



Universitetet
i Stavanger

DET TEKNISK-NATURVITENSKAPELIGE FAKULTET

BACHELOROPPGAVE

Studieprogram/spesialisering: Bachelor i ingeniørfag / Automatisering og elektronikkdesign	Vårsemesteret, 2021 Åpen
Forfatter: Jakob Sønneland Tislevoll, Gunnar Sveinsvoll Jonassen	
Fagansvarlig: Sven Ole Aase Veileder(e): Ståle Freyer, Sven Ole Aase	
Tittel på bacheloroppgaven: Energibehandlingssystem for korttidslagring av energi i private boliger Engelsk tittel: Energy management system for short term storage of energy in private residences	
Studiepoeng: 20	
Emneord: Energibehandlingssystem Optimalisering Lineær programmering Peak Shaving Balansering av strømnettet Batterilagring	Sidetall: 102 + vedlegg/annet: 61 Stavanger, 15. mai 2022 dato/år

Energibehandlingssystem for korttidslagring av energi i private boliger

Jakob Sønneland Tislevoll (242159), Gunnar Sveinsvoll Jonassen (254674)

May 15, 2022

Sammendrag

Oppgaven er en samling av teoretiske kasusstudier av et energibehandlingssystem (**EBS**) for ulike forbruksmønstre, strømprisser, batteripakker, og tilkoblede energikilder. Et energibehandlingssystem (**EBS**) bestående av et analyseverktøy og en reguleringsmetode for overvåking og kontroll av sammensatte energisystem utvikles med et overordnet formål om å redusere strømpris, lagre overflødig energi, stabilisere strømtrekk og minske klimaavtrykk. Systemet må produsere en avkastning på en tidshorisont som rettferdiggjør den initiale investeringen. Besparelse innføres som et mål på fortjeneste etter investering relativt til uten installert **EBS**. Ytelsen til **EBS** vurderes ut i fra hvilken grad overordnede formål oppnås.

Analyseverktøyet presenterer relevant data. Det gir mulighet for sammenlikning og konfigurering av **EBS**. Reguleringsmetoden bestemmer optimal ladesyklus for en batteripakke installert i en privat bolig. Den tar inn datasett for strømforbruk, tilhørende strømpriser og egen energiproduksjon. Reguleringsmetodens kjerne er det lineære optimaliseringsproblemet, bestående av en likning for total kostnad som minimeres innenfor lineære begrensninger.

Reguleringsmetoden har to algoritmer. Algoritmene skilles i metoden de behandler samme datasett på, og representerer hvert sitt tilfelle. I det ene tilfellet representerer resultatene som følger et ideelt **EBS**, og algoritmen har nøyaktige data på fremtidig strømforbruk og produksjon. I det andre tilfellet representerer resultatene som følger et realistisk **EBS**, og tilgangen på datasettet er begrenset til hva som ville vært tilgjengelig i praksis. I sistnevnte tilfelle forsøker algoritmen å predikere fremtiden med begrenset data, og danner et realistisk **EBS** kalt *Sanntidskjøring* i denne oppgaven.

Ideelt **EBS** vektlegges og vurderes i størst grad. Hvordan ytelsen til **EBS** påvirkes av ulike forbruksmønstre, strømprisser, batteristørrelser og tilkoblede energikilder brukes for å bestemme om det er mulig med optimal utnyttelse av **EBS** å oppnå formålet med oppgaven. Informasjonen kan i tillegg brukes under utviklingen og planleggingen av mer avansert sanntidsalgoritmer dersom resultatet av ideell optimalisering tilatter lønnsom installasjon.

Realistisk **EBS** tilegnes et eget kapittel *Sanntidskjøring*. Målet er å komme så nær som mulig den ideelle løsningen, som brukes til sammenlikning med realistisk **EBS**. Avvik fra ideell løsning undersøkes for å kartlegge svakhetene og forbedringspotensiale for realistisk løsning. Resultatet ble at reduksjon i strømpris for realistisk **EBS** uten tilkoblet energikilde avvirket fra optimal **EBS** med 0,05%, og med tilkoblet energikilde avvirket med 4,2%.

Analyseverktøyet til energibehandlingssystemet (**EBS**) utviklet i denne oppgaven produserer resultater som tyder på at reguleringsmetoden reduserer årlig strømkostnad mellom 2,9%-22% avhengig av forbruksmønster, strømprisene, batteripakkens størrelse og tilkoblet energikilde. Den årlige besparelsen i EUR varierte fra 111 EUR - 527 EUR. Besparelsen i forhold til å ikke ha **EBS** installert var større for alle tilfeller med tilkoblet energikilde. Det ble funnet at strømtrekket fra strømmettet var motsyklisk, som betyr at belastningen på strømmettet reduseres når **EBS** er installert.

Abstract

This paper is a collection of theoretical case-studies of an energy management system (**EMS**) for different usage patterns, price zones, battery packs and connected energy sources. An energy management system (**EMS**) comprised of a tool for analysis and a regulating method for surveillance and control of modular energy systems is developed with an overall goal to reduce electricity prices, store excess energy, stabilize loads and reduce environmental impact. The system must produce a return in a time that justifies the original investments. Saving is introduced as a measure of profitability of the investment relative to not installing **EMS**. The performance is assessed as the degree to which the overall goals are achieved.

The analysis tool presents relevant data. This can be used to compare results or configure **EMS**. The regulation method decides an optimal charging cycle for a battery pack installed in a private home. Inputs are power-usage, associated power prices and energy production. The core of the regulation method is the linear optimizationproblem, consisting of an equation for total cost which is minimized within linear constraints.

The regulating method has two algorithms. The algorithms are separated by different methods of treating the same set of data, and represent different cases. In one case, the results represent what would be an ideal **EMS**, and the algorithm has access to precise data on future power consumption and production. In the other case, the results represent a realistic **EMS**, and the access to precise data is restrained to what would be practically available. In the latter case, the algorithm attempts to predict the future using limited data, and form the realistic **EMS** referred to as *real-time operation* in this paper.

The ideal **EMS** is emphasized and evaluated the most. How the **EMS** performance is affected by different usage patterns, price zones, battery packs and connected energy sources, is used to determine the possibility of an optimal usage and configuration of **EMS** to achieve the overall goals. The information can also be used during development and planning of more advanced real-time algorithms given that the results from ideal optimization argue for profitable installation of **EMS**.

Realistic **EMS** is dedicated to it's own chapter, *Real-time operation*. The goal is to come as close as possible to the solution of the ideal **EMS**, which is used to compare to realistic **EMS**. Deviations from the ideal solution are recorded and investigated in order to map the weaknesses and potential for improvement of realistic **EMS**. In conclusion, the reduction in electricity cost for realistic **EMS** without external energy source deviate from the ideal **EMS** with 0,05%. Adding an external energy source results in a 4,2% deviation from ideal **EMS**

The analysis tool developed in this paper produces results that suggest that the regulation method reduces yearly electricity prices between 2,9%-22% dependent on usage patter, price zone, size of battery pack and external energy source. The yearly savings in EUR varied from 111 EUR - 527 EUR. Savings compared to not having installed **EMS** were larger for every case using an external energy source. A countercyclical current draw indicates that the load on the

grid is reduced using **EMS**.

