porovnání s ostatními oběhy a zároveň by toto řešení nebylo tak komplikované a drahé, jako použití i dělené expanze, což je tím, že všechny součásti, které přidáváme, jsou určené pro nižší teploty a jsou na ně tedy i menší nároky, což znamená že budou i levnější.

Přínos této práce může spočívat v prozkoumání možností, jak zlepšit účinnost plynových oběhů, především pro aplikace na systémy nepřímé akumulace elektrické energie, s ohledem na technickou proveditelnost, složitost, velikost a cenu onoho provedení. Dále se jedná o část návrhu většího zařízení, které má potenciál podílet se na řešení problémů s nestabilitou výkonu OZE a obecněji s výkyvy, jak na straně výroby, tak i na straně spotřeby elektrické energie.

Je třeba dodat, že tato práce ani zdaleka toto téma nevyčerpala a je tu značný prostor pro další

zkoumání v této oblasti. Další zkoumání, které by tuto záležitost značně posunulo, by mohlo spočívat například v optimalizaci reálných komponentů oběhu, jako je turbína a kompresor, speciálně pro tyto účely. Nebo obdobná optimalizace oběhu, jaká je obsažena v této práci, ale pro problematičtější RC oběh, kde by šlo dosáhnout na vyšší účinnosti, za cenu vyšší složitosti a ceny. Případně by také jistě bylo zajímavé zkusit i jiná pracovní média, jako je třeba N_2 , což by umožnilo jít i k nižším teplotám, a tedy realizovat i systém na nepřímou akumulaci elektrické energie nejen ohřívající akumulární médium na co nejvyšší teplotu, ale zároveň ochlazující jiné akumulární médium na co nejnižší teplotu, což by znamenalo větší rozdíl minimální a maximální teploty, a tedy i možnost dostat se k vyšším účinnostem oběhu. Dále by stálo za prozkoumání a optimalizaci i uspořádání a počet stupňů dělené komprese a expanze.

5 Použitá literatura

- [1] BEČVÁŘ, Josef. *Tepelné turbíny*. Praha: Nakladatelství technické literatury, 1968.
- [2] KADRNOŽKA, Jaroslav. *Tepelné elektrárny a teplárny*. Praha: Nakladatelství technické literatury, 1984
- [3] AZZUNI, Abdelrahman; BREYER, Christian. Energy security and energy storage technologies. *Energy Procedia*, 2018.
- [4] VIGNAROOBAN, K., et al. Heat transfer fluids for concentrating solar power systems—a review. *Applied Energy*, 2015.
- [5] GUNEY, Mukrimin Sevket; TEPE, Yalcin. Classification and assessment of energy storage systems. *Renewable and Sustainable Energy Reviews*, 2017.
- [6] AHN, Yoonhan, et al. Review of supercritical CO₂ power cycle technology and current status of research and development. *Nuclear Engineering and Technology*, 2015.
- [7] SCIACOVELLI, Adriano; VECCHI, Andrea; DING, Yulong. Liquid air energy storage (LAES) with packed bed cold thermal storage—From component to system level performance through dynamic modelling. *Applied energy*, 2017.
- [8] PENG, Xiaodong, et al. Liquid air energy storage flexibly coupled with LNG regasification for improving air liquefaction. *Applied Energy*, 2019.
- [9] AZAD, Murari Lal; KHURSHEED, Aizad; KUMAR, Vikas. Mitigating power oscillations in wind power plants using ESS. In: 2015 international conference on futuristic trends on computational analysis and knowledge management (ABLAZE). IEEE, 2015.
- [10] GEORGIU, Solomos; SHAH, Nilay; MARKIDES, Christos N. A thermo-economic analysis and comparison of pumped-thermal and liquid-air electricity storage systems. *Applied Energy*, 2018.
- [11] KERE, A., et al. Dynamic behavior of a sensible-heat based thermal energy storage. *Energy Procedia*, 2014.
- [12] MACHALA, Jiří. *Používané principy chlazení v průmyslu*. 2010.
- [13] KOZÁK, Tomáš. *Termodynamika parního cyklu jaderných elektráren*. 2013.
- [14] BOBČÍK, Marek. *Ideální oběhy plynových turbín*. 2011.
- [15] CIBULKA, Šimon. *Nabíjecí teplotní charakteristiky elektrochemických zdrojů*. 2017.

[16]PALMOVÁ, I.; SCHONGUT, J. Outlook of production and utilization of hydrogen. *Chemické listy*, 2004.

[17]PBS. *PBS* [online]. [cit. 2021-5-10]. Dostupné z:

<https://www.pbs.cz/cz/Produkty/Energetika/Turbiny/Expanzni-plynove-turbiny>

[18]OEnergetice.cz: Energostat [online]. Kpt. Nálepky 620/7, Nové Dvory, 674 01 Třebíč: OM Solutions, 2021 [cit. 2021-6-6]. Dostupné z:

<https://oenergetice.cz/energostat?fbclid=IwAR0q0KIGUNt3w7vSBteGgiHvmkd-3z4Oin-Iojst5Psznpi02netOaqtBDw>

[19]TZBinfo [online]. 2020 [cit. 2021-6-14]. Dostupné z: <https://oze.tzb-info.cz/akumulace-elektřiny/21610-skladovani-elektřiny-prostřednictvím-tepla-aneb-carnotovy-baterie-1-díl-principy-a-přehled>

[20]Nanosun [online]. [cit. 2021-6-20]. Dostupné z: <https://www.nanosun.cz/en/lg-ess.html#>

```

# PŘÍLOHA 01    Oběh s Regenerací
#-----
"""
@author: Pokorný V.
"""
#-----
#Načtení knihoven

import numpy as np
import scipy
import matplotlib.pyplot as plt
import CoolProp
from CoolProp.CoolProp import PropsSI

#-----
# Vstupy

t1 = 50          # Zadání teploty na vstupu do kompresoru v °C
p1 = 2.5        # Zadání tlaku na vstupu do kompresoru v MPa

t3 = 550        # Zadání teploty na vstupu do turbíny v °C
p3 = 6.5        # Zadání tlaku na vstupu do turbíny v MPa

fluid = 'CO2'   # Definice média

Z = 50          # Zadání složitost výpočtu

Tm = 15         # Posunutí teploty abychom docílili dostatečného teplotního gradientu n

nv = 0.98       # účinnost výměníku
nk = 0.82       # účinnost kompresoru
nt = 0.80       # účinnost turbíny
#-----
# Kritický bod a saturační křivka

p_crit = PropsSI('Pcrit', fluid)
t_crit = PropsSI('Tcrit', fluid)
s_crit = PropsSI('S', 'T', t_crit, 'P', p_crit, fluid)
t = np.linspace(t_crit-50, t_crit-0.01, num=Z)
s_sat0=np.zeros(len(t))
for i in range(len(t)):
    s_sat0[i] = PropsSI('S', 'T', t[i], 'Q', 0, fluid)
s_sat1=np.zeros(len(t))
for i in range(len(t)):
    s_sat1[i] = PropsSI('S', 'T', t[i], 'Q', 1, fluid)

#-----
# Definice bodu 1

t1 = t1 + 273.15
p1 = p1 * 1e6
s1 = PropsSI('S', 'P', p1, 'T', t1, fluid)
D1 = PropsSI('D', 'T', t1, 'P', p1, fluid)
h1 = PropsSI('H', 'S', s1, 'T', t1, fluid)

# Definice bodu 3
t3 = t3 + 273.15
p3 = p3 * 1e6
s3 = PropsSI('S', 'P', p3, 'T', t3, fluid)
D3 = PropsSI('D', 'T', t3, 'P', p3, fluid)
h3 = PropsSI('H', 'S', s3, 'T', t3, fluid)

```

```

# Definice bodu 2
s2 = s1
t2 = PropsSI('T', 'S', s1, 'P', p3, fluid)
h2 = PropsSI('H', 'S', s2, 'T', t2, fluid)

# Definice bodu 4
s4 = s3
t4 = PropsSI('T', 'S', s3, 'P', p1, fluid)
h4 = PropsSI('H', 'S', s4, 'T', t4, fluid)

# Definice bodu B
tb = t2 + Tm
pb = p1
sb = PropsSI('S', 'T', tb, 'P', pb, fluid)
hb = PropsSI('H', 'S', sb, 'T', tb, fluid)

# Definice bodu A
q4b = hb - h4
ha = h2 - q4b * (1 / nv)
pa = p3
sa = PropsSI('S', 'H', ha, 'P', pa, fluid)
ta = PropsSI('T', 'S', sa, 'P', pa, fluid)

# Izobara p2=p3
p23 = p3
s23 = np.linspace(s2, s3, Z)
t23 = np.zeros(Z)
for i in range(Z):
    t23[i] = PropsSI('T', 'S', s23[i], 'P', p23, fluid)

# Izobara p4=p1
p41 = p1
s41 = np.linspace(s4, s1, Z)
t41 = np.zeros(Z)
for i in range(Z):
    t41[i] = PropsSI('T', 'S', s41[i], 'P', p41, fluid)

#-----
# Výpočet účinnosti

qa3 = h3 - ha
qb1 = h1 - hb
q23 = h3 - h2
q41 = h1 - h4
q2a = ha - h2

nc = (1 - (t1 / t3))
nc = nc * 100

Wo = (q23 + q41)
n = Wo / q23
n = n * 100

Wr = qa3 + qb1
nr = (Wr / qa3)
nr = nr * 100

#-----
# Převody jednotek

```

```

t = t - 273.15
ta = ta - 273.15
tb = tb - 273.15
t1 = t1 - 273.15
t2 = t2 - 273.15
t3 = t3 - 273.15
t4 = t4 - 273.15
t41 = t41 - 273.15
t23 = t23 - 273.15
p_crit = p_crit / 1e6
print('Kritický tlak pk =', round(p_crit, 2), 'MPa')
t_crit = t_crit - 273.15
print('Kritická teplota tk =', round(t_crit), '°C')

#-----

# sloučení dat o entropii
s_obeh=np.zeros(2*Z+4)
s_obeh[0] = s1
s_obeh[1] = s2
for i in range(2,Z+2):
    s_obeh[i] = s23[i-2]
s_obeh[Z+2] = s3
s_obeh[Z+3] = s4
for i in range(Z+4,2*Z+4):
    s_obeh[i] = s41[i-(Z+4)]

# sloučení dat o teplotě
t_obeh=np.zeros(2*Z+4)
t_obeh[0] = t1
t_obeh[1] = t2
for i in range(2,Z+2):
    t_obeh[i] = t23[i-2]
t_obeh[Z+2] = t3
t_obeh[Z+3] = t4
for i in range(Z+4,2*Z+4):
    t_obeh[i] = t41[i-(Z+4)]

# Výstupy
#-----
# Parametry

print('Médiem je', fluid)
print('p1 =', round(p1 / 1e6,2), 'MPa', ' t1 =', round(t1), '°C')
print('p3 =', round(p3 / 1e6,2), 'MPa', ' t3 =', round(t3), '°C')
print('Účinnost carnotova cyklu nc = ', round(nc,2), '%')
print('Účinnost s regenerací nr =', round(nr,2), '%')
print('Účinnost n =', round(n,2), '%')

#-----
# T-s Diagram

plt.plot(s_obeh,t_obeh)
plt.plot([s_crit],[t_crit],'o', color = 'purple')
plt.text(s_crit - 60, t_crit + 20, 'K', fontsize=10)
plt.plot(s_sat0,t, 'g-')
plt.plot(s_sat1,t, 'g-')
plt.plot([s1,s2,s3,s4,sa,sb],[t1,t2,t3,t4,ta,tb],'o', color = 'purple')
plt.xlim(500, round(s3/1000)*1000 + 100)
plt.ylim(0, round(t3/100)*100)

```

```

plt.text(s1 - 100, t1 - 10, '1', fontsize=10)
plt.text(s2 - 100, t2 + 10, '2', fontsize=10)
plt.text(s3 + 50, t3, '3', fontsize=10)
plt.text(s4 + 50, t4, '4', fontsize=10)
plt.text(sa - 60, ta + 20, 'a', fontsize=10)
plt.text(sb + 40, tb - 25, 'b', fontsize=10)
plt.text(s_crit - 300, t3 - 75, 'T-s Diagram oběhu CO2', fontsize=12)
plt.text(s_crit - 300, t3 - 125, 'Účinnost ' + str(round(nr,1)) + ' %', fontsize=12)
plt.xlabel('Entropie s[J/kg.K]')
plt.ylabel('Teplota t[°C]')
plt.show()

```

```

#-----

```

```

# Množství plynu => velikost zařízení

```

```

P = 1 * 1e6

```

```

m = P / Wo

```

```

v1 = m / D1

```

```

v3 = m / D3

```

```

print('Potřebný hmotnostní průtok pro výrobu 1 MW je ', round(m,2), ' [kg/s]')

```

```

print('Objemový průtok na vstupu do kompresoru je ', round(v1,2), '[m3/s]')

```

```

print('Objemový průtok na vstupu do turbíny je ', round(v3,2), '[m3/s]')

```

```

#-----

```

```

# PŘÍLOHA 02    Oběh s dělenou kompresí
#-----
"""
@author: Pokorny V.
"""

#Načtení knihoven
import numpy as np
import scipy
import matplotlib.pyplot as plt
import CoolProp
from CoolProp.CoolProp import PropsSI
# Vstupy
#-----
t1 = 50          # Zadání teploty na vstupu do kompresoru v °C
p1 = 0.1        # Zadání tlaku na vstupu do kompresoru v MPa

t3 = 550        # Zadání teploty na vstupu do turbíny v °C
p3 = 0.25       # Zadání tlaku na vstupu do turbíny v MPa

fluid = 'CO2'   # Definice média

Z = 50          # Složitost výpočtu

Tm = 15         # Posunutí teploty pro dosažení teplotního gradientu

TP = 0          # Posunutí bodu D od bodu 1 o určitý počet °C

Kp1p2 = 0.50    # Koeficient určující kde mezi p1 a p2 bude ležet pc
#-----

print('Médium je', fluid)

#Kritický bod a saturační křivka
p_crit = PropsSI('Pcrit', fluid)
t_crit = PropsSI('Tcrit', fluid)
s_crit = PropsSI('S', 'T', t_crit, 'P', p_crit, fluid)
t = np.linspace(t_crit-50,t_crit-0.01, num=Z)
s_sat0=np.zeros(len(t))
for i in range(len(t)):
    s_sat0[i] = PropsSI('S', 'T', t[i], 'Q', 0, fluid)
s_sat1=np.zeros(len(t))
for i in range(len(t)):
    s_sat1[i] = PropsSI('S', 'T', t[i], 'Q', 1, fluid)

#Definice bodu 1
t1 = t1 + 273.15
p1 = p1 * 1e6
s1 = PropsSI('S', 'P', p1, 'T', t1, fluid)
D1 = PropsSI('D', 'T', t1, 'P', p1, fluid)
h1 = PropsSI('H', 'S', s1, 'T', t1, fluid)

#Definice bodu 3
t3 = t3 + 273.15
p3 = p3 * 1e6
s3 = PropsSI('S', 'P', p3, 'T', t3, fluid)
D3 = PropsSI('D', 'T', t3, 'P', p3, fluid)
h3 = PropsSI('H', 'S', s3, 'T', t3, fluid)

#Definice bodu 4
s4 = s3

```

```

t4 = PropsSI('T', 'S', s3, 'P', p1, fluid)
h4 = PropsSI('H', 'S', s4, 'T', t4, fluid)

#Definice bodu C
pc = p1 + (p3 - p1) * (Kp1p2) # Tlak na Kp1p2*100% cesty mezi p1 a p2
sc = s1
tc = PropsSI('T', 'P', pc, 'S', sc, fluid)
hc = PropsSI('H', 'S', sc, 'T', tc, fluid)

#Definice bodu D
pd = pc
td = t1 + TP #PropsSI('T', 'S', sd, 'P', pd, fluid)
sd = PropsSI('S', 'P', pd, 'T', td, fluid)
hd = PropsSI('H', 'S', sd, 'T', td, fluid)

#Definice bodu 2
s2 = sd
p2 = p3
t2 = PropsSI('T', 'S', s2, 'P', p2, fluid)
h2 = PropsSI('H', 'S', s2, 'T', t2, fluid)

# Definice bodu B
tb = t2 + Tm
pb = p1
sb = PropsSI('S', 'T', tb, 'P', pb, fluid)
hb = PropsSI('H', 'S', sb, 'T', tb, fluid)

# Definice bodu A
q4b = hb - h4
ha = h2 - q4b
pa = p3
sa = PropsSI('S', 'H', ha, 'P', pa, fluid)
ta = PropsSI('T', 'S', sa, 'P', pa, fluid)

#Izobara p2=p3
p23 = p3
s23 = np.linspace(s2,s3,Z)
t23 = np.zeros(Z)
for i in range(Z):
    t23[i] = PropsSI('T', 'S', s23[i], 'P', p23, fluid)

#Izobara p4=p1
p41 = p1
s41 = np.linspace(s4,s1,Z)
t41 = np.zeros(Z)
for i in range(Z):
    t41[i] = PropsSI('T', 'S', s41[i], 'P', p41, fluid)

#Izobara pc=pd
pcd = pc
scd = np.linspace(sc,sd,Z)
tcd = np.zeros(Z)
for i in range(Z):
    tcd[i] = PropsSI('T', 'S', scd[i], 'P', pcd, fluid)

#Výpočet účinnosti
qa3 = h3 - ha
qb1 = h1 - hb
q23 = h3 - h2
q41 = h1 - h4
qcd = hd - hc

```



```
q2a = ha - h2
```

```
nc = ( 1 - (t1 / t3))  
nc = nc * 100
```

```
Wo = q23 + q41 + qcd  
n = Wo / q23  
n = n * 100
```

```
Wr = qa3 + qb1 + qcd  
nr = Wr / qa3  
nr = nr * 100
```

```
#Převody jednotek
```

```
t = t - 273.15  
ta = ta - 273.15  
tb = tb - 273.15  
tc = tc - 273.15  
td = td - 273.15  
t1 = t1 - 273.15  
t2 = t2 - 273.15  
t3 = t3 - 273.15  
t4 = t4 - 273.15  
t41 = t41 - 273.15  
t23 = t23 - 273.15  
tcd = tcd - 273.15  
p_crit = p_crit / 1e6  
print('Kritický tlak pk =', round(p_crit), 'MPa')  
t_crit = t_crit - 273.15  
print('Kritická teplota tk =', round(t_crit), '°C')
```

```
#sloučení dat o entropii
```

```
s_obeh=np.zeros(3*Z+6)  
s_obeh[0] = s1  
s_obeh[1] = sc  
for i in range(2,Z+2):  
    s_obeh[i] = scd[i-2]  
s_obeh[Z+2] = sd  
s_obeh[Z+3] = s2  
for i in range(Z+4,2*Z+4):  
    s_obeh[i] = s23[i-(Z+4)]  
s_obeh[2*Z+4] = s3  
s_obeh[2*Z+5] = s4  
for i in range(2*Z+6,3*Z+6):  
    s_obeh[i] = s41[i-(2*Z+6)]
```

```
#sloučení dat o teplotě
```

```
t_obeh=np.zeros(3*Z+6)  
t_obeh[0] = t1  
t_obeh[1] = tc  
for i in range(2,Z+2):  
    t_obeh[i] = tcd[i-2]  
t_obeh[Z+2] = td  
t_obeh[Z+3] = t2  
for i in range(Z+4,2*Z+4):  
    t_obeh[i] = t23[i-(Z+4)]  
t_obeh[2*Z+4] = t3  
t_obeh[2*Z+5] = t4  
for i in range(2*Z+6,3*Z+6):  
    t_obeh[i] = t41[i-(2*Z+6)]
```

```

#Parametry
print('p1 =', round(p1 / 1e6), 'MPa', ' t1 =', round(t1), '°C')
print('p3 =', round(p3 / 1e6), 'MPa', ' t3 =', round(t3), '°C')
print('Účinnost carnotova cyklu nc = ', round(nc,2), '%')
print('Účinnost s regenerací nr =', round(nr,2), '%')
print('Účinnost n =', round(n,2), '%')

#T-s Diagram
plt.plot(s_obeh,t_obeh)
plt.plot([s_crit],[t_crit],'o', color = 'purple')
plt.text(s_crit - 60, t_crit + 15, 'K', fontsize=10, )
plt.plot(s_sat0,t, 'g-')
plt.plot(s_sat1,t, 'g-')
plt.plot([s1,s2,s3,s4,sa,sb,sc,sd],[t1,t2,t3,t4,ta,tb,tc,td],'o', color = 'purple')
plt.xlim(500, round(s3/1000)*1000 - 200)
plt.ylim(0, round(t3/100)*100)
plt.text(s1 + 40, t1 - 25, '1', fontsize=10)
plt.text(s2 - 80, t2 + 15, '2', fontsize=10)
plt.text(s3 + 40, t3, '3', fontsize=10)
plt.text(s4 + 40, t4, '4', fontsize=10)
plt.text(sa - 60, ta + 15, 'a', fontsize=10)
plt.text(sb + 40, tb - 25, 'b', fontsize=10)
plt.text(sc + 20, tc + 10, 'c', fontsize=10)
plt.text(sd - 95, td - 20, 'd', fontsize=10)
plt.text(s_crit - 300, t3 - 75, 'T-s Diagram oběhu CO2', fontsize=12)
plt.text(s_crit - 300, t3 - 125, 'Účinnost ' + str(round(nr,1)) + '%', fontsize=12)
plt.xlabel('Entropie s[J/kg.K]')
plt.ylabel('Teplota t[°C]')
plt.show()

#Množství plynu => velikost zařízení
P = 1 * 1e6
m = P / Wo
v1 = m / D1
v3 = m / D3
print('Potřebný hmotnostní průtok pro výrobu 1 MW je ', round(m,2), ' [kg/s]')
print('Objemový průtok na vstupu do kompresoru je ', round(v1,3), '[m3/s]')
print('Objemový průtok na vstupu do turbíny je ', round(v3,3), '[m3/s]')

```

```

# PŘÍLOHA 03    Oběh s dělenou expanzí
#-----
"""
@author: Pokorný V.
"""
#Načtení knihoven
import numpy as np
import scipy
import matplotlib.pyplot as plt
import CoolProp
from CoolProp.CoolProp import PropsSI
# Vstupy
#-----
t1 = 50          # Zadání teploty na vstupu do kompresoru v °C
p1 = 0.1         # Zadání tlaku na vstupu do kompresoru v MPa

t3 = 550        # Zadání teploty na vstupu do turbíny v °C
p3 = 0.25       # Zadání tlaku na vstupu do turbíny v MPa

fluid = 'CO2'   # Definice média

Z = 50          # Složitost výpočtu

Tm = 15         # Posunutí teploty pro dosažení teplotního gradientu

TP = 0          # Posunutí bodu D od bodu 1 o určitý počet °C

Kp1p2 = 0.50    # Koefficient určující kde mezi p1 a p2 bude ležet pc
#-----

print('Médium je', fluid)
#Kritický bod a saturační křivka
p_crit = PropsSI('Pcrit', fluid)
t_crit = PropsSI('Tcrit', fluid)
s_crit = PropsSI('S', 'T', t_crit, 'P', p_crit, fluid)
t = np.linspace(t_crit-50, t_crit-0.01, num=Z)
s_sat0=np.zeros(len(t))
for i in range(len(t)):
    s_sat0[i] = PropsSI('S', 'T', t[i], 'Q', 0, fluid)
s_sat1=np.zeros(len(t))
for i in range(len(t)):
    s_sat1[i] = PropsSI('S', 'T', t[i], 'Q', 1, fluid)

#Definice bodu 1
t1 = t1 + 273.15
p1 = p1 * 1e6
s1 = PropsSI('S', 'P', p1, 'T', t1, fluid)
D1 = PropsSI('D', 'T', t1, 'P', p1, fluid)

#Definice bodu 3
t3 = t3 + 273.15
p3 = p3 * 1e6
s3 = PropsSI('S', 'P', p3, 'T', t3, fluid)
D3 = PropsSI('D', 'T', t3, 'P', p3, fluid)

#Definice bodu 2
s2 = s1
t2 = PropsSI('T', 'S', s2, 'P', p3, fluid)
h2 = PropsSI('H', 'S', s2, 'T', t2, fluid)
p2 = PropsSI('P', 'S', s2, 'T', t2, fluid)

```

```

#Dělená expanze

#Definice bodu E
t44 = PropsSI('T', 'S', s3, 'P', p1, fluid)
pe = p1 + (p3 - p1) * (Kp1p2)
se = s3
te = PropsSI('T', 'S', se, 'P', pe, fluid)

#Definice bodu F
pf = pe
tf = t3
sf = PropsSI('S', 'T', tf, 'P', pf, fluid)

#Definice bodu 4
s4 = sf
t4 = PropsSI('T', 'S', s4, 'P', p1, fluid)
h4 = PropsSI('H', 'S', s4, 'T', t4, fluid)

#Regenerace

# Definice bodu B
tb = t2 + Tm
pb = p1
sb = PropsSI('S', 'T', tb, 'P', pb, fluid)
hb = PropsSI('H', 'S', sb, 'T', tb, fluid)

# Definice bodu A
q4b = hb - h4
ha = h2 - q4b
pa = p3
sa = PropsSI('S', 'H', ha, 'P', pa, fluid)
ta = PropsSI('T', 'S', sa, 'P', pa, fluid)

#Izobara p2=p3
p23 = p3
s23 = np.linspace(s2,s3,Z)
t23 = np.zeros(Z)
for i in range(Z):
    t23[i] = PropsSI('T', 'S', s23[i], 'P', p23, fluid)

#Izobara p4=p1
p41 = p1
s41 = np.linspace(s4,s1,Z)
t41 = np.zeros(Z)
for i in range(Z):
    t41[i] = PropsSI('T', 'S', s41[i], 'P', p41, fluid)

#Izobara pe=pf
pef = pe
sef = np.linspace(se,sf,Z)
tef = np.zeros(Z)
for i in range(Z):
    tef[i] = PropsSI('T', 'S', sef[i], 'P', pef, fluid)

#Výpočet účinnosti
h1 = PropsSI('H', 'S', s1, 'T', t1, fluid)
h2 = PropsSI('H', 'S', s2, 'T', t2, fluid)
h3 = PropsSI('H', 'S', s3, 'T', t3, fluid)
h4 = PropsSI('H', 'S', s4, 'T', t4, fluid)
ha = PropsSI('H', 'S', sa, 'T', ta, fluid)

```

```

hb = PropsSI('H', 'S', sb, 'T', tb, fluid)
he = PropsSI('H', 'S', se, 'T', te, fluid)
hf = PropsSI('H', 'S', sf, 'T', tf, fluid)

qa3 = h3 - ha
qb1 = h1 - hb
q23 = h3 - h2
q41 = h1 - h4
qef = hf - he

nc = ( 1 - (t1 / t3))
nc = nc * 100

Wo = q23 + q41 + qef
n = Wo / (q23 + qef)
n = n * 100

Wr = qa3 + qb1 + qef
nr = Wr / (qa3 + qef)
nr = nr * 100

#Převody jednotek
t = t - 273.15
ta = ta - 273.15
tb = tb - 273.15
te = te - 273.15
tf = tf - 273.15
t1 = t1 - 273.15
t2 = t2 - 273.15
t3 = t3 - 273.15
t4 = t4 - 273.15
t41 = t41 - 273.15
t23 = t23 - 273.15
tef = tef - 273.15
p_crit = p_crit / 1e6
print('Kritický tlak pk =', round(p_crit), 'MPa')
t_crit = t_crit - 273.15
print('Kritická teplota tk =', round(t_crit), '°C')

#sloučení dat o entropii
s_obeh=np.zeros(3*Z+6)
s_obeh[0] = s1
s_obeh[1] = s2
for i in range(2,Z+2):
    s_obeh[i] = s23[i-2]
s_obeh[Z+2] = s3
s_obeh[Z+3] = se
for i in range(Z+4,2*Z+4):
    s_obeh[i] = sef[i-(Z+4)]
s_obeh[2*Z+4] = sf
s_obeh[2*Z+5] = s4
for i in range(2*Z+6,3*Z+6):
    s_obeh[i] = s41[i-(2*Z+6)]

#sloučení dat o teplotě
t_obeh=np.zeros(3*Z+6)
t_obeh[0] = t1
t_obeh[1] = t2
for i in range(2,Z+2):
    t_obeh[i] = t23[i-2]
t_obeh[Z+2] = t3

```

```

t_obeh[Z+3] = te
for i in range(Z+4,2*Z+4):
    t_obeh[i] = tef[i-(Z+4)]
t_obeh[2*Z+4] = tf
t_obeh[2*Z+5] = t4
for i in range(2*Z+6,3*Z+6):
    t_obeh[i] = t41[i-(2*Z+6)]

#Parametry
print('p1 =',round(p1 / 1e6), 'MPa',' t1 =',round(t1), '°C')
print('p3 =',round(p3 / 1e6), 'MPa',' t3 =',round(t3), '°C')
print('Účinnost carnotova cyklu nc = ',round(nc,2), '%')
print('Účinnost s regenerací nr =',round(nr,2), '%')
print('Účinnost n =',round(n,2), '%')

#T-s Diagram
plt.plot(s_obeh,t_obeh)
plt.plot([s_crit],[t_crit],'o', color = 'purple')
plt.text(s_crit - 60, t_crit - 35, 'K', fontsize=10)
plt.plot(s_sat0,t, 'g-')
plt.plot(s_sat1,t, 'g-')
plt.plot([s1,s2,s3,s4,sa,sb,se,sf],[t1,t2,t3,t4,ta,tb,te,tf],'o', color = 'purple')
plt.xlim(500, round(s3/1000)*1000 - 200)
plt.ylim(0, round(t3/100)*100)
plt.text(s1 - 90, t1 - 10, '1', fontsize=10)
plt.text(s2 - 90, t2 + 10, '2', fontsize=10)
plt.text(s3 - 80, t3 + 15, '3', fontsize=10)
plt.text(s4 + 30, t4, '4', fontsize=10)
plt.text(sa - 80, ta + 10, 'a', fontsize=10)
plt.text(sb + 40, tb - 25, 'b', fontsize=10)
plt.text(se - 55, te - 30, 'e', fontsize=10)
plt.text(sf + 50, tf + 15, 'f', fontsize=10)
plt.text(s_crit - 400, t3 - 75, 'T-s Diagram oběhu CO2', fontsize=12)
plt.text(s_crit - 400, t3 - 125, 'Účinnost ' + str(round(nr,1)) + '%', fontsize=12)
plt.xlabel('Entropie s[J/kg.K]')
plt.ylabel('Teplota t[°C]')
plt.show()

#Množství plynu => velikost zařízení
P = 1 * 1e6
m = P / Wo
v1 = m / D1
v3 = m / D3
print('Potřebný hmotnostní průtok pro výrobu 1 MW je ',round(m,2),' [kg/s]')
print('Objemový průtok na vstupu do kompresoru je ', round(v1,3),' [m3/s]')
print('Objemový průtok na vstupu do turbíny je ',round(v3,3),' [m3/s]')

```

```

# PŘÍLOHA 04    Oběh s dělenou kompresí a expanzí
#-----
"""
@author: Pokorny V.
"""
#Načtení knihoven
import numpy as np
import scipy
import matplotlib.pyplot as plt
import CoolProp
from CoolProp.CoolProp import PropsSI
# Vstupy
#-----
t1 = 50          # Zadání teploty na vstupu do kompresoru v °C
p1 = 0.1        # Zadání tlaku na vstupu do kompresoru v MPa

t3 = 550        # Zadání teploty na vstupu do turbíny v °C
p3 = 0.25       # Zadání tlaku na vstupu do turbíny v MPa

fluid = 'CO2'   # Definice média

Z = 50          # Složitosat výpočtu

Tm = 15         # Posunutí teploty pro dosažení teplotního gradientu

TP = 0          # Posunutí bodu D od bodu 1 o určitý počet °C

Kp1p2 = 0.50    # Koefficient určující kde mezi p1 a p2 bude ležet pc
#-----

print('Médium je', fluid)

#Kritický bod a saturační křivka
p_crit = PropsSI('Pcrit', fluid)
t_crit = PropsSI('Tcrit', fluid)
s_crit = PropsSI('S', 'T', t_crit, 'P', p_crit, fluid)
t = np.linspace(t_crit-50,t_crit-0.01, num=Z)
s_sat0=np.zeros(len(t))
for i in range(len(t)):
    s_sat0[i] = PropsSI('S', 'T', t[i], 'Q', 0, fluid)
s_sat1=np.zeros(len(t))
for i in range(len(t)):
    s_sat1[i] = PropsSI('S', 'T', t[i], 'Q', 1, fluid)

#Definice bodu 1
t1 = t1 + 273.15
p1 = p1 * 1e6
s1 = PropsSI('S', 'P', p1, 'T', t1, fluid)
D1 = PropsSI('D', 'T', t1, 'P', p1, fluid)

#Definice bodu 3
t3 = t3 + 273.15
p3 = p3 * 1e6
s3 = PropsSI('S', 'P', p3, 'T', t3, fluid)
D3 = PropsSI('D', 'T', t3, 'P', p3, fluid)

#Dělená komprese

#Definice bodu C
t22 = PropsSI('T', 'S', s1, 'P', p3, fluid)
tc = (t1 + t22) / 2

```

```

sc = s1
pc = PropsSI('P', 'T', tc, 'S', sc, fluid)

#Definice bodu D
pd = pc
td = t1
sd = PropsSI('S', 'T', td, 'P', pd, fluid)

#Dělená expanze

#Definice bodu E
t44 = PropsSI('T', 'S', s3, 'P', p1, fluid)
te = (t3 + t44) / 2
se = s3
pe = PropsSI('P', 'T', te, 'S', se, fluid)

#Definice bodu F
pf = pe
tf = t3
sf = PropsSI('S', 'T', tf, 'P', pf, fluid)

#Definice bodu 2
s2 = sd
t2 = PropsSI('T', 'S', sd, 'P', p3, fluid)
h2 = PropsSI('H', 'S', s2, 'T', t2, fluid)

#Definice bodu 4
s4 = sf
t4 = PropsSI('T', 'S', s4, 'P', p1, fluid)
h4 = PropsSI('H', 'S', s4, 'T', t4, fluid)

#Regenerace

# Definice bodu B
tb = t2 + Tm
pb = p1
sb = PropsSI('S', 'T', tb, 'P', pb, fluid)
hb = PropsSI('H', 'S', sb, 'T', tb, fluid)

# Definice bodu A
q4b = hb - h4
ha = h2 - q4b
pa = p3
sa = PropsSI('S', 'H', ha, 'P', pa, fluid)
ta = PropsSI('T', 'S', sa, 'P', pa, fluid)

#Izobara p2=p3
p23 = p3
s23 = np.linspace(s2,s3,Z)
t23 = np.zeros(Z)
for i in range(Z):
    t23[i] = PropsSI('T', 'S', s23[i], 'P', p23, fluid)

#Izobara p4=p1
p41 = p1
s41 = np.linspace(s4,s1,Z)
t41 = np.zeros(Z)
for i in range(Z):
    t41[i] = PropsSI('T', 'S', s41[i], 'P', p41, fluid)

```



```

#Izobara pc=pd
pcd = pc
scd = np.linspace(sc,sd,Z)
tcd = np.zeros(Z)
for i in range(Z):
    tcd[i] = PropsSI('T', 'S', scd[i], 'P', pcd, fluid)

#Izobara pe=pf
pef = pe
sef = np.linspace(se,sf,Z)
tef = np.zeros(Z)
for i in range(Z):
    tef[i] = PropsSI('T', 'S', sef[i], 'P', pef, fluid)

#Výpočet účinnosti
h1 = PropsSI('H', 'S', s1, 'T', t1, fluid)
h2 = PropsSI('H', 'S', s2, 'T', t2, fluid)
h3 = PropsSI('H', 'S', s3, 'T', t3, fluid)
h4 = PropsSI('H', 'S', s4, 'T', t4, fluid)
ha = PropsSI('H', 'S', sa, 'T', ta, fluid)
hb = PropsSI('H', 'S', sb, 'T', tb, fluid)
hc = PropsSI('H', 'S', sc, 'T', tc, fluid)
hd = PropsSI('H', 'S', sd, 'T', td, fluid)
he = PropsSI('H', 'S', se, 'T', te, fluid)
hf = PropsSI('H', 'S', sf, 'T', tf, fluid)

qa3 = h3 - ha
qb1 = h1 - hb
q23 = h3 - h2
q41 = h1 - h4
qcd = hd - hc
qef = hf - he

nc = ( 1 - (t1 / t3))
nc = nc * 100

Wo = q23 + q41 + qcd + qef
n = Wo / (q23 + qef)
n = n * 100

Wr = qa3 + qb1 + qcd + qef
nr = Wr / (qa3 + qef)
nr = nr * 100

#Převody jednotek
t = t - 273.15
ta = ta - 273.15
tb = tb - 273.15
tc = tc - 273.15
td = td - 273.15
te = te - 273.15
tf = tf - 273.15
t1 = t1 - 273.15
t2 = t2 - 273.15
t3 = t3 - 273.15
t4 = t4 - 273.15
t41 = t41 - 273.15
t23 = t23 - 273.15
tcd = tcd - 273.15
tef = tef - 273.15
p_crit = p_crit / 1e6

```

```

print('Kritický tlak pk =',round(p_crit), 'MPa')
t_crit = t_crit -273.15
print('Kritická teplota tk =',round(t_crit), '°C')

#sloučení dat o entropii
s_obeh=np.zeros(4*Z+8)
s_obeh[0] = s1
s_obeh[1] = sc
for i in range(2,Z+2):
    s_obeh[i] = scd[i-2]
s_obeh[Z+2] = sd
s_obeh[Z+3] = s2
for i in range(Z+4,2*Z+4):
    s_obeh[i] = s23[i-(Z+4)]
s_obeh[2*Z+4] = s3
s_obeh[2*Z+5] = se
for i in range(2*Z+6,3*Z+6):
    s_obeh[i] = sef[i-(2*Z+6)]
s_obeh[3*Z+6] = sf
s_obeh[3*Z+7] = s4
for i in range(3*Z+8,4*Z+8):
    s_obeh[i] = s41[i-(3*Z+8)]

#sloučení dat o teplotě
t_obeh=np.zeros(4*Z+8)
t_obeh[0] = t1
t_obeh[1] = tc
for i in range(2,Z+2):
    t_obeh[i] = tcd[i-2]
t_obeh[Z+2] = td
t_obeh[Z+3] = t2
for i in range(Z+4,2*Z+4):
    t_obeh[i] = t23[i-(Z+4)]
t_obeh[2*Z+4] = t3
t_obeh[2*Z+5] = te
for i in range(2*Z+6,3*Z+6):
    t_obeh[i] = tef[i-(2*Z+6)]
t_obeh[3*Z+6] = tf
t_obeh[3*Z+7] = t4
for i in range(3*Z+8,4*Z+8):
    t_obeh[i] = t41[i-(3*Z+8)]

#Parametry
print('p1 =',round(p1 / 1e6), 'MPa',' t1 =',round(t1), '°C')
print('p3 =',round(p3 / 1e6), 'MPa',' t3 =',round(t3), '°C')
print('Účinnost carnotova cyklu nc = ',round(nc,2), '%')
print('Účinnost s regenerací nr =',round(nr,3), '%')
print('Účinnost n =',round(n,2), '%')

#T-s Diagram
plt.plot(s_obeh,t_obeh)
plt.plot([s_crit],[t_crit],'o', color = 'purple')
plt.text(s_crit - 60, t_crit + 20, 'K', fontsize=10)
plt.plot(s_sat0,t, 'g-')
plt.plot(s_sat1,t, 'g-')
plt.plot([s1,s2,s3,s4,sa,sb,sc,sd,se,sf],[t1,t2,t3,t4,ta,tb,tc,td,te,tf], 'o', color = 'purple')
plt.xlim(500, round(s3/1000)*1000 - 200)
plt.ylim(0, round(t3/100)*100)
plt.text(s1 + 50, t1 - 15, '1', fontsize=10)
plt.text(s2 - 90, t2 + 15, '2', fontsize=10)
plt.text(s3 - 80, t3 + 15, '3', fontsize=10)

```

```

plt.text(s4 + 50, t4 + 15, '4', fontsize=10)
plt.text(sa - 90, ta + 10, 'a', fontsize=10)
plt.text(sb + 40, tb - 25, 'b', fontsize=10)
plt.text(sc + 20, tc + 15, 'c', fontsize=10)
plt.text(sd - 100, td - 20, 'd', fontsize=10)
plt.text(se - 60, te - 30, 'e', fontsize=10)
plt.text(sf + 50, tf + 15, 'f', fontsize=10)
plt.text(s_crit - 400, t3 - 75, 'T-s Diagram oběhu CO2', fontsize=12)
plt.text(s_crit - 400, t3 - 125, 'Účinnost ' + str(round(nr,1)) + '%', fontsize=12)
plt.xlabel('Entropie s[J/kg.K]')
plt.ylabel('Teplota t[°C]')
plt.show()

```

#Množství plynu => velikost zařízení

```
P = 1 * 1e6
```

```
m = P / Wo
```

```
v1 = m / D1
```

```
v3 = m / D3
```

```
print('Potřebný hmotnostní průtok pro výrobu 1 MW je ', round(m,2), ' [kg/s]')
```

```
print('Objemový průtok na vstupu do kompresoru je ', round(v1,3), '[m3/s]')
```

```
print('Objemový průtok na vstupu do turbíny je ', round(v3,3), '[m3/s]')
```

```

# PŘÍLOHA 05      Citlivostní analýza teplot
#-----
"""
@author: Pokorny V.
"""
#-----
#Načtení knihoven
import numpy as np
import scipy
import matplotlib.pyplot as plt
import CoolProp
from CoolProp.CoolProp import PropsSI
#-----

# Funkce
#-----
def obeh(t1,p1,t3,p3):

    #-----
    # Vstupy

    fluid = 'CO2'      # Definice média

    Z = 50              # Zadání složitost výpočtu

    Tm = 15            # Posunutí teploty abychom docílili dostatečného teplotního gradientu

    nv = 0.98          # účinnost výměníku

    #-----
    # Definice bodu 1

    t1 = t1 + 273.15
    p1 = p1 * 1e6
    s1 = PropsSI('S', 'P', p1, 'T', t1, fluid)
    D1 = PropsSI('D', 'T', t1, 'P', p1, fluid)
    h1 = PropsSI('H', 'S', s1, 'T', t1, fluid)

    # Definice bodu 3
    t3 = t3 + 273.15
    p3 = p3 * 1e6
    s3 = PropsSI('S', 'P', p3, 'T', t3, fluid)
    D3 = PropsSI('D', 'T', t3, 'P', p3, fluid)
    h3 = PropsSI('H', 'S', s3, 'T', t3, fluid)

    # Definice bodu 2
    s2 = s1
    t2 = PropsSI('T', 'S', s1, 'P', p3, fluid)
    h2 = PropsSI('H', 'S', s2, 'T', t2, fluid)

    # Definice bodu 4
    s4 = s3
    t4 = PropsSI('T', 'S', s3, 'P', p1, fluid)
    h4 = PropsSI('H', 'S', s4, 'T', t4, fluid)

    # Definice bodu B
    tb = t2 + Tm
    pb = p1
    sb = PropsSI('S', 'T', tb, 'P', pb, fluid)
    hb = PropsSI('H', 'S', sb, 'T', tb, fluid)

```

```

# Definice bodu A
q4b = hb - h4
ha = h2 - q4b * (1 / nv)
pa = p3
sa = PropsSI('S', 'H', ha, 'P', pa, fluid)
ta = PropsSI('T', 'S', sa, 'P', pa, fluid)

# Izobara p2=p3
p23 = p3
s23 = np.linspace(s2,s3,Z)
t23 = np.zeros(Z)
for i in range(Z):
    t23[i] = PropsSI('T', 'S', s23[i], 'P', p23, fluid)

# Izobara p4=p1
p41 = p1
s41 = np.linspace(s4,s1,Z)
t41 = np.zeros(Z)
for i in range(Z):
    t41[i] = PropsSI('T', 'S', s41[i], 'P', p41, fluid)

#-----
# Výpočet účinnosti

qa3 = h3 - ha
qb1 = h1 - hb
q23 = h3 - h2
q41 = h1 - h4
q2a = ha - h2

nc = ( 1 - (t1 / t3))
nc = nc * 100

Wo = q23 + q41
n = Wo / q23
n = n * 100

Wr = qa3 + qb1
nr = Wr / qa3
nr = nr * 100

#-----
# Převody jednotek

ta = ta - 273.15
tb = tb - 273.15
t1 = t1 - 273.15
t2 = t2 - 273.15
t3 = t3 - 273.15
t4 = t4 - 273.15
t41 = t41 - 273.15
t23 = t23 - 273.15

#-----
# výstupy

return(nr)

```

```
# Vstupy
```

```

t1 = 50          # Zadání teploty na vstupu do kompresoru v °C
p1 = 2.5        # Zadání tlaku na vstupu do kompresoru v MPa

t3 = 550        # Zadání teploty na vstupu do turbíny v °C
p3 = 6.5        # Zadání tlaku na vstupu do turbíny v MPa

print('účinnost s regenerací je ', round(obeh(t1,p1,t3,p3),2), '%')

T1 = np.linspace(50, 300, 50)
nrt1 = np.zeros(50)
for i in range(50):
    nrt1[i] = obeh(T1[i],p1,t3,p3)

T3 = np.linspace(300, 700, 50)
nrt3 = np.zeros(50)
for i in range(50):
    nrt3[i] = obeh(t1,p1,T3[i],p3)

plt.plot(T1,nrt1)
plt.plot(T3,nrt3)
plt.xlabel('Teplota t[°C]')
plt.ylabel('Účinnost nr[%]')
#plt.text(80, 45, 't1', fontsize=10)
#plt.text(340, 26, 't3', fontsize=10)
plt.legend(['t1', 't3'], loc = 'lower right')

```

```

# PŘÍLOHA 06      Citlivostní analýza tlaků
#-----
"""
@author: Pokorny V.
"""
#-----
#Načtení knihoven
import numpy as np
import scipy
import matplotlib.pyplot as plt
import CoolProp
from CoolProp.CoolProp import PropsSI
#-----

# Funkce
#-----
def obeh(t1,p1,t3,p3):

    #-----
    # Vstupy

    fluid = 'CO2'      # Definice média

    Z = 50             # Zadání složitost výpočtu

    Tm = 15           # Posunutí teploty abychom docílili dostatečného teplotního gradien

    nv = 0.98         # účinnost výměníku

    #-----
    # Definice bodu 1

    t1 = t1 + 273.15
    p1 = p1 * 1e6
    s1 = PropsSI('S', 'P', p1, 'T', t1, fluid)
    D1 = PropsSI('D', 'T', t1, 'P', p1, fluid)
    h1 = PropsSI('H', 'S', s1, 'T', t1, fluid)

    # Definice bodu 3
    t3 = t3 + 273.15
    p3 = p3 * 1e6
    s3 = PropsSI('S', 'P', p3, 'T', t3, fluid)
    D3 = PropsSI('D', 'T', t3, 'P', p3, fluid)
    h3 = PropsSI('H', 'S', s3, 'T', t3, fluid)

    # Definice bodu 2
    s2 = s1
    t2 = PropsSI('T', 'S', s1, 'P', p3, fluid)
    h2 = PropsSI('H', 'S', s2, 'T', t2, fluid)

    # Definice bodu 4
    s4 = s3
    t4 = PropsSI('T', 'S', s3, 'P', p1, fluid)
    h4 = PropsSI('H', 'S', s4, 'T', t4, fluid)

    # Definice bodu B
    tb = t2 + Tm
    pb = p1
    sb = PropsSI('S', 'T', tb, 'P', pb, fluid)
    hb = PropsSI('H', 'S', sb, 'T', tb, fluid)

```

```

# Definice bodu A
q4b = hb - h4
ha = h2 - q4b * (1 / nv)
pa = p3
sa = PropsSI('S', 'H', ha, 'P', pa, fluid)
ta = PropsSI('T', 'S', sa, 'P', pa, fluid)

# Izobara p2=p3
p23 = p3
s23 = np.linspace(s2,s3,Z)
t23 = np.zeros(Z)
for i in range(Z):
    t23[i] = PropsSI('T', 'S', s23[i], 'P', p23, fluid)

# Izobara p4=p1
p41 = p1
s41 = np.linspace(s4,s1,Z)
t41 = np.zeros(Z)
for i in range(Z):
    t41[i] = PropsSI('T', 'S', s41[i], 'P', p41, fluid)

#-----
# Výpočet účinnosti

qa3 = h3 - ha
qb1 = h1 - hb
q23 = h3 - h2
q41 = h1 - h4
q2a = ha - h2

nc = ( 1 - (t1 / t3))
nc = nc * 100

Wo = q23 + q41
n = Wo / q23
n = n * 100

Wr = qa3 + qb1
nr = Wr / qa3
nr = nr * 100

#-----
# Převody jednotek

ta = ta - 273.15
tb = tb - 273.15
t1 = t1 - 273.15
t2 = t2 - 273.15
t3 = t3 - 273.15
t4 = t4 - 273.15
t41 = t41 - 273.15
t23 = t23 - 273.15

#-----
# výstupy

return(nr)

#-----
# Vstupy

```



```

t1 = 50          # Zadání teploty na vstupu do kompresoru v °C
p1 = 2.5        # Zadání tlaku na vstupu do kompresoru v MPa

t3 = 550        # Zadání teploty na vstupu do turbíny v °C
p3 = 6.5        # Zadání tlaku na vstupu do turbíny v MPa

Z = 50          # Zadání složitost výpočtu
#-----

print('účinnost s regenerací je ', round(obeh(t1,p1,t3,p3)), '%')

P1 = np.linspace( p1 / 2 , p3 - (p1 / 4), Z)
nrp1 = np.zeros(Z)
for i in range(Z):
    nrp1[i] = obeh(t1,P1[i],t3,p3)

P3 = np.linspace(p1 + (p1 / 4), p3 * 2, Z)
nrp3 = np.zeros(Z)
for i in range(Z):
    nrp3[i] = obeh(t1,p1,t3,P3[i])

plt.plot(P1,nrp1)
plt.plot(P3,nrp3)
plt.xlabel('Tlak p[MPa]')
plt.ylabel('Účinnost nr[%]')
#plt.text(0.95, 39.5, 'p1', fontsize=10)
#plt.text(2.7, 27, 'p3', fontsize=10)
plt.legend(['p1','p3'], loc = 'lower right')

```

```
# PŘÍLOHA 07    Citlivostní analýza plaků s vazbou 2,5x
```

```
#-----
```

```
"""
```

```
@author: Pokorny V.
```

```
"""
```

```
#-----
```

```
#Načtení knihoven
```

```
import numpy as np
```

```
import scipy
```

```
import matplotlib.pyplot as plt
```

```
import CoolProp
```

```
from CoolProp.CoolProp import PropsSI
```

```
#-----
```

```
# Funkce
```

```
#-----
```

```
def obeh(t1,p1,t3,p3):
```

```
#-----
```

```
# Vstupy
```

```
fluid = 'CO2'    # Definice média
```

```
Z = 50          # Zadání složitost výpočtu
```

```
Tm = 15         # Posunutí teploty abychom docílili dostatečného teplotního gradien
```

```
nv = 0.98       # účinnost výměníku
```

```
#-----
```

```
# Definice bodu 1
```

```
t1 = t1 + 273.15
```

```
p1 = p1 * 1e6
```

```
s1 = PropsSI('S', 'P', p1, 'T', t1, fluid)
```

```
D1 = PropsSI('D', 'T', t1, 'P', p1, fluid)
```

```
h1 = PropsSI('H', 'S', s1, 'T', t1, fluid)
```

```
# Definice bodu 3
```

```
t3 = t3 + 273.15
```

```
p3 = p3 * 1e6
```

```
s3 = PropsSI('S', 'P', p3, 'T', t3, fluid)
```

```
D3 = PropsSI('D', 'T', t3, 'P', p3, fluid)
```

```
h3 = PropsSI('H', 'S', s3, 'T', t3, fluid)
```

```
# Definice bodu 2
```

```
s2 = s1
```

```
t2 = PropsSI('T', 'S', s1, 'P', p3, fluid)
```

```
h2 = PropsSI('H', 'S', s2, 'T', t2, fluid)
```

```
# Definice bodu 4
```

```
s4 = s3
```

```
t4 = PropsSI('T', 'S', s3, 'P', p1, fluid)
```

```
h4 = PropsSI('H', 'S', s4, 'T', t4, fluid)
```

```
# Definice bodu B
```

```
tb = t2 + Tm
```

```
pb = p1
```

```
sb = PropsSI('S', 'T', tb, 'P', pb, fluid)
```

```
hb = PropsSI('H', 'S', sb, 'T', tb, fluid)
```

```

# Definice bodu A
q4b = hb - h4
ha = h2 - q4b * (1 / nv)
pa = p3
sa = PropsSI('S', 'H', ha, 'P', pa, fluid)
ta = PropsSI('T', 'S', sa, 'P', pa, fluid)

# Izobara p2=p3
p23 = p3
s23 = np.linspace(s2,s3,Z)
t23 = np.zeros(Z)
for i in range(Z):
    t23[i] = PropsSI('T', 'S', s23[i], 'P', p23, fluid)

# Izobara p4=p1
p41 = p1
s41 = np.linspace(s4,s1,Z)
t41 = np.zeros(Z)
for i in range(Z):
    t41[i] = PropsSI('T', 'S', s41[i], 'P', p41, fluid)

#-----
# Výpočet účinnosti

qa3 = h3 - ha
qb1 = h1 - hb
q23 = h3 - h2
q41 = h1 - h4
q2a = ha - h2

nc = ( 1 - (t1 / t3))
nc = nc * 100

Wo = q23 + q41
n = Wo / q23
n = n * 100

Wr = qa3 + qb1
nr = Wr / qa3
nr = nr * 100

#-----
# Převody jednotek

ta = ta - 273.15
tb = tb - 273.15
t1 = t1 - 273.15
t2 = t2 - 273.15
t3 = t3 - 273.15
t4 = t4 - 273.15
t41 = t41 - 273.15
t23 = t23 - 273.15

#-----
# výstupy

return(n, nr)

#-----
# Vstupy

```

```

t1 = 50          # Zadání teploty na vstupu do kompresoru v °C
t3 = 550        # Zadání teploty na vstupu do turbíny v °C

p1 = 2.5         # Zadání tlaku na vstupu do kompresoru v MPa
Kp3p1 = 2.5     # Poměr p3/p1
p3 = p1 * Kp3p1 # Zadání tlaku na vstupu do turbíny v MPa

Z = 80          # Zadání složitost výpočtu
#-----

print('účinnost s regenerací je ', round(obeh(t1,p1,t3,p3)[1],2), '%')

P1 = np.linspace( 0.1 , 100 , Z)
nrp1 = np.zeros(Z)
np1 = np.zeros(Z)
for i in range(Z):
    np1[i], nrp1[i] = obeh(t1, P1[i], t3, P1[i] * Kp3p1)

P3 = np.linspace( 0.2 , 100 , Z)
nrp3 = np.zeros(Z)
np3 = np.zeros(Z)
for i in range(Z):
    np3[i], nrp3[i] = obeh(t1, P3[i] * (1 / Kp3p1), t3, P3[i])

plt.plot(P1,nrp1)
plt.plot(P3,nrp3)
plt.plot(P1,np1)
plt.plot(P3,np3)
plt.xlim(0, P3[-1])
plt.ylim(10, 50)
plt.xlabel('Tlak p[MPa]')
plt.ylabel('Účinnost nr[%]')
plt.legend(['p1 s reg.', 'p3 s reg.', 'p1 bez reg.', 'p3 bez reg.'], loc = 'best')
#plt.text(80, 49, 'p1', fontsize=10)
#plt.text(80, 51.5, 'p3', fontsize=10)

```

```
# PŘÍLOHA 08      Hmotnostní průtok
```

```
#-----
```

```
"""
```

```
@author: Pokorný V.
```

```
"""
```

```
#-----
```

```
#Načtení knihoven
```

```
import numpy as np
```

```
import scipy
```

```
import matplotlib.pyplot as plt
```

```
import CoolProp
```

```
from CoolProp.CoolProp import PropsSI
```

```
#-----
```

```
# Funkce
```

```
#-----
```

```
def obeh(t1,p1,t3,p3):
```

```
#-----
```

```
# Vstupy
```

```
fluid = 'CO2'      # Definice média
```

```
Z = 50            # Zadání složitost výpočtu
```

```
Tm = 15          # Posunutí teploty abychom docílili dostatečného teplotního gradientu
```

```
nv = 0.98        # účinnost výměníku
```

```
#-----
```

```
# Definice bodu 1
```

```
t1 = t1 + 273.15
```

```
p1 = p1 * 1e6
```

```
s1 = PropsSI('S', 'P', p1, 'T', t1, fluid)
```

```
D1 = PropsSI('D', 'T', t1, 'P', p1, fluid)
```

```
h1 = PropsSI('H', 'S', s1, 'T', t1, fluid)
```

```
# Definice bodu 3
```

```
t3 = t3 + 273.15
```

```
p3 = p3 * 1e6
```

```
s3 = PropsSI('S', 'P', p3, 'T', t3, fluid)
```

```
D3 = PropsSI('D', 'T', t3, 'P', p3, fluid)
```

```
h3 = PropsSI('H', 'S', s3, 'T', t3, fluid)
```

```
# Definice bodu 2
```

```
s2 = s1
```

```
t2 = PropsSI('T', 'S', s1, 'P', p3, fluid)
```

```
h2 = PropsSI('H', 'S', s2, 'T', t2, fluid)
```

```
# Definice bodu 4
```

```
s4 = s3
```

```
t4 = PropsSI('T', 'S', s3, 'P', p1, fluid)
```

```
h4 = PropsSI('H', 'S', s4, 'T', t4, fluid)
```

```
# Definice bodu B
```

```
tb = t2 + Tm
```

```
pb = p1
```

```
sb = PropsSI('S', 'T', tb, 'P', pb, fluid)
```

```
hb = PropsSI('H', 'S', sb, 'T', tb, fluid)
```

```

# Definice bodu A
q4b = hb - h4
ha = h2 - q4b * (1 / nv)
pa = p3
sa = PropsSI('S', 'H', ha, 'P', pa, fluid)
ta = PropsSI('T', 'S', sa, 'P', pa, fluid)

# Izobara p2=p3
p23 = p3
s23 = np.linspace(s2,s3,Z)
t23 = np.zeros(Z)
for i in range(Z):
    t23[i] = PropsSI('T', 'S', s23[i], 'P', p23, fluid)

# Izobara p4=p1
p41 = p1
s41 = np.linspace(s4,s1,Z)
t41 = np.zeros(Z)
for i in range(Z):
    t41[i] = PropsSI('T', 'S', s41[i], 'P', p41, fluid)

#-----
# Výpočet účinnosti

qa3 = h3 - ha
qb1 = h1 - hb
q23 = h3 - h2
q41 = h1 - h4
q2a = ha - h2

nc = (1 - (t1 / t3))
nc = nc * 100

Wo = q23 + q41
n = Wo / q23
n = n * 100

Wr = qa3 + qb1
nr = Wr / qa3
nr = nr * 100

#-----
# Převody jednotek

ta = ta - 273.15
tb = tb - 273.15
t1 = t1 - 273.15
t2 = t2 - 273.15
t3 = t3 - 273.15
t4 = t4 - 273.15
t41 = t41 - 273.15
t23 = t23 - 273.15

#-----
# výstupy

P = 1 * 1e6
m = P / Wo

```

```

v1 = m / D1
v3 = m / D3

return(m)
#-----
# Vstupy
t1 = 50          # Zadání teploty na vstupu do kompresoru v °C
t3 = 550        # Zadání teploty na vstupu do turbíny v °C

p1 = 2          # Zadání tlaku na vstupu do kompresoru v MPa
Kp3p1 = 2.5     # Poměr p3/p1
p3 = p1 * Kp3p1 # Zadání tlaku na vstupu do turbíny v MPa

Z = 80         # Zadání složitosti výpočtu
#-----

P3 = np.linspace( 0.2 , 4 , Z)
mp3 = np.zeros(Z)
for i in range(Z):
    mp3[i] = obeh(t1, P3[i] * (1 / Kp3p1), t3, P3[i])

plt.plot(P3,mp3)
plt.xlabel('Tlak p3[MPa]')
plt.ylabel('Hmotnostní průtok potřebný pro 1MW [kg/s]')
#plt.text(0.37, 17, 'p1', fontsize=10)
#plt.text(0.75, 42, 'p3', fontsize=10)

```