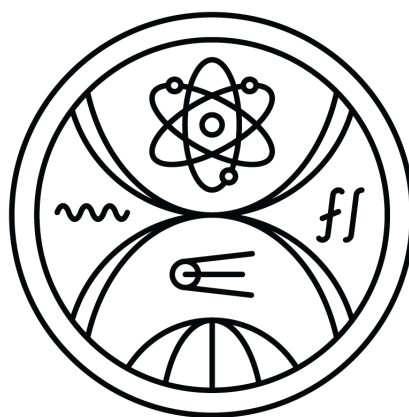


UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY

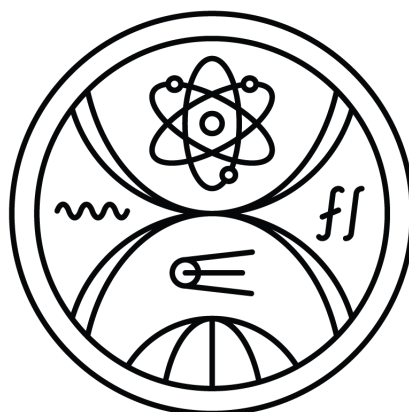


PODPORA INTELIGENTNÉHO RIADENIA
ENERGETICKÝCH SIETÍ
DIPLOMOVÁ PRÁCA

2023

BC. OMAR AL-SHAFE'I

UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY



PODPORA INTELIGENTNÉHO RIADENIA
ENERGETICKÝCH SIETÍ
DIPLOMOVÁ PRÁCA

Študijný program: Informatika
Študijný odbor: Aplikovaná informatika
Školiace pracovisko: Katedra aplikovanej informatiky
Školiteľ: prof. RNDr. Mária Lucká, PhD.

Bratislava, 2023
Bc. Omar Al-Shafe'i



Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta: Bc. Omar Al-Shafe'i
Študijný program: aplikovaná informatika (Jednoodborové štúdium, magisterský II. st., denná forma)
Študijný odbor: informatika
Typ záverečnej práce: diplomová
Jazyk záverečnej práce: slovenský
Sekundárny jazyk: anglický

Názov: Podpora inteligentného riadenia energetických sietí
Support for intelligent management of smart energy networks

Anotácia: Stabilita inteligentnej energetickej siete a zabezpečenie dodávok elektrickej energie môže byť potenciálne narušená integráciou obnoviteľných zdrojov energie (napr. fotovoltických panelov), veľkých úložísk energie (batérií), ale aj významným rozvojom elektromobility. Tieto nové prvky sa v súčasnosti v čoraz väčšej miere stávajú súčasťou moderných inteligentných energetických sietí a preto ich integrácia do celej sústavy sa stala mimoriadne dôležitou. Pôvodne jednosmerná sieť, kde sa energia od veľkých výrobcov elektriny dodávala spotrebiteľom, sa postupne mení na obojsmernú sieť, v ktorej sa mnohí odberatelia - vďaka obnoviteľným zdrojom - stávajú súčasne aj drobnými výrobcami – prosumeri. Kvôli garancii stability celej sústavy a zabezpečeniu minimálnej ceny bolo potrebné vytvoriť tzv. agregátorov flexibility, ktorým účastníci siete môžu poskytnúť svoju flexibilitu. Pod flexibilitou pritom rozumieme práva (1) na odber vyrobenej alebo uskladnenej elektrickej energie (napr. z fotovoltiky, batérie alebo elektromobilu) v určitom čase a (2) obmedzenie spotreby (napr. vypnutie kúrenia). Navrhnete a overte model práce agregátora flexibility, ktorý vďaka inteligentným algoritmom a dátovej analýze dokáže optimalizovať tok energie, zabezpečiť stabilné dodávky energie a minimalizovať náklady spotrebiteľov. Svoje riešenie obmedzte pre vybraný typ prosumerov/spotrebiteľov. Použite pritom vhodné optimalizačné metódy, predovšetkým metódy strojového učenia. Svoje riešenie implementujte a porovnajte s existujúcimi riešeniami na dostupných dátach.

Literatúra:

1. Li, T., Sun, B., Chen, Y., Ye, Z., Low, S. H., & Wierman, A. (2020). Real-time Aggregate Flexibility via Reinforcement Learning. 1–17. <http://arxiv.org/abs/2012.11261>
2. Fernando Lezama, Joao Soares, Bruno Canizes, Zita Vale: Flexibility management model of home appliances to support DSO requests in smart grids, Sustainable Cities and Society, Volume 55, 2020, ISSN 2210-6707, <https://doi.org/10.1016/j.scs.2020.102048>
3. Tongxin Li and Bo Sun and Yue Chen and Zixin Ye and Steven H. Low and Adam Wierman: Learning-Based Predictive Control via Real-Time Aggregate Flexibility, IEEE Transactions on Smart Grid, 12 (6), 2021, 97–4913, <https://doi.org/10.1109%2Ftsg.2021.3094719>.



Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

4. Steve Wattam: Artificial intelligence and machine learning approaches to energy demand-side response: A systematic review, Renewable and Sustainable Energy Reviews, Volume 130, 2020, 109899, ISSN 1364-0321, <https://doi.org/10.1016/j.rser.2020.109899>.

Vedúci: prof. RNDr. Mária Lucká, PhD.
Katedra: FMFI.KAI - Katedra aplikovanej informatiky
Vedúci katedry: doc. RNDr. Tatiana Jajcayová, PhD.
Dátum zadania: 13.12.2022

Dátum schválenia: 13.12.2022

prof. RNDr. Roman Ďurikovič, PhD.
garant študijného programu

.....
študent

.....
vedúci práce

Pod'akovanie: Chcel by som pod'akovať mojej školiteľke za cenné rady počas tvorby diplomovej práce.

Abstrakt

Slovenský abstrakt v rozsahu 100-500 slov, jeden odstavec. Abstrakt stručne sumarizuje výsledky práce. Mal by byť pochopiteľný pre bežného informatika. Nemal by teda využívať skratky, termíny alebo označenie zavedené v práci, okrem tých, ktoré sú všeobecne známe.

Kľúčové slová: jedno, druhé, tretie (prípadne štvrté, piate)

Abstract

Abstract in the English language (translation of the abstract in the Slovak language).

Keywords:

Obsah

Úvod	1
1 Opis problému	3
1.1 Formulácia problému.	3
1.2 Súčasný stav riešenej problematiky.	4
1.3 Súčasný stav riešenej problematiky v reálnom živote.	4
1.4 Súčasný stav riešenej problematiky v literatúre.	4
2 Cieľ práce	6
2.1 Úlohy agregátora flexibility	6
3 Návrh riešenia	8
3.1 Model Predictive Control.	8
3.2 Metóda zlatého rezu.	9
3.3 Metóda dichotómie.	9
3.4 Metóda rovnomerného delenia intervalu.	9
3.5 Neriadené nabíjanie.	9
3.6 Round Robin.	9
3.7 Triediace plánovacie algoritmy	10
3.7.1 Least Laxity First.	11
3.7.2 Smoothed Least Laxity First.	11
3.7.3 Enhanced Least Laxity First.	11
3.7.4 Earliest Deadline First.	11
3.7.5 Group Earliest Deadline First.	12
3.7.6 First-Come First-Served.	12
3.7.7 Last-Come First-Served.	12
3.7.8 Longest Remaining Processing Time.	12
3.7.9 Shortest Job Next.	12
3.8 Cena nabíjania.	13
3.9 Least Laxity First	13

4 Implementácia	14
4.1 Optimalizačné metódy	14
5 Experimenty	16
5.1 Ukážka vstupných dát ACN-Data	16
5.2 Využité technológie	16
5.2.1 Balík acnportal.	17
5.2.2 Architektúra simulátora ACN-Sim	18
5.2.3 Balík adacharge.	18
5.2.4 Balík acnportal-experiments.	19
5.3 Konfigurácia a obmedzenia nabíjacích sietí	19
5.3.1 Typy nabíjania.	19
5.3.2 Infraštruktúra nabíjacej stanice.	20
5.3.3 Obmedzenia pri nabíjaní.	21
5.4 Prvý experiment	21
5.5 Druhý experiment	21
5.6 Tretí experiment	21
Záver	22
Príloha A	26
Príloha B	27

Zoznam obrázkov

2.1	Funkcie agregátora flexibility	7
3.1	Príklad plánovacie algoritmu Round Robin v praxi.	10
3.2	System toku energie a ziskov.	13
4.1	Kód optimalizačnej metódy zlatého rezu	15
5.1	Ukážka vstupných dát ACN-Data	16
5.2	Schéma acnportal.	17
5.3	Architektúra simulátora ACN-Sim.	18
5.4	Typy nabíjania elektromobilov	20

Zoznam tabuliek

Úvod

S neustále pribúdajúcim množstvom elektromobilov na trhu sa zväčšuje množstvo elektrickej energie, ktorú treba dodať elektromobilom. Môže nastať situácia, že kapacity jednotlivých nabíjacích staníc nebudú stačiť pre nabitie každého elektromobilu požadovaným množstvom energie. Preto sa postupne aj odberatelia elektrickej energie stávajú výrobcami elektrickej energie pomocou obnoviteľných zdrojov. Tým sa jednosmerná sieť, v rámci ktorej veľkí výrobcovia dodávajú energiu spotrebiteľom, mení na obojsmernú sieť, kde odberatelia, ktorí sú výrobcami elektrickej energie môžu poskytovať za týmto účelom podporné služby.

Riešením týchto problémov sú inteligentné takzvané smart siete. Tieto siete obsahujú agregátor flexibility, ktorému poskytujú flexibilitu používatelia nabíjacej siete. To znamená, že agregátor flexibility má právo riadiť odber elektrickej energie pre elektromobily a prispôbovať ho podľa potrieb a podľa podmienok nabíjacej siete. Používatelia, ktorí sú tiež výrobcami elektrickej energie z obnoviteľných zdrojov, môžu poskytovať podporné služby prostredníctvom agregátora flexibility (takýchto používateľov nazývame prosumeri), a to tým, že napríklad agregátor flexibility od nich kúpi energiu. Používatelia v takýchto sieťach môžu nastaviť svoje preferencie, ako napríklad preferovaný čas nabíjania a množstvo požadovanej energie. Agregátor flexibility vie následne pomocou inteligentných algoritmov optimalizovať tok energie. Využíva pritom optimalizačné metódy alebo metódy strojového učenia (napr. neurónové siete). Agregátor flexibility tiež minimalizuje cenu elektrickej energie pre jej spotrebiteľov tým, že nakupuje elektrickú energiu v čase, keď je najlacnejšia a predáva elektrickú energiu v čase, keď je najdrahšia.

Cieľom tejto práce je overiť model práce agregátora flexibility, ktorý vie pomocou inteligentných algoritmov a dátovej analýzy optimalizovať tok energie, zabezpečiť stabilné dodávky energie a zároveň minimalizovať náklady používateľov elektromobilov. Porovnáme naše riešenia tohoto problému s inými doterajšími riešeniami. [15]

V prvej kapitole rozoberieme východiská pri tvorbe nášho modelu agregátora flexibility, opíšeme existujúce riešenia a aj obmedzenia infraštruktúry pri nabíjaní elektromobilov. V druhej kapitole predstavujeme model práce nášho agregátora flexibility a spomíname inteligentné algoritmy, ktoré sme v našom modeli použili.

V tretej kapitole overujeme model práce agregátora flexibility, kde porovnáваме naše riešenie s existujúcimi riešeniami, a to vzhľadom na pomer dodanej energie, množstvo dodanej

energie, cenu dodanej energie a počet výmen elektromobilov. Vstupné dáta o elektromobiloch získavame z nabíjacích staníc pre elektromobily: stanica ACN a stanica JPL. Výstupom našej implementácie je optimálny plán nabíjania elektromobilov vzhľadom k dodanej energii. V našej implementácii sa snažíme počet výmen elektromobilov minimalizovať, keďže to často vedie k nespokojnosti používateľov elektromobilov.

1 Opis problému

V tejto kapitole formulujeme problém, ktorý riešime a hovoríme aj o súčasnom stav jeho riešenia. Spomíname viacero prístupov od iných autorov, ktoré tento problém riešia.

1.1 Formulácia problému.

Problém, ktorý riešime formulujeme takto: Hľadáme plán nabíjania elektromobilov v nabíjacej sieti, ktorý zabezpečí maximálne dodávky požadovanej energie. Plán nabíjania musí dodržiavať obmedzenia nabíjacej siete. Obmedzenia, ktoré používame pri riešení problému sú:

1. Jednofázové. Jednofázové obmedzenia infraštruktúry sú množinou lineárných obmedzení.
2. Nevyvážené trojfázové. Nevyvážené trojfázové obmedzenia pozostávajú z množiny obmedzení kužeľa druhého rádu. Tento typ obmedzení má nabíjacia sieť Caltech ACN.

Ďalšie parametre nabíjacej siete, ktoré pri riešení problému nastavujeme sú nabíjačky (ich počet a typ), transformátor (jeho kapacita). Musíme si zvolit' aj tarify (ceny nabíjania v čase). Viac informácii o obmedzeniach a parametroch sa nachádza v článkoch [15, 11].

Problém, ktorému sa venujeme je optimalizačným problémom, lebo hľadáme taký plán nabíjania, ktorý maximalizuje účelovú funkciu.

Formálnejšie definujeme tento optimalizačný problém takto: ^{above} *main*

$$\max_r^{SCH} U_k(r) \quad (1.1)$$

$$\text{subject to:} \quad (1.2)$$

$$0 \leq r_i(t) \leq \bar{r}_i(t) \quad t < d_i, i \in V \quad (1.3)$$

$$r_i(t) = 0 \quad t \geq d_i, i \in V \quad (1.4)$$

$$\sum_{t=a_i}^{d_i-1} r_i(t) \delta \leq e_i \quad i \in V \quad (1.5)$$

$$f_j(r_1(t), \dots, r_N(t)) \leq R_j(t) \quad t \in T, j \in \hat{R} \quad (1.6)$$

$\hat{R} ??$

1.2 Súčasný stav riešenej problematiky.

Množstvo elektromobilov (okrem dvojkoľových a trojkoľových) sa má podľa predikcií [6] zvýšiť od roku 2022 do roku 2030 o 230 miliónov. Zvýši sa počet nielen súkromných elektrických áut, ale napríklad aj počet elektrických autobusov v mestskej hromadnej doprave. [6] Veľa nabíjacích staníc dnes používa na nabíjanie elektromobilov plánovací algoritmus neriadeného nabíjania. Tento algoritmus sa už nebude môcť pri pribúdajúcom množstve elektromobilov použiť z dôvodu, že neberie do úvahy kapacitu nabíjacej stanice na energiu. [15]

V poslednom desaťročí sa vedci snažili hľadať rôzne ďalšie algoritmy, ktoré by vedeli vyriešiť problém nabíjania elektromobilov pri zachovaní obmedzení infraštruktúry. [6] Potreby používateľov elektromobilov ako spotrebiteľov sú veľmi dôležité pri formulovaní optimalizačného problému, ktorý má tieto algoritmy riešiť.

My sa venujeme optimalizačnému problému dodania najväčšieho množstva požadovanej energie elektromobilom pri zachovaní obmedzení infraštruktúry siete. Riešením tohoto problému vieme zabezpečiť to, že nabíjané elektromobily získajú čo najväčšie množstvo požadovanej energie.

1.3 Súčasný stav riešenej problematiky v reálnom živote.

Problematiku riešenia optimalizačného problému dodania najväčšieho množstva požadovanej energie elektromobilom pri zachovaní obmedzení infraštruktúry riešia v článkoch [16, 17, 2]. Na riešenie tohoto optimalizačného problému sa v súčasnosti využíva neriadené nabíjanie, spomína sa v [15]. Jednou z komplikácií pri takomto riešení optimalizačného problému je, že môže dôjsť k preťaženiu infraštruktúry nabíjacej siete. Nižšie spomíname, ako tento optimalizačný problém riešia v literatúre, a ako ho riešime my. [15]

1.4 Súčasný stav riešenej problematiky v literatúre.

V dostupnej literatúre je veľa zaujímavých algoritmov, ktoré sa však nedajú využiť priamo v praxi. Hlavným dôvodom prečo sa nedajú aplikovať je podľa [16] to, že vychádzajú z predpokladov, ktoré v praxi nefungujú alebo im chýba schopnosť splniť praktické obmedzenia a ciele.

V článku [17] tento problém rieši hlavne systémový operátor. Systémový operátor vyrieši optimalizačný problém, a tým získa množstvo energie, ktorú pošle agregátorovi. Agregátor pomocou plánovacieho algoritmu získa plán nabíjania elektromobilov a dodá energiu elektromobilom.

My pri riešení tohoto optimalizačného problému využívame online plánovacie algoritmy s využitím balíka acnportal. Vstupom do online algoritmov sú elektromobily a ich para-

metre, ako napríklad čas ich príchodu, čas ich odchodu a ich požadovaná energia. Pomocou online plánovacích algoritmov získame poradie, v ktorom elektromobily nabíjame. Aby sme zistili množstvo energie, ktorú treba každému elektromobilu dodať, musíme vyriešiť spomínaný optimalizačný problém. Naším zámerom je riešiť optimalizačný problém osobitne pre každý elektromobil. Tento optimalizačný problém riešime metódou lineárneho vyhľadávania, metódy bisekcie, metódy dichotómie a pomocou kontrolných algoritmov (MPC). Riešením tohoto optimalizačného problému je optimálny plán nabíjania elektromobilov vzhľadom k dodanej energii.

2 Cieľ práce

Cieľ práce je navrhnuť a overiť model agregátora flexibility pre zvolený typ prosumerov. To znamená, že musíme popísať ako agregátor flexibility kupuje energiu, a ako ju následne elektromobilom dodáva. Energiu agregátor flexibility bude dodávať elektromobilom na základe plánovacieho algoritmu, ktorý rieši optimalizačný problém. Vyriešime optimalizačný problém dodania najväčšieho množstva energie elektromobilom pri zachovaní obmedzení infraštruktúry. Využívame pri riešení optimalizačného problému online plánovacie algoritmy (kontrolné alebo triediace), pričom vychádzame zo základných tried v balíku acn portal, ktoré definujú obmedzenia nabíjacej siete a zjednodušujú prácu s novo implementovanými algoritmami. Vstupom algoritmu, ktorý rieši tento optimalizačný problém sú údaje o príchodoch elektromobilov do nabíjacej siete Caltech. Výstupom tohoto algoritmu je celkový pomer dodanej požadovanej energie elektromobilom a celková cena dodanej energie elektromobilom atď. Overíme si navrhnutý model agregátora flexibility na verejne dostupných vstupných dátach z Caltech nabíjacej siete.

2.1 Úlohy agregátora flexibility

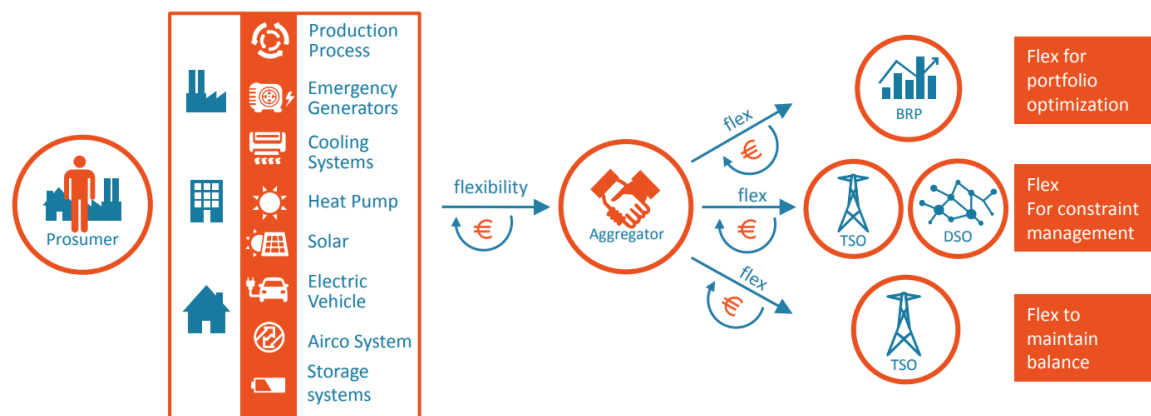
V súčasnosti pozorujeme pokrok v oblasti distribuovania obnoviteľnej energie. S čoraz vyšším dopytom po energii sa vedci musia zaoberať výskumnou otázkou: Ako môžu spotrebitelia dodávať energiu a zároveň ju prijímať? To znamená, že treba zmeniť jednosmernú sieť, kde veľké firmy dodávajú energiu spotrebiteľom, na obojsmernú sieť, kde spotrebitelia dodávajú a zároveň prijímajú energiu (skrátene takých spotrebiteľov nazývame prosumeri). Vedci ako odpoveď na túto výskumnú otázku zaviedli novú kategóriu hráčov na trhu s energetikou. Túto novú kategóriu hráčov nazývame agregátori.

Agregátor je prostredníkom medzi prosumeri a trhom s elektrinou, tým že združuje flexibilitu od prosumerov, a následne flexibilitu predáva (ako jedna entita) na trhoch s elektrinou. Prosumeri sú odmeňovaní za poskytovanie flexibility agregátorovi.

Rola agregátora zatiaľ nie je presne definovaná. Rola agregátora môže byť určená treťou stranou alebo môže byť určená už účinkujúcim hráčom na trhu s elektrinou, ktorý má napríklad záujem použiť agregátor na doplnujúce veci.

Nezávislý agregátor môže poskytovať jednu službu alebo viac služieb. Medzi služby agregátora patria agregovanie flexibility, dodávky energie spotrebiteľom a vyváženie dopytu

a ponuky na trhoch s elektrinou. [5]



Obr. 2.1: Distribúcia flexibility medzi prosumeri, agregátorom a trhom s elektrinou. Zdroj obrázka je [25].

UNFINISHED

3 Návrh riešenia

V tejto kapitole popisujeme navrhnuté riešenie optimalizačného problému dodania maximálneho množstva požadovanej energie elektromobilom pri zachovaní obmedzení infraštruktúry nabíjacej siete.

3.1 Model Predictive Control.

Algoritmus Model Predictive Control vie riešiť optimalizačný problém dodania najväčšieho množstva požadovanej energie elektromobilom pri zachovaní infraštruktúry. Pseudokód algoritmu MPC je nasledujúci:

```
for  $q \in Q$  do
  Get active EVs:
   $V_q := \{i \in V \mid e_i(q) > 0 \text{ and } d_i(q) > 0\}$ 
  if event fired or time since last computation  $> P$ :
    then
      Get optimal schedule for optimisation horizon  $T$ 
      by solving the optimisation problem:
       $(r_1^*(i), \dots, r_T^*(i), i \in V_q) := \text{solver}(V_q, U_q, R_q)$ 

      Update schedule:
       $r_{q+\tau}(i) := r_i^*(1 + \tau), \tau = 0, \dots, T - 1$ 
    end
    set pilot signal from recent schedule  $r_q(i), i \in V_q$ 
    Update energy remaining and remaining duration:
     $e_{q+1}(i) = e_q(i) - r_{q+1}(i)$ 
     $d_{q+1}(i) = d_q(i) - 1$ 
  end
```

Nižšie uvádzame optimalizačné metódy, ktoré využívame v experimentoch na riešenie optimalizačného problému s využitím online algoritmov.

3.2 Metóda zlatého rezu.

Metóda zlatého rezu sa používa pri riešení problému nájdenia maxima alebo minima unimodálnej funkcie. Unimodálna funkcia je taká, ktorá obsahuje len jedno maximum alebo minimum na intervale $[a, b]$. (zdroj prvý) [] Táto metóda je analogická k optimalizačnej metóde bisekcie. Analogickosť metódy zlatého rezu a metódy bisekcie spočíva v tom, že počiatočný interval $[a, b]$ je nahradený viacerými intervalmi $[a_i, b_i], i = 2, 3, \dots$ [21]

3.3 Metóda dichotómie.

Metóda dichotómie rieši jednorozmerné optimalizačné problémy a vieme ju aplikovať len na unimodálne funkcie (unimodálna funkcia, ktorú definujeme na intervale $[a, b]$ má na tomto intervale presne jeden bod, ktorý je maximom alebo minimom funkcie).

Postup, ktorý metóda dichotómie využíva spočíva z viacerých krokov:

1. Uvažujme unimodálnu funkciu f , ktorá má na intervale $[a, b]$ minimum alebo maximum.
2. Vygenerujeme dva nové body $c = \frac{a+b}{2} - \varepsilon$ a $d = \frac{a+b}{2} + \varepsilon$. Platí, že $\varepsilon > 0$ a $2 \cdot \varepsilon < (a - b)$.
3. Skontrolujeme, či $f(c) > f(d)$ alebo $f(c) < f(d)$. Ak platí $f(c) > f(d)$, tak dostaneme nový interval $[a, b] = [a, d]$. Inak $[a, b] = [c, b]$.

Kroky 2 a 3 metóda dichotómie opakuje dokým nenájde riešenie. Metóda dichotómie nájde riešenie vtedy, keď nový interval bude dostatočne malý (menší než epsilon).

3.4 Metóda rovnomerného delenia intervalu.

3.5 Neriadené nabíjanie.

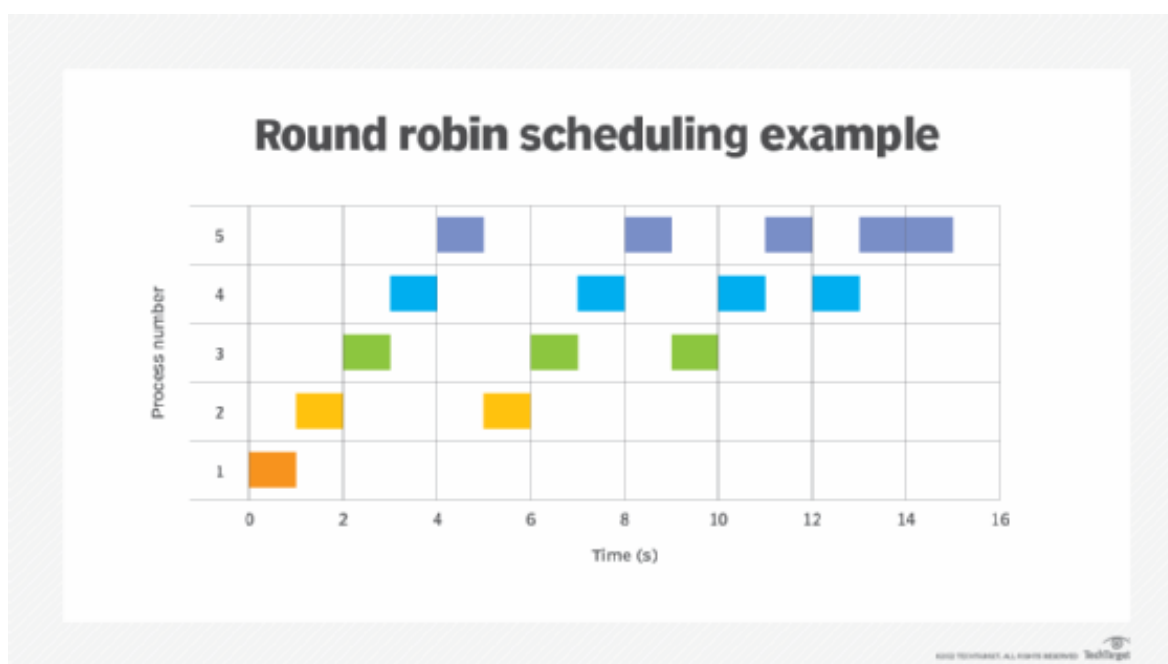
Tento plánovací algoritmus pridelí uje každému elektromobilu maximálne množstvo energie a pritom neberie do úvahy obmedzenia infraštruktúry nabíjacej stanice. Toto je najjednoduchší typ nabíjania, ktorý je stále používaný vo väčšine nabíjacích systémov na svete. [15]

3.6 Round Robin.

Plánovací algoritmus Round Robin, ktorý nazývame skráteno RR je algoritmus, ktorý je založený na myšlienke zdieľania nabíjacej kapacity medzi všetkými aktívnymi elektromobilmi. Pre každý aktívny elektromobil kontroluje algoritmus dve podmienky:

1. Je možné zvýšiť nabíjanie o 1kWh tak, aby sme zachovali obmedzenia nabíjacej siete? Ak áno, tak nabijeme elektromobil so zvýšenou rýchlosťou nabíjania a následne elektromobil pridáme na koniec radu.
2. Nie je možné zvýšiť nabíjanie o 1kWh tak, aby sme zachovali obmedzenia nabíjacej siete? Ak áno, tak nabijeme elektromobil s fixným nabíjaním. Elektromobil necháme vo radu na rovnakom mieste.

Elektromobily, ktoré dostali požadovanú energiu sú z radu odstránené. Algoritmus RR skončí vtedy, keď v rade nebude žiaden elektromobil. [15]



Obr. 3.1: Pridel'ovanie času procesom na základe plánovacieho algoritmu Round Robin. Každý spotrebiteľ má prístup k rovnakým zdrojom pomocou algoritmu Round Robin. Zdroj obrázka: [24].

3.7 Triediace plánovacie algoritmy

Triediace plánovacie algoritmy, ktoré spomíname v tejto sekcii pridel'ujú energiu elektromobilom v takom poradí, v akom tie elektromobily utriedili. Triediace plánovacie algoritmy triedia elektromobily na základe jedného parametra alebo viacerých parametrov.

Následne na riešenie problému maximálneho pridelenia energie elektromobilu pri zachovaní obmedzení nabíjacej siete využívajú triediace plánovacie algoritmy metódu bisekcie alebo iných metód (napr. metóda lineárneho vyhľadávania). Pre každý elektromobil sa vyrieši tento problém jednotlivo a algoritmus postupne pridelí energiu elektromobilom.

Tieto algoritmy môžu byť implementované ako online, a aj ako offline algoritmy. V prípade offline verzie triediacich plánovacích algoritmov vieme lepšie minimalizovať cenu

ako v online plánovacích algoritmoch. [2]

3.7.1 Least Laxity First.

Algoritmus Least Laxity First (skrátene nazývame LLF), je triediaci plánovací algoritmus, ktorý prideluje elektromobilom energiu v poradí od elektromobilu s najnižšou laxitou po elektromobil s najvyššou laxitou. Nevýhodou triediaceho plánovacieho algoritmu LLF je, že spôsobuje veľké výkyvy v rýchlosti nabíjania pri nabíjaní elektromobilov. Napríklad, keď príde veľa elektromobilov, ktoré majú ešte nižšiu laxitu ako pôvodný elektromobil s najnižšou laxitou, tak sa môže stať, že pôvodnému elektromobilu sa nepridelí žiadna energia alebo málo energie. Dlhodobo, taký spôsob nabíjania s výkyvmi v rýchlosti nabíjania môže skrátiť životnosť batérii elektromobilov.[2]

Laxitu počítame na základe vzorca:

$$\text{Laxita} = \text{zostávajúci čas do odchodu} - \frac{\text{zostávajúca požadovaná energia}}{\text{rýchlosť nabíjania}}, \quad (3.1)$$

kde predpokladáme, že rýchlosť nabíjania je vždy konštantná.

3.7.2 Smoothed Least Laxity First.

Smoothed Least Laxity First (skrátene sLLF) je triediaci plánovací algoritmus založený na algoritme LLF. Tento algoritmus lepšie rieši optimalizačný problém pridelovania maximálnej energie elektromobilom pri zachovaní obmedzení infraštruktúry nabíjacej siete lepšie ako algoritmus LLF. Algoritmus sLLF zároveň neobsahuje veľké výkyvy v rýchlosti nabíjania pri pridelovaní energie elektromobilom na rozdiel od algoritmu LLF.

Popis algoritmu sLLF sa nachádza v článku [2].

3.7.3 Enhanced Least Laxity First.

Ako reakciu na problém oscilácie pri nabíjaní elektromobilov v algoritme LLF vzniklo mnoho variánt vylepšeného algoritmu LLF. Jedným z nich je Enhanced Least Laxity First (ELLF). Algoritmus ELLF zaradí elektromobily s rovnakou najnižšou laxitou do jednej skupiny. Potom algoritmus ELLF pridelí energiu elektromobilom v skupine použitím scheduling algoritmu EDF. Na pridelenie energie zvyšku elektromobilov algoritmus ELLF aplikuje LLF. [19]

3.7.4 Earliest Deadline First.

Algoritmus Early deadline first skrátene nazývame EDF, je triediaci plánovací algoritmus. Algoritmus EDF prideluje energiu elektromobilom v poradí od elektromobilov s najskorším časom odchodu po elektromobily s najneskorším časom odchodu. [15]

3.7.5 Group Earliest Deadline First.

Algoritmus Group Earliest Deadline First je plánovací algoritmus, ktorý vytvorili zo zámerom zlepšenia výkonnosť algoritmu EDF počas preťaženia reálnej multimediálnej aplikácie. Tento scheduling algoritmus sa zakladá na myšlienke skupinového schedulingu. To znamená, že elektromobily, ktoré majú podobné časy príchodu patria do skupiny. Potom sa priradí uje energia týmto elektromobilom na základe scheduling algoritmu Shortest Job First. [22]

3.7.6 First-Come First-Served.

Triediaci plánovací algoritmus First-Come First-Served (skrátene FCFS) má veľmi podobnú funkcionálnosť ako algoritmus neriadeného nabíjania. Hlavný rozdiel spočíva v tom, že algoritmus FCFS je časovo oneskorený, lebo obsahuje elektromobily v rade, takže každý elektromobil si musí počkať pre svoje nabíjanie. [15]

3.7.7 Last-Come First-Served.

Triediaci plánovací algoritmus Last-Come First-Served (skrátene LCFS) priradí uje energiu elektromobilom, prioritne od vozidla s najneskorším časom pripojenia na nabíjačku po vozidlo s najskorším časom pripojenia na nabíjačku. [15]

3.7.8 Longest Remaining Processing Time.

Triediaci plánovací algoritmus Longest Remaining Processing Time je algoritmus, ktorý prideliť uje energiu elektromobilom takto:

1. Utriedi vstupné pole aktívnych elektromobilov, a to od elektromobilu s najdlhším zostávajúcim časom nabíjania po elektromobil s najkratším zostávajúcim časom nabíjania.
2. Pre každý elektromobil v tom poradí algoritmus vyrieši optimalizačný problém pridelenia maximálnej energie pri zachovaní obmedzení infraštruktúry.

3.7.9 Shortest Job Next.

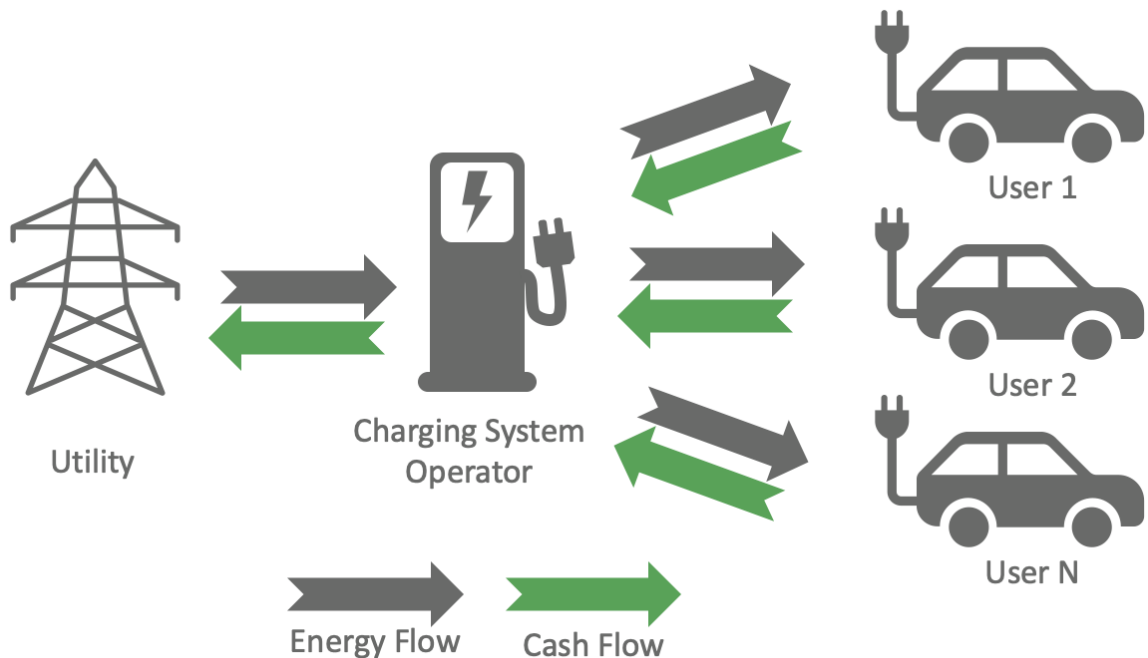
Algoritmus Shortest Job Next, skrátene SJN je triediaci plánovací algoritmus. SJN funguje ako Longest Remaining Processing Time až na jeden rozdiel, ktorý spočíva v tom, že pole aktívnych elektromobilov triedi od elektromobilu s najkratším zostávajúcim časom nabíjania, po vozidlo s najdlhším zostávajúcim časom nabíjania.

3.8 Cena nabíjania.

Náš spôsob definovania ceny nabíjania je: na konci každého mesiaca nastavíme ceny nabíjania ako α_i^* , pre každé nabíjanie i v tom mesiaci. Hovoríme, že e_i je dodaná energia pre spotrebiteľa j počas nabíjania i , kde S_j je množina všetkých nabíjaní používateľa j . Potom náklady za nabíjanie spotrebiteľa j sa na konci mesiaca vypočítajú takto:

$$\sum_{i \in S_j} \alpha_i^* e_i. \quad (3.2)$$

Narozdiel od konštantných cien a cien závislých na čase (napríklad v [17]) my používame ceny (tarify), ktoré zachytávajú skutočné ceny energie. Skutočné ceny energie závisia od jej dopytu a od preťaženie infraštruktúry. [10]



Obr. 3.2: Systémy toku energie a aj zisky za energiu. Zdroj obrázka je: [27].

3.9 Least Laxity First

4 Implementácia

Použili sme Jupyter notebook a programovací jazyk Python na implementáciu, a aj na experimenty. V balíku acn-portal je v programovacom jazyku Python implementovaná základná trieda pre infraštruktúru nabíjacej siete a aj základná trieda pre plánovacie algoritmy.

Ak vytvárame novú infraštruktúru nabíjacej siete alebo plánovací algoritmus, tak vždy nami vytvorené triedy dedia od základných tried v balíku acn-portal.

4.1 Optimalizačné metódy

Všetky optimalizačné metódy implementujeme v jednej triede nazývanej SortedAlgorithms, ktorá taktiež obsahuje triediace plánovacie algoritmy. Triediace plánovacie algoritmy používajú tieto optimalizačné metódy:

1. Optimalizačnú metódu lineárneho vyhľadávania. Metóda lineárneho vyhľadávania používame v prípade, ak sieť obsahuje nabíjačky, ktoré nevedia dodávať akékoľvek množstvo energie v stanovenom intervale. Napríklad typ nabíjačky DeadbandEVSE nepodporuje dodávky od 0 do 6 ampérov. [15]
2. Optimalizačnú metódu zlatého rezu.

```
def golden_section(
    _index: int, _lb: float, _ub: float, _schedule: np.ndarray
) -> float:
    """ Use the golden section method to find the maximum feasible charging
        rate for the EV. """
    # print('golden section')
    a, b = _lb, _ub
    r = 0.61803
    iterations = 0
    epsilon = 1e-3
    while abs(a - b) > epsilon and iterations < 100:
        x1 = a + (1 - r) * (b - a)
        x2 = b - (1 - r) * (b - a)
        iterations += 1
        _new_schedule = copy.copy(schedule)
        _new_schedule[_index] = (a + b) / 2
        if not infrastructure_constraints_feasible(_new_schedule, infrastructure):
            b = x2
        else:
            a = x1
    _new_schedule = copy.copy(schedule)
    _new_schedule[_index] = (a + b) / 2
    if not infrastructure_constraints_feasible(_new_schedule, infrastructure):
        # to speed up algorithm, it is not necessary to approximate value if not found within it iterations
        return a
    return (a + b) / 2
```

Obr. 4.1: Kód optimalizačnej metódy zlatého rezu.

5 Experimenty

5.1 Ukážka vstupných dát ACN-Data

```
{
  "_id": "5bc90cb9f9af8b0d7fe77cd2",
  "clusterID": "0039",
  "connectionTime": "Wed, 25 Apr 2018 11:08:04 GMT",
  "disconnectTime": "Wed, 25 Apr 2018 13:20:10 GMT",
  "doneChargingTime": "Wed, 25 Apr 2018 13:21:10 GMT",
  "kWhDelivered": 7.932,
  "sessionID": "2_39_78_362_2018-04-25 11:08:04.400812",
  "siteID": "0002",
  "spaceID": "CA-496",
  "stationID": "2-39-78-362",
  "timezone": "America/Los_Angeles",
  "userID": null,
  "userInputs": null
},
```

Obr. 5.1: Vstupné dáta z nabíjania jedného elektromobilu, ktorý prišiel na nabíjaciu stanicu 25. Apríla 2018. Zdroj: [1].

Obrázok vstupných dát 5.1 obsahuje viacero parametrov, ktoré používateľ musí zadať (napr kWhDelivered), aby mohol nabíjať svoj elektromobil na jednej staníc Caltech, JPL alebo Office001. Iné nabíjacie stanice používajú odlišné vstupné dáta, uvádza sa v [16].

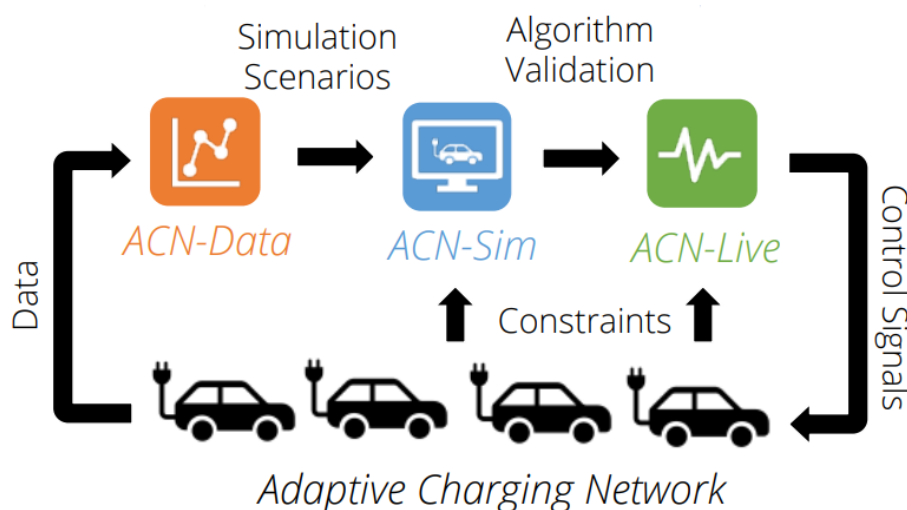
5.2 Využité technológie

V tejto sekcii spomíname technológie, na ktorých spočíva naša implementácia. Balíky popisujeme hlavne na základe informácií uvedených v ich README súboroch a na základe článku [15]. Podrobný návod, ako vytvoriť algoritmy na základe nasledujúcich balíkov sa nachádza v [15].

5.2.1 Balík acnportal.

Balík acnportal obsahuje záznamy nabíjania elektromobilov získane z nabíjacích staníc Caltech, JPL a Office001. Každý záznam nabíjania elektromobilu obsahuje jeho príchod, odchod a požadovanú energiu. Balík acnportal sa nachádza v [9]. Balík acnportal pozostáva z viacerých komponentov:

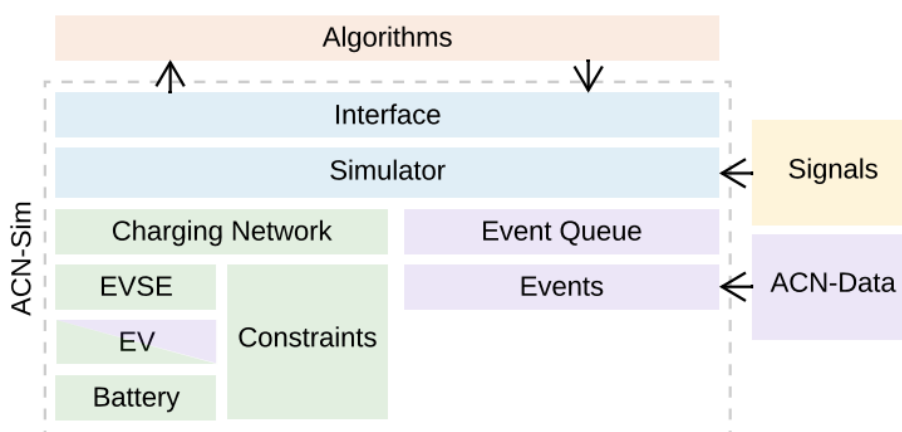
1. **ACN-Data:** Dáta získané z nabíjacích staníc pre elektromobily, konkrétne zo staníc Caltech, JPL a Office001. Vodiči elektromobilov musia cez mobilnú aplikáciu zadať oskenovaný QR kód nabíjačky a potom zadať približný čas odchodu a množstvo požadovanej energie. Dáta získané od vodičov elektromobilov, ale aj dáta slúžiace ku konfigurácii siete sú uložené v relačnej databáze. Vďaka tejto úložnej vrstve vieme vytvárať vizualizácie pre vodičov elektromobilov a pre sieťových operátorov. Ide hlavne o vizualizáciu stavu systému pre sieťových operátorov a stav nabíjania elektromobilov pre vodičov elektromobilov.
2. **ACN-Sim:** ACN-Sim je simulátor používaný na testovanie a overovanie funkcionality algoritmov a systémov. Simulátor zabezpečuje realistické prostredie na overovanie funkcionality algoritmov, hlavne pre výskumníkov, ktorí nemajú prístup k reálnym nabíjacím systémom pre nabíjanie elektromobilov.
3. **ACN-Live:** ACN-Live je hardvér, na ktorom bežia plánovacie algoritmy nabíjania elektromobilov v reálnom čase. Keďže má rovnaké rozhranie ako simulátor ACN-Sim, tak vieme testovať algoritmy implementované v ACN-Sim bez zmeny kódu. [15, 13]



Obr. 5.2: Schéma acnportal. Zdroj obrázka je [15].

Obrázok 5.2 ilustruje interakciu medzi hlavnými komponentami acnportal. Dáta o nabíjaní získava komponent ACN-Data. O obmedzenia siete a validáciu algoritmov sa stará komponent ACN-Sim. V ACN-Live uvádzame konkrétne algoritmy z ACN-Sim do prevádzky. Na takú operáciu netreba meniť kód, lebo ACN-Sim a ACN-Live fungujú cez rovnaké rozhranie. [15, 16]

5.2.2 Architektúra simulátora ACN-Sim



Obr. 5.3: Architektúra simulátora ACN-Sim. Obrázok pochádza z článku [15].

Obrázok 5.3 popisuje architektúru simulátora ACN-Sim. Simulátor ACN-Sim má modulárnu, objektovo orientovanú architektúru. Pod každou krabicou v obrázku 5.3 chápeme základnú triedu, od ktorej môžu dediť nové triedy. Tieto nové triedy môžu napríklad obsahovať nové funkcie.

Simulátor ACN-Sim obsahuje udalosti popisujúce príchod a odchod elektromobilov. Každá udalosť obsahuje informáciu, v ktorom časovom kroku behu simulátora sa má vykonať. V každom časovom kroku behu simulátora sa vykonávajú udalosti v predchádzajúcom časovom kroku alebo udalosti v aktuálnom časovom kroku. Po každej vykonanej udalosti sa spustí plánovací algoritmus a stav infraštruktúry sa aktualizuje. [15]

5.2.3 Balík adacharge.

Balík adacharge obsahuje kontrolný algoritmus MPC, ktorý je schopný riešiť viacero optimalizačných problémov v rámci plánovania nabíjania elektromobilov. Tento MPC algoritmus rieši aj problém, ktorý my riešime, a to problém pridelenia maximálnej energie elektromobilom pri zachovaní obmedzení infraštruktúry pomocou konvexnej optimalizácie. Podrobnejší popis a zdrojový kód balíka Adacharge sa nachádza v [8].

5.2.4 Balík acnportal-experiments.

Balík acnportal-experiment obsahuje experimenty, ktoré využívajú balík acnportal. Tieto experimenty následne vedú zodpovedať výskumné otázky. Pre tvorbu nových experimentov, ktoré riešia náš problém sú podstatné experimenty 1.2 a 2.1. Tieto experimenty využívajú viacero plánovacích algoritmov a konfigurácii siete. Krátky popis experimentov:

1. Experiment 1.2: Cieľom experimentu je porovnať viacero konfigurácií infraštruktúry siete a zároveň plánovacie algoritmy. Tento experiment poukazuje na výhody smart sietí. Jednou z výhod smart sietí je, že vedú riešiť problém dodania maximálneho množstva energie pri zachovaní obmedzení infraštruktúry siete.
2. Experiment 2.1: Cieľom experimentu je porovnať výkonnosť týchto plánovacích algoritmov: round robin, earliest deadline first, least laxity first. Jedným z výsledkov experimentu je koľko percent požadovanej energie dodá elektromobilom každý algoritmus.

Podrobnejší popis a výsledky experimentov vieme nájsť v [7].



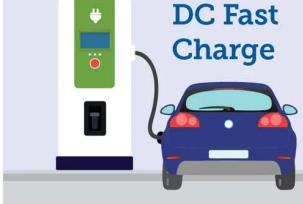
5.3 Konfigurácia a obmedzenia nabíjacích sietí

V tejto sekcii uvádzame jednotlivé konfigurácie a obmedzenia nabíjacích sietí, na ktorých navrhujeme a overujeme model agregátora flexibility. Vysvetľujeme postupne rôzne typy nabíjania, obmedzenia sietí atď.

5.3.1 Typy nabíjania.

Experimenty, ktoré robíme využívajú nabíjanie typu AC level 1 alebo nabíjanie typu AC level 2. Nabíjanie AC level 1 je predovšetkým určené pre vlastníkov elektromobilov, ktorí ich chcú nabíjať celý deň (8 až 10 hodín). Rýchlosť nabíjania pri AC level 1 je 1.4 kWh až do 1.9 kWh.

Typ nabíjania AC level 2 je rýchlejší typ nabíjania ako AC level 1. Rýchlosť nabíjania pri AC level 2 je od 2.5 kWh do 19.2 kWh. Pri takejto rýchlosti nabíjania sa môže vystriedať pri nabíjaní počas dňa viacero spotrebiteľov. [28]

KNOW YOUR EV CHARGING STATIONS		
 <p>AC Level One</p> <p>VOLTAGE 120V 1-Phase AC</p> <p>AMPS 12–16 Amps</p> <p>CHARGING LOAD 1.4–1.9 kW</p> <p>CHARGING TIME 3–5 Miles per Hour</p>	 <p>AC Level Two</p> <p>VOLTAGE 208V or 240V 1-Phase AC</p> <p>AMPS 12–80 Amps (Typ. 32 Amps)</p> <p>CHARGING LOAD 2.5–19.2 kW (Typ. 6.6 kW)</p> <p>CHARGING TIME 12–60 Miles per Hour</p>	 <p>DC Fast Charge</p> <p>VOLTAGE 208V or 480V 3-Phase AC</p> <p>AMPS >100 Amps</p> <p>CHARGING LOAD 50–350 kW</p> <p>CHARGING TIME 60–80 Miles in 20 Minutes</p>

Obr. 5.4: Všetky dnešné typy nabíjania elektromobilov s príslušnými údajmi o rýchlosti nabíjania a kapacite kábla. Zdroj obrázka: [28].

5.3.2 Infraštruktúra nabíjacej stanice.

Simulátor ACN-Sim využíva inštancie triedy `ChargingNetwork` na modelovanie infraštruktúry nabíjacej siete. To znamená, že trieda `ChargingNetwork` modeluje nabíjačky (napríklad ich počet, typ nabíjačiek atď), transformátor (určený na prenos energie v nabíjacej sieti), prepínacie panely a káble.

Na zmenu alebo rozšírenie funkcionality triedy `ChargingNetwork` je nutné vytvoriť novú triedu, ktorá dedí od triedy `ChargingNetwork`. Takto bola vytvorená trieda `StochasticNetwork`, ktorá sa odlišuje od triedy `ChargingNetwork` v pridelovaní nabíjačiek prichádzajúcim elektromobilom. Popis, ako sa pridelujú nabíjačky elektromobilom:

ChargingNetwork

V tomto type nabíjacej siete je každý prichádzajúci elektromobil priradený vopred priradený určenej nabíjačke v nabíjacej sieti.

StochasticNetwork

Tento typ nabíjacej siete prideluje každému prichádzajúcemu elektromobilu voľnú nabíjačku náhodne. V prípade, keď príde elektromobil do nabíjacej stanice a žiadna nabíjačka v nabíjacej stanici nie je voľná, tak elektromobil pridá do na koniec čakacieho radu. Potom v okamihu, keď sa nabíjačka uvoľní, tak sa priradí prvému elektromobilu v čakacom rade

(ktoré sa z čakacieho radu odstráni).

Typ nabíjacej siete StochasticNetwork je viac vhodný než typ nabíjacej siete Charging-Network hlavne pre uplatnenie v praxi, ale aj v pri generovaní udalostí zo štatistických modelov. [15]

Keď že my chceme, aby naše experimenty boli aplikovateľné v reálnom živote, tak použijeme typ siete StochasticNetwork v experimentoch. Tiež používame v našim experimentoch len nabíjačky, ktoré povolia prenos akéhokoľvek množstva energie medzi dolnou a hornou hranicou množstva energie.

5.3.3 Obmedzenia pri nabíjaní.

Nabíjacie systémy často fungujú na princípe radiálnych sietí. Musíme preto obmedziť množstvo voltov prechádzajúcich cez každé úzke miesto siete. Pomocou Kirchhoffových zákonov vieme definovať obmedzenia pri nabíjaní takto:

$$|I_j(t)| = \left| \sum_{i=1}^N A_{ij} r_i(t) e^{j\phi_i} \right| \leq R_j, \quad (5.1)$$

kde R_j je veľkosť prúdu, $I_j(t)$ je prúd prúdiaci cez úzke miesto siete, N je počet nabíjačiek v nabíjacej stanici, $r_i(t)$ je prúd poskytovaný nabíjačkou i v čase t . Dokopy máme T časových krokov. Parametrom ϕ_i vyjadrujeme fázový uhol pre aktuálny fázor, ktorý závisí na tom ako je nabíjačka i zapojená do siete. [15]

UNFINISHED

5.4 Prvý experiment

5.5 Druhý experiment

5.6 Tretí experiment

Záver

Literatúra

- [1] Caltech. Acn-data. <https://ev.caltech.edu/dataset>, 2023. Accessed: 3rd December, 2023.
- [2] Niangjun Chen, Christian Kurniawan, Yorie Nakahira, Lijun Chen, and Steven H. Low. Smoothed least-laxity-first algorithm for ev charging, 2021.
- [3] Collimator. What is a control algorithm? <https://www.collimator.ai/reference-guides/what-is-a-control-algorithm#:~:text=Definition%20of%20a%20control%20algorithm&text=These%20algorithms%20are%20designed%20to,is%20either%20undesirable%20or%20impossible.>, 2023. Accessed: 3rd December, 2023.
- [4] Collimator. What is model predictive control? <https://www.collimator.ai/reference-guides/what-is-model-predictive-control>, 2023. Accessed: 3rd December, 2023.
- [5] Pedro Crespo del Granado, Jayaprakash Rajasekharan, Surya Venkatesh Pandiyan, Asgeir Tomasgard, Güray Kara, Hossein Farahmand, and Stefan Jaehnert. Flexibility characterization, aggregation, and market design trends with a high share of renewables: a review. *Current Sustainable/Renewable Energy Reports*, 10(1):12–21, Mar 2023.
- [6] International Energy Agency (IEA). Global ev outlook 2023, 2023. License: CC BY 4.0.
- [7] Z. J. Lee and S. Sharma. acnportal-experiments. <https://github.com/caltech-netlab/acnportal-experiments>, Dec. 2023.
- [8] Z. J. Lee and S. Sharma. adacharge. <https://github.com/caltech-netlab/adacharge>, Dec. 2023.
- [9] Z. J. Lee, S. Sharma, and D. Johansson. acnportal. <https://github.com/zach401/acnportal>, Dec. 2023.
- [10] Zachary Lee, John Pang, and Steven Low. Pricing ev charging service with demand charge. *Electric Power Systems Research*, 189:106694, 12 2020.

- [11] Zachary J. Lee, Daniel Chang, Cheng Jin, George S. Lee, Rand Lee, Ted Lee, and Steven H. Low. Large-scale adaptive electric vehicle charging. *2018 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*, pages 1–7, 2018.
- [12] Zachary J. Lee, Daniel Johansson, and Steven H. Low. Acn-sim: An open-source simulator for data-driven electric vehicle charging research. In *2019 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*, pages 1–6, 2019.
- [13] Zachary J. Lee, George Lee, Ted Lee, Cheng Jin, Rand Lee, Zhi Low, Daniel Chang, Christine Ortega, and Steven H. Low. Adaptive charging networks: A framework for smart electric vehicle charging, 2020.
- [14] Zachary J. Lee, Tongxin Li, and Steven H. Low. Acn-data: Analysis and applications of an open ev charging dataset. In *Proceedings of the Tenth ACM International Conference on Future Energy Systems, e-Energy '19*, page 139–149, New York, NY, USA, 2019. Association for Computing Machinery.
- [15] Zachary J. Lee, Sunash Sharma, Daniel Johansson, and Steven H. Low. Acn-sim: An open-source simulator for data-driven electric vehicle charging research, 2021.
- [16] Zachary Jordan Lee. *The Adaptive Charging Network Research Portal: Systems, Tools, and Algorithms*. Dissertation, California Institute of Technology, 2021.
- [17] Tongxin Li, Bo Sun, Yue Chen, Zixin Ye, Steven H. Low, and Adam Wierman. Learning-based predictive control via real-time aggregate flexibility. *IEEE Transactions on Smart Grid*, 12(6):4897–4913, nov 2021.
- [18] Sofie Mannerstråle. Flexibility aggregators in the future energy system. <https://www.capgemini.com/se-en/insights/research-library/flexibility-aggregators-in-the-future-energy-system/>, 2023. Accessed: 4th January, 2024.
- [19] microcontrollerslab. Least laxity first (llf) scheduling algorithm. <https://microcontrollerslab.com/least-laxity-first-llf/>, 2023. Accessed: 3rd December, 2023.
- [20] Joy Chandra Mukherjee and Arobinda Gupta. A review of charge scheduling of electric vehicles in smart grid. *IEEE Systems Journal*, 9(4):1541–1553, 2015.
- [21] Lothar Reichel. Lecture 8: Optimization. Accessed: 13th February, 2023.
- [22] Vijayshree Shinde and Seema C. Comparison of real time task scheduling algorithms. *International Journal of Computer Applications*, 158:37–41, 01 2017.

- [23] Wanrong Tang, Suzhi Bi, and Ying Jun Angela Zhang. Online charging scheduling algorithms of electric vehicles in smart grid: An overview. *IEEE Communications Magazine*, 54:76–83, 2016.
- [24] TechTarget. round robin. <https://www.techtarget.com/whatis/definition/round-robin>, 2023. Accessed: 3rd December, 2023.
- [25] USEF. Usef white paper - flexibility value chain 1.0. Technical report, Arnhem, 2018.
- [26] Jake VanderPlas. *Python data science handbook : essential tools for working with data*. O'Reilly Media, Inc, Sebastopol, CA, 2016.
- [27] Steven H. Low Zachary J. Lee, John Z.F. Pang. Pricing ev charging service with demand charge. <https://resnick.caltech.edu/documents/17395/RSIonline-RH-Lee-Z-SEPT-2020.pdf>, 2020. Accessed: 3rd December, 2023.
- [28] ZDWL. Ev charger levels. <https://zdwl-tec.com/news/ev-charger-levels/>, 2023. Accessed: 3rd December, 2023.

Príloha A: obsah elektronickej prílohy

Príloha B: Používateľská príručka