Innholdsfortegnelse

	6
1 Innledning	8
1.1 Bakgrunn og motivasjon	8
1.2 Definere problemet	9
1.3 Mål og metode	11
1.4 Antagelser og begrensinger	13
1.5 Modell av mikronett	14
2 Bakgrunnsstoff	15
2.1 Tidligere forskning innenfor energibehandlingssystem (EBS)	15
2.2 Batteriteknologi	19
2.3 Inverter	25
2.4 Strømmerketet	27
3 Relevant data	32
3.1 Strømpris	32
3.2 Forbruk	43
3.3 Solceller	46
3.4 Inverter karakteristikk	49
3.5 Batteri karakteristikk	50
4 Optimaliseringsproblem	51
4.1 Valg av algoritme	51
4.2 Generell formulering	52
4.3 Formel for bruk av Bestemmelsevariabler	57
5 Resultat	58
5.1 Ytelse med batteristørrelse på 13.5kWh for enebolig i pris-sone Kristiansand 2021	58
5.2 Ytelse med batteristørrelse på 13.5kWh for enebolig med solceller i pris-sone Kristiansand 2021	64
5.3 Ytelse med batteristørrelse på 36 kWh for enebolig med solceller i pris-sone Kristiansand 2021	70
5.4 Sanntidskjøring	75
6 Økonomisk oversikt	79
7 Miljøregnskap	80
8 Diskusjon	82
8.1 Ulike forbruksmønstre og prissoner	82
8.2 Sanntidskjøring	92
8.3 Systemets innvirkning på strømnettet	92

8.4	Framtidsutsikter	94
8.5	Potensielle feilkilder	95
8.6	Videre arbeid	96
9	Konklusjon:	97
10	Vedlegg	102
10.1	Opprinnelig oppgavebeskrivelse	102
10.2	Notater	105
10.3	Tekniske begrensninger	119
10.4	strømmarkedet	120
10.5	Kode optimalisering	122
10.6	Kode realtid-kjøring	128
10.7	Kode GUI	136

Forord

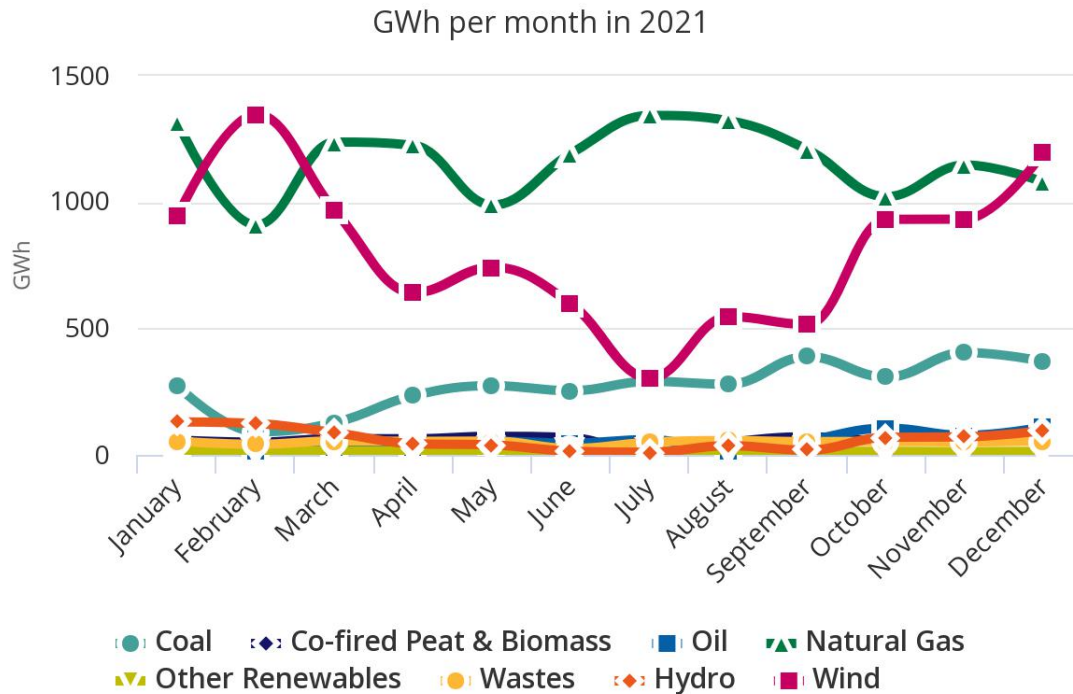
En stor takk til professor Sven Ole Aase og sjefingeniør Ståle Freyer ved IDE-UiS for et veldig godt samarbeid og fremragende veiledning. Takk til Envyon, Statnett, Nord Pool og Agder Energi for opplysende samtaler og støtte. Denne oppgaven er et åndsverk som markerer slutten på en 3-årig utdanning i automatisering og elektronikkdesign ved Instituttet for Data- og Elektroteknikk, Universitetet i Stavanger. All innlært teori og praktisk erfaring forsøkes samlet til en verktøykasse som med fremtidrettet ånd benyttes til å definere og løse et voksende problem innenfor kraft- og husholdningsmarkedet.

1 Innledning

1.1 Bakgrunn og motivasjon

Verden skal bort fra fossilenergi og over på fornybar energi. Denne overgangen skaper både muligheter og potensielle problemer. Strømnettet blir i økende grad koblet opp mot ikke-fleksible, men fornybare energikilder. Kraften som produseres fra fornybare kilder avhenger av været, i motsetning til kull, gass og olje hvor produksjonen kan justeres i forhold til behovet til en hver tid. Strømmerket globaliseres, og Norge eksporterer egenprodusert strøm utover Europa gjennom undersjøiske kabler som NordLink og NorthSeaLink. Uforutsigbar kraftproduksjon og eksportavtaler legger press på landets kraftforsyning. Det medfører et behov for regulering og stabilisering av ekstremer på strømmarkedet og strømmettet. Behovet for lagring av energi, over kort og lang tidshorisont blir således styrket. 13. mai 2022 er *day-ahead* prisene på *Nord Pool* for Kristiansand(**NO2**) 6,48 EUR/MWh mellom 14:00-15:00 og 143,9 EUR/MWh mellom 16:00-17:00. Eksempelet er ikke enestående, men representativt for energisituasjonen i Europa.

Ikke-fleksibel, fornybar kraftproduksjon suppleres globalt i økende grad med fleksibel produksjon fra gasskraftverk i perioder med lite vind eller sol. Kraftproduksjon fra gass har dermed økt i takt med fornybar kraftproduksjon [13].



Figur 1: Irlands kraftproduksjon fordelt på kilde for året 2021[12].

Energibyrådet *European Union Agency for the Cooperation of Energy Regulators (ACER)* er med på å binde norsk kraftpolitikk og strømmarkedet til det europeiske. Gjennom Parisavtalen er Norges klimamål å redusere utslipp med minst 50% innen 2030[41]. Europas mål om utslippsreduksjon krever en hurtig omstilling av kraftproduksjon til fornybare energikilder. Overgangen medfølger foruten volatile strømpriser også et tydelig behov for effektiv regulering. Nettopp dette er motivasjonen i oppgaven.

1.2 Definere problemet

Prissvingningene på strømmarkedet varierer med historisk høy frekvens. Differansen mellom prisene er historisk høy. Utbygging av overføringsnett i Norge har blitt nedprioritert til fordel for utenlandskabler, noe som skaper store prisvariasjoner innad i Norge. Mange land er midt i et grønt skifte og går vekk fra kull og kjernekraft. Mer uregulerbar fornybar energi øker prissvingningene. I Norge er vi fortsatt veldig væravhengige på grunn av vannkraft og internasjonale strømkabler [20]. Konsekvenser for privatpersoner kan potensielt være uforutsigbarhet og økonomisk press. Byrden på privatpersonen avhenger av livssituasjon og bosituasjon. Et individ kan foreta strømbesparende tiltak som utkobling av hvitevarer og oppvarming, slukking av lys eller benytte seg av døgnets laveste priser. Slike tiltak letter ikke alltid byrden på privatpersonen tilstrekkelig.

Flere ikke-fleksible energikilder gir en større belastning på strømmettet og bidrar ytterligere til uforutsigbarhet. Teknisk må strømmettet kunne operere uten for store svingninger i frekvens, unngå flaskehals i strømovertføring og være tilrettelagt for innføring og videreutvikling av fornybare energikilder. På nasjonalt nivå vurderes det nødvendig med installering av netttilknyttede, storskala batterisystemer[22]. Motivasjonen for tilsvarende løsninger i husholdningsmarkedet er hovedsaklig å redusere strømregningen, men også å bidra til optimalisering av de tekniske utfordringene til strømmettet på nasjonalt nivå. Markedet for privatpersoner med gunstige forretningsmodeller må derfor eksistere for å oppmuntre til installasjon av *energibehandlingssystem (EBS)* i husholdningsmarkedet.

Energi kan lagres og brukes avhengig av behov. Langtidslagring av energi har blant annet blitt realisert gjennom hydrogenproduksjon. Et mulig alternativ i Norge og andre land med kupert terreng som fjorder og daler er pumpe-vannkraftverk der strøm pumpes tilbake i magasinet når prisene er lave. Da fungerer vannmagasinet som et batteri, med et energitap ved pumping av vann og gjennom fordampning.

Denne oppgaven tar for seg korttidslagring av energi på et batteri. Hvilket type batteri som er mest hensiktsmessig vurderes utifra kriterier i kapittelet om batteriteknologi 2.2. Det er viktig å nevne at det varsles global mangel på essensielle grunnstoff til batteriproduksjon som nikkel, kobolt[58] og litium[55] med flere. Eterspørselen etter elbiler har fordoblet seg på verdensbasis i 2021[2] og ser ikke ut til å stoppe der[1]. Selv om litium-ion batteri vurderes som mest relevant for den teoretiske gjennomførselen i denne oppgaven 2.2, kan fremtiden by på utfordringer innenfor tilbud og teknologiske fremskritt som gjør andre batteriteknolgieer mer relevante. Her kan teknologier som *Vanadium Redox Flow Batteri (VRFB)* og *Solid-State* batteri nevnes som høyst aktuelle. El-bil industrien satser på *Solid-State*[40], og det gjøres store fremskritt innenfor forskning på *VRFB* i *EBS* [9]. Mer om dette i 8.

1.3 Mål og metode

Batteri som virkemiddel for å stabilisere strømmettet er kjernen av denne oppgaven. Kraftflyt skal reguleres inn og ut av et batteri. Hvordan kraftflyten reguleres bestemmes av en algoritme, som skal basere seg på batteriets indre egenskaper så vel som viktige eksterne parametre. Strømpris, vær og egenproduksjon av kraft er noen av parametrene som tenkes regulert. Hvor stor påvirkning hver av faktorene har for stabilisering av strømmettet, og grad av fortjeneste, kan tenkes å avhenge av størrelsen på **EBS**. Derfor er valg av batteriteknologi og batterikapasitet essensielt. Funksjonen algoritmen i **EBS** skal utøve er å bruke mest mulig av egenprodusert kraft, og selge minst mulig tilbake. På den måten skal det forsøkes å oppnå en mer stabil og forutsigbar kraftflyt, samt mindre volatile priser. Andre fordeler kan være sikkerhet mot strømbrudd. I begge tilfeller skal den mest hensiktsmessige konfigurasjonen vurderes utifra kravene:

- Reduksjon i strømpris(% og EUR)
- Lagre overflødig energi
- Stabilisere strømtrekk
- Nedbetalingstid på systemet
- Minske klimaavtrykk

Metoden for å lage energibehandlingssystem samt undersøke kravene til batteripakken er å utvikle:

- Analyseverktøy
- Reguleringsmetode

Målene og metoden danner rammeverket for **EBS** som skal utvikles ved å:

- Hente relevant data
- Velge ut komponenter
- Definere optimaliseringsproblem
- Utvikle python-program
- Anvende resultat til diskusjon

En foreslått algoritme med tilhørende programvare skal utvikles basert på tidligere forskning. Parametrene blir (Se antakelser og begrensninger):

- Strømpris (svingninger på strømmettet)
- Kraftproduksjon
- Forbruk
- Nettleie
- Degraderingkarakterstikk
- Inverterkarakterstikk
- Ladekurve

Målet er å undersøke lønnsomheten av et **EBS** tilkoblet private hjem. Både i forbindelse med egenprodusert energi samt enkeltstående system med kun batteripakke. Det skal undersøkes hvilken batteristørrelse som er mest hensiktsmessig med hensyn på utslipp under produksjon og kortest mulig nedbetalingstid. Det skal utvikles en algoritme som skal utnytte kraften som blir produsert, samt svingningene på strømmettet slik at strømtrekket blir jevnere og toppene reduseres. Programkode skal utvikles for å anvende algoritmen og presentere reguleringsresultat. Deretter diskuteres og sammenliknes resultat.

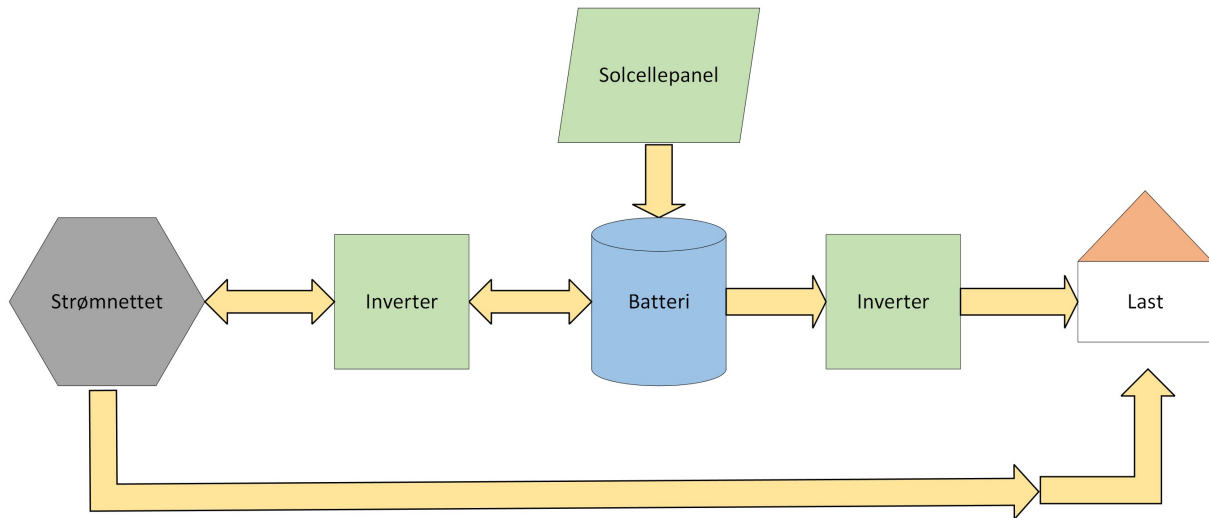
1.4 Antagelser og begrensinger

Ett batteri med én kraftforsyning, ikke et nettverk (som ville vært ideelt nasjonalt/globalt) rettet mot privatpersoner.

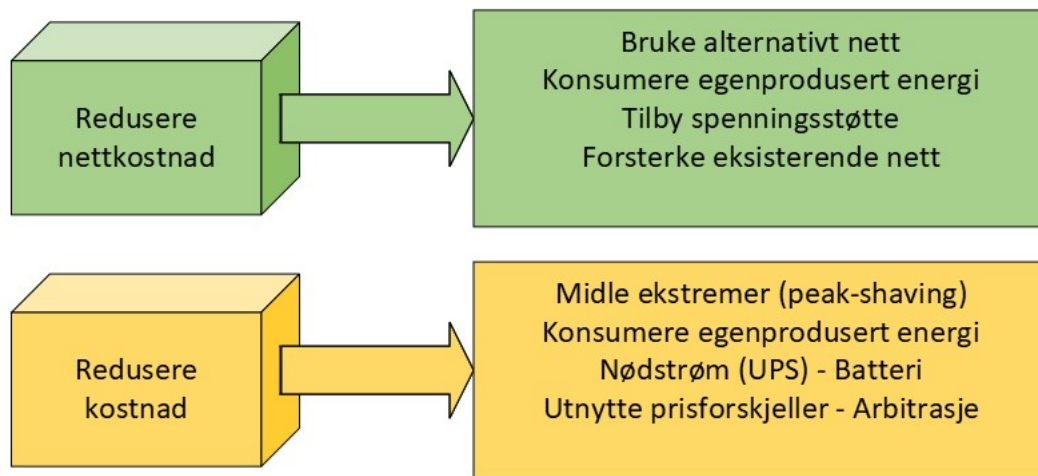
- Oppsettet skal bestå av solcellepaneler, inverter (AC-DC og DC-AC) og en batteripakke.
- Det skal være mulig å kjøpe strøm fra strømmettet og lagre på batteri
- Strøm fra solcellene kan enten lagres på batteriet, bli brukt av lasten, eller selges.
- Strøm lagret på batteriet skal kunne selges på strømmettet dersom det er nødvendig, men skal ideelt bli brukt av lasten (Dette fordi avgifter gjør det økonomisk ugunstig å selge)
- Historisk *Day-ahead* strømpris brukes, og algoritmen skal basert på forventet produksjon fra solcellene, forventet prissvingning, forventet last, og tilgjengelig kapasitet på batteriet, bestemme strømflyten.
- Det er ønskelig med et forhold mellom batterikapasitet og effekt fra solcelle slik at mer enn 5 timer med full produksjon kan lagres på batteriet.
- System skal kjøres over flere år for å undersøke hvordan systemet arbeider under ulike forhold. (lite vind i Europa, mye vind i Europa, kald vær, varmt vær, ulike strømtrekk fra det nasjonale strømmettet).

1.5 Modell av mikronett

MIP-metoden optimaliserer for variabelen **opplading** og resultatet blir brukt til å beregne og presentere ytelsen til **EBS**.



Figur 2: Oversikt over systemet



Figur 3: Oversikt over funksjonalitet

2 Bakgrunnsstoff

2.1 Tidligere forskning innenfor energibehandlingssystem (EBS)

Tidligere forskning på **EBS** er omfattende. Siden årtusenskifte har forskning på mikronett med tilkoblede energikilder og mulighet for energilagring blitt særdeles relevant. Det finnes mange ulike måter å modellere et slikt system på, alt gjort forholdsmessig til sitt bruk. Konseptet med mikronett er å ha et lokalt og regulert strømmnett som opererer uavhengig av hovedstrømnettet. Slike mikronett kan være helt selvforsynte, eller komplementære til hovedstrømnettet. Et eksempel på et fullstendig selvforsynt mikronett utredes i en forskningartikkel kalt *12 years operation of the Gaidouromantra Microgrid in Kythnos island*, utgitt i 2012 av Stathis Tselepis.[51]. Mikronettet omtalt av Tselepis har vært operativt siden 2001, og var blant de første av sin sort. Det består av 12 sommerhus, og er ikke tilkoblet et offentlig strømmnett. Mikronettet er dermed isolert og selvforsynt med solcellepanel som hovedstrømkilde. En dieselgenerator er også satt opp til nødbruk. Hver husstand er utstyrt med en intelligent lastkontroller, hvorav flere er tilkoblet både solcellepanel og batteri. Enkelte husstander er bare koblet rett på mikronettet.

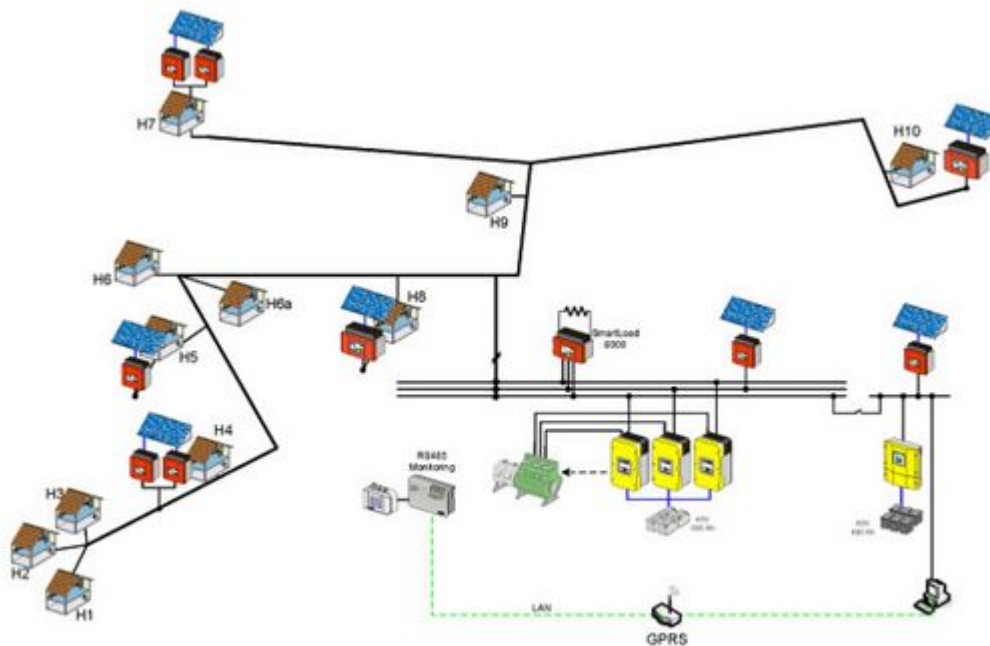


Figure 2: Single line schematic diagram of Gaidouromantra microgrid, Kythnos island.

Figur 4: Konfigurering av "Gaidouromantra microgrid" i Kythnos. Hentet fra [51]

I dette mikronettet er det brukt batteri-invertere til å utføre energibehandling ved å regulere frekvens, enten for å oppnå belastningsreduksjon på nettet eller effektreduksjon av solcellene. Altså brukes frekvens her som kommunikasjonsmetode mellom lastene og kildene.

Kommunikasjon og kontroll av kilder og last er hva et **EBS** skal utføre. Forskingen på dette området har tatt fart siden mikronettet på Kythnos ble operativt, og alternative metoder for **EBS** har blitt utviklet. Frem til 1990-tallet, var de fleste **EBS** i bruk basert på selskapers egenutviklede maskinvare og operativsystem. Innen 2006 kunne forbruker benytte seg av *linux*-, *windows*- og *UNIX*-baserte **EBS**. I dag kan privatpersoner bruke gratis programvare til å utvikle sine egne metoder.

Flere forskningapirer på det spesifikke problemet vi ønsker å løse har blitt publisert i nyere tid, blant annet et kalt *Mixed-Integer-Linear-Programming-Based Energy Management System for Hybrid PV-Wind-Battery Microgrids: Modeling, Design, and Experimental Verification* fra bidragyttere på Universitetet i Aalborg.[30]. Her vektlegges selve modelleringen og utformingen av problemet som et lineært optimaliseringsproblem. I seg selv er *Mixed Integer Programming (MIP)* problem nå et veletablert forskningsfelt med mulighet for å modellere virkelige situasjoner. I skrivende stund er *IBM ILOG CPLEX Solver*, *Gurobi Solver* og *COIN-ORs CBC 2.0* de mest aktuelle *MIP*-løserne, og pakken *python-MIP* som benyttes som løser i denne oppgaven bruker løseren *CBC 2.0*. I boka *Hybrid Metaheuristic* skriver forfatter Andrea Lodi om *The heuristic (Dark) Side of MIP Solvers* i et kapittel[29]. Det fremkommer at løserne i økende grad er heuretiske i seg selv, med heuretiske bestemmelser og snarveier skjult innenfor løseren og med et bruksområde som i økende grad er heuretisk. *MIP*-løserne foreslås derfor å sees på som et åpent rammeverk som kan utnyttes med få restriksjoner til algoritmen den utfører, altså programkoden.

Hvordan optimaliseringsproblemet best mulig kan modelleres er en del av kjernen i denne oppgaven. Gjennom kontakt med oppstartsbedrifter i feltet som *RePack* da vi begynte oppgaven, som nå heter *Evyon*[18], har vi fått innsikt i dagsaktuelle og kommersielle problemstillinger. Medstifter og CEO av *Evyon*, Jørgen Erdal, publiserte i 2017 sin masteroppgave på *NTNU* kalt *Stochastic Optimisation of Battery System Operation Strategy under different Utility Tariff Structures*. Denne forskningen hadde en stokastisk optimalisering, og gjorde dette med hensyn på fremtidige endringer i nettleiestrukturen som nå er særdeles dagsaktuelt.[17].

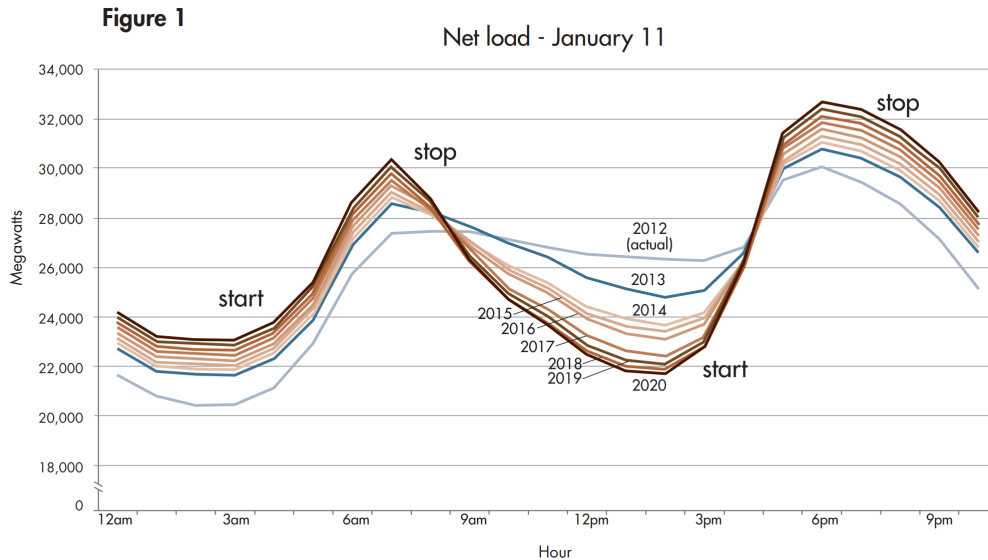
I forbindelse med programkoden og optimeringsalgoritmen må et grafisk brukergrensesnitt utvikles for å kunne endre inngangsvariablene til **EBS** og presentere relevant informasjon på en god måte. *EMD International* hevder å tilby ledende programvare på dette området kalt **energyPRO**[26], som blant annet *Statkraft AS* eier lisenser til. Programvaren tilbyr modellering og analyse av en stor variasjon komplekse energiprojekt, med en velutviklet modularitet og tekno-økonomisk analyse. Optimeringsalgoritmen brukt av denne industriledende programvaren fremkommer i en *how-to guide* publisert av *EMD International* kalt *The MILP Solver optimization method in energyPRO*.[27]. Her brukes spesifikt **MILP** som løsningsmetode, som innebærer en lineær objektfunksjon med tilsvarende lineære begrensninger. Løseren som tilbys fra **energyPRO** er standard **CBC**, med mulighet for tilkobling av betalt Gurobi-løser. Disse løserne har mulighet for å løse **MIP** problem som ikke nødvendigvis er lineære, og har algoritmer for eksempelvis kvadratiske betingelser, eller i Gurobis tilfelle også ikke-lineære ob-

jektfunksjoner. Graden av kompleksitet påvirker løserenes nøyaktighet, og de blir i tilfeller nødt til å ta heuristiske beslutninger. **MIP**-løseres optimeringsevne kan tenkes å være avgjørende for **EBS**ets reguleringsevne og lønnsomhet. Dermed kan det være et betydelig forbedringspotensial tilknyttet **EBS** som avhenger av løserenes presisjon.

I stor grad har **EBS** med optimaliseringsproblem benyttet seg av *Mixed-Integer-Linear-Programming (MILP)*. Det innebærer at objektfunksjonen som skal optimeres er lineær. Noen av bestemmelsesvariablene må være heltall, men ikke alle, og betingelsene lineære. Foruten **energyPRO**, har Maik Naumann og Nam Truong sammen med Marc Moeller og Daniel Kucevic ved *Det Tekniske Universitetet i Münchens (TUM) Institutt for Elektrisk Energilagringsteknologi (EES)* utviklet *SimSES - Software for techno-economic simulation of stationary energy storage systems*.^[15] **SimSES** kommer med en verktøykasse for optimalisering som benytter seg av ren lineær programmering (LP), altså uten mulighet for bestemmelsesvariabler som ikke er heltall, eller ikke-lineære begrensninger. Programvaren kommer med åpen kildekode, og har blitt utgitt som pythonkode etter først å ha blitt utviklet i MATLAB.

Maskinlæring og kunstig intelligens for å kunne estimere fremtidig forbruk og strømpris er i utvikling i dag. PhD-kandidat Fredrik Skaug Fadnes har startet et forskningsprosjekt kalt *Smart solutions for energy systems of the future* på akkurat dette. Gjennom forskningsgruppen for *Applied AI for Energy Systems Engineering* ved Instituttet for energi- og petroleumsteknologi på Universitetet i Stavanger, skal det undersøkes AI-baserte verktøy og metoder for overvåkning, analyse, design og kjøring av energisystem. Et av de tre målene til prosjektet er å analysere modeller for å kunne forutse produksjon, etterspørsel og status på energisystem. Et av de tre sentrale spørsmålene som skal besvares er hvordan AI-baserte **EBS** for kontroll av tilførsel og etterspørselssiden påvirker faktorer som energikonsum, kostnader og utslipp av drivhusgass. Resultatene fra dette forskningsprosjektet og liknende internasjonale prosjekt vil være verdifulle for videreutvikling av smarte energisystem, og kan bidra til å gjøre realtidskjøring av **EBS** lønnsomt for private husholdninger såvel som kommersielle produksjonsanlegg.

California er et område som kan brukes som eksempel på konsekvensen av høy andel ikke-fleksibel energi i nettet. Klimamessig kan ikke resultatet direkte overføres til Norge, men en er klar over ulikheten i forbruk kan det være en god illustrasjon på konsekvensen av det fornybare skiftet.



Figur 5: Last på strømmettet i California [28]

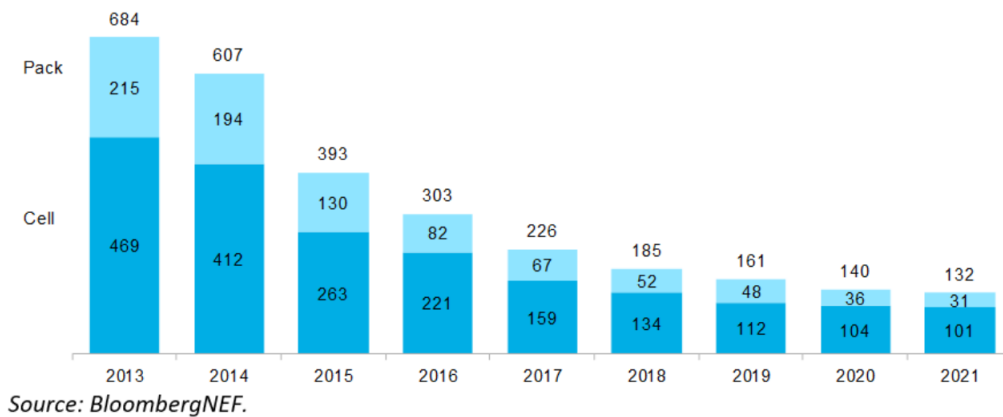
Grafen 5 for last beveger seg nærmest som en sinus-graf på grunn av strømproduksjon fra solcellepanel. Solinnstrålingen starter kl 9 am, er størst rundt kl 2 pm, men avtar deretter. Solcellenes strømproduksjon følger nøyaktig samme trend. Behovet for strøm fra nettet øker idet egenprodusert energi avtar og skaper store problemer for netteier, som må øke produksjonen med 9MW iløpet av 2 timer. En slik økning er vanskelig å få til gjennom vanlig produksjonsregulering, da både sol og vind er ikke-fleksible energikilder. Høy produksjon over kort tid krever mye ledig nettkapasitet, og er en dårlig ressursutnyttelse som øker strømprisen for alle forbrukere. Løsningen er å bruke batterikapasitet til å glatte ut stigningene, slik at produksjonen klarer holde følge. Oppgaven ser på om det er mulig å redusere størrelsen på U-en i graf 5 ved at forbrukere selv lagrer energien de produserer og med det glatter ut forbrukslasten.

Et godt eksempel er data fra det som var verdens største batterianlegg, *Horndale Power Reserve*, som har en utladningskapasitet på 100MW og en lagringskapasitet på 129MWh [4]. Anlegget er lokalisert sør i Australia og er koblet til 275kV nettverket gjennom samme punkt som Hornsdale vindpark på 30MW. Evalueringsrapporten utredet av Aurecon hevder at i løpet av 2019 reduserte batterilagringsanlegget kostanden på det nasjonale nettet med 119 millioner australske dollar. Av hvilket kommer 14 millioner dollar fra en 5 timers separasjon der batteriet var med på å stabilisere den sør australske delen som ble frakoblet resten av landet. Batteriet klarte å stabilisere nettet på $50 \pm 0.15Hz$ etter en topp på 50.85Hz etter separeringen iløpet av en periode på 16 minutter. Hornsdale prosjektet beviser at det er behov for batteri koblet til nettet både for å redusere kostnader i forbindelse med frekvens kontroll men også for å sikre at strømmettet er operativt selv om deler av nettet kobles fra hverandre i perioder.

2.2 Batteriteknologi

Batteriet er den viktigste komponenten i et **EBS**. Utviklingen skjer fort og det blir stadig utviklet nye batterityper. Det betyr at det er en risiko for at batteriteknologien blir utdatert før livsløpet til et batteri er ferdig. Valg av korrekt teknologi blir essensielt, og må basere seg på informasjon tilgjengelig i nåtid samt prognoser for fremtiden. Spesifikt i denne oppgaven er levetid, energitetthet, kontinuerlig kraft og kostnad/kWh viktige parametre.

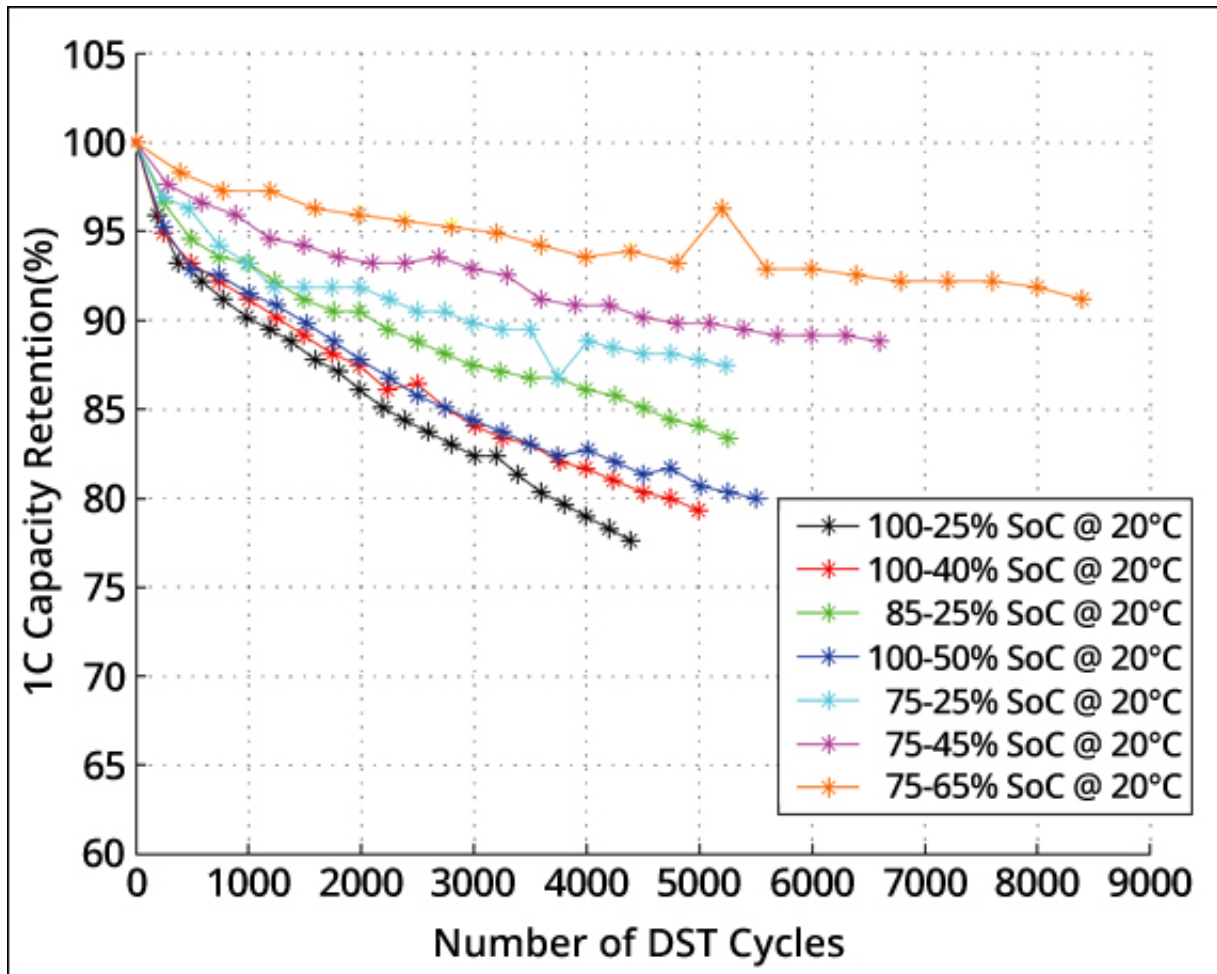
Litium-ion er den meste brukte batteritypen, fordi kostnaden er relativt lav på en snittpris på 132 dollar per kWh for 2021. Diagram 73 viser at kostnaden for litium-ion celler stadig reduseres, mye på grunn av press fra bilprodusenter [6]. Dog har råvaremangel i 2022 ført til at prisene på råstoff har begynt å stige, som kan redusere bruken av litium-ion batterier i fremtiden 1.2.



Figur 6: Diagram som viser utviklingen i pris for litium-ion celler

Litium-ion har idag en energi tetthet på $< 250\text{Wh/kg}$ og $< 650\text{Wh/l}$ [31]. Energitettheten når det kommer til batterilagring koblet til strømmettet er ikke avhengig av at energitettheten økes med mindre det er begrenset areal der energilagringssystemet skal installeres, dog fordelen med øke energitetthet lavere forbruk av råvarer fordi mer energi er lagret på mindre fysisk plass. Litium batterier har en anbefalt laderate på 1C men klarer opp mot 3C [52].

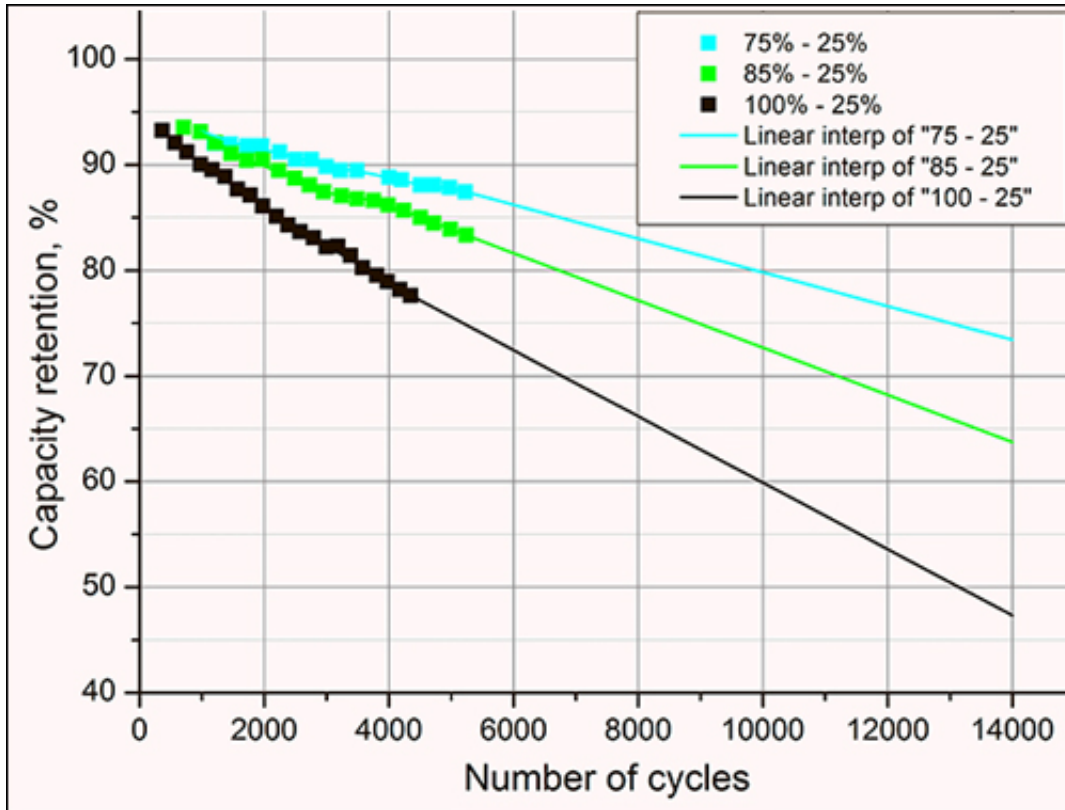
Litium-ion celler har lang levetid dersom **EBS** systemet tar hensyn til at fulle ladesykluser reduserer kapasiteten og livstiden på cellene.



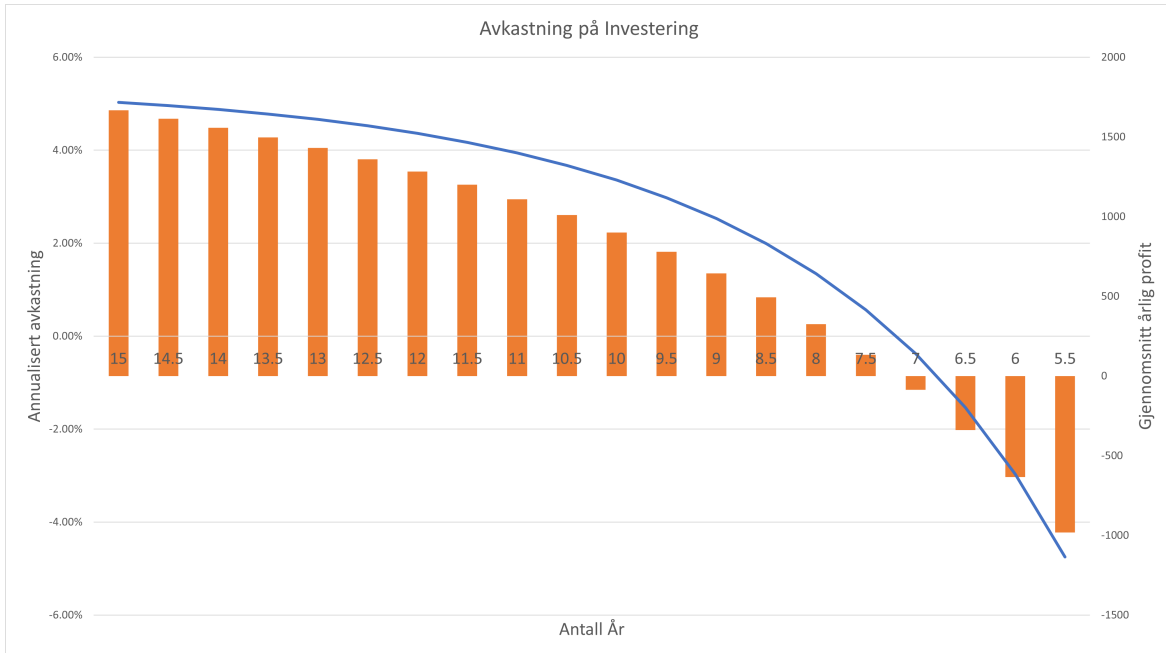
Figur 7: Graf som viser hvordan levetiden og kapasiteten reduseres avhengig av ulike lade og utladning områder, hentet fra [53]

EBS sikter mot å utnytte tilgjengelig batterikapasitet for å kunne forskyve forbruk mot pris mest mulig, men også lengst mulig levetid. Tar derfor utgangspunkt i grafene for 85%-25% og 75%-25% i figuren og setter levetiden til 5000 sykluser for å ha en margin som sikrer mot overestimert levetid, ettersom det har stor påvirkning på lønnsomheten. Dog er grafene tilnærmet lineære, så kapasitetsreduksjon til 80% av opprinnelig kapasitet vurderes som gunstig for bruksområdet, lagring av strøm i private hjem.

Graf 8 viser estimert degradering basert på tallene i graf 74 som prosentandel av total kapasitet med hensyn på antall sykluser for ulike ladeintervall. Lineær regresjon er benyttet til å estimere opp til 14000 sykluser. Ladeintervallet 75%-25% (lyseblå) fører til minst degradering. Ladeintervallet er en derfor en viktig parameter for å oppnå høyest mulig besparelse gjennom hele levetiden til **EBS**.



Figur 8: Forlenget levetid, hentet fra [53]



Figur 9: Figur som viser annualisert avkastning avhengig av levetid

Figuren 82 viser hvordan gjennomsnittlig profitt og annualisert avkastning endrer seg mot levetiden. Gjennomsnittsprifitt beskrives med likningen

$$Gjennomsnittsprifitt = \frac{\text{Årliginntekt} \cdot \text{levetid} - (\text{Kostnad}/kWh \cdot \text{TotalBatterikapasitet})}{\text{Levetid}} \quad (1)$$

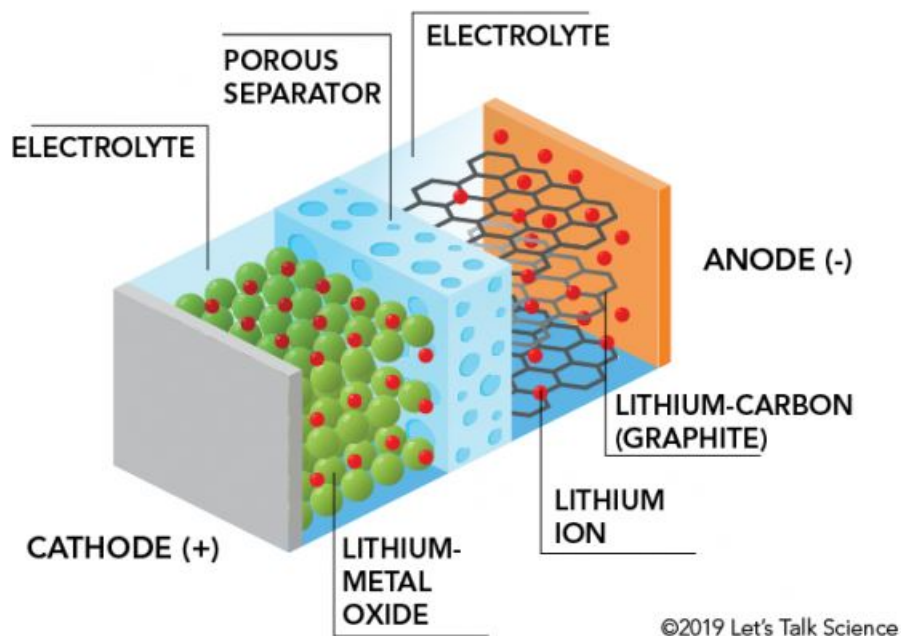
Den annualiserte avkastningen beskrives med likningen

$$Annualisertavkastning = \left(\frac{\text{Fortjeneste}}{\text{Kostnad}/kWh \cdot \text{TotalBatterikapasitet}} + 1 \right)^{\left(\frac{1}{\text{Levetid}} \right)} - 1 \quad (2)$$

Foretatte modelleringer viser en betydelig profittøkning ved forlenget levetid de første årene. Deretter flater kurven i 82 ut jo nærmere den kommer uendelig levetid. Ut i fra 82 vil en forlengelse av levetid fra 7.5 år til 8 år, resulterer i en økning i gjennomsnittsprifitt fra respektivt 133kr til 325kr. Det er en økning av fortjeneste på 244% gitt et halvt år forlenget levetid, som utgjør 1602 NOK over hele levetiden. Det faktum at inntekten ikke er konstant, men avhenger av strømprisen, egenprodusert energi og ytelsen til algoritmen, påvirker avkastningen på investeringen i stor grad, vist i resultatkapittelet.

Litium-ion cellene består av en negativ elektrode som er laget av litium metalloksid, den delen blir typisk lagt opp et lag med kobber som vist i figur 75. Den positive elektrodene som består av grafitt blir typisk lagt opp på aluminium. Elektrolytt materialet som er mest brukt er LiPF₆, som består av litium, fosfat og fluor. Den forhindrer elektronene å flytte seg mellom negativ og positiv katode. [57].

PARTS OF A LITHIUM-ION BATTERY



Figur 10: Detaljert grafisk fremvisning av en litium-ion celle

I denne oppgaven blir det tatt utgangspunkt i to ulike størrelser batteri. Batteristørrelsen har mye å si for **EBS** sin karakteristikker. Batterikapasitet bestemmer hvor lenge og hvor ofte batteriet kan supplere med strøm. Ytelse til **EBS** i form av kostnadsreduksjon og teknisk reguleringsevne er også i stor grad avhengig av batterikapasiteten. Det negative er at kapasitet er dyrt, så for at besparelse skal være høyest mulig må en finne skjæringspunktet mellom kostnad på **EBS** system og reduksjon i strømpris. Enkelt forklart punktet der en får mindre igjen for å øke kapasiteten på batteri. Det minste batteriet som blir modellert er 13.5kWh, det er samme størrelse som **Tesla Powerwall** som er et ferdig batterisystem tilgjengelig for privatmarkedet. Der finnes det data på effektivitet, ladeeffekt og kostnad [47].

Det største batteriet som blir modellert er 36kWh, det er et batteri produsert av RePack (nå Evyon 2.1), et norsk selskap som gjenbraker brukte elbilbatterier. Gjenbruk kan bli høyst aktuelt da litiumpriser har gått fra å være historisk lave 73 til å øke drastisk i pris fra høsten 2021/22. Andre edle batterimetaller som nikkell og kobolt varsles global mangel på [58],[55], beskrevet i kapitler 1.2 og 2.2. Samtidig med generelt økte råvarepriser nå våren 2022, blir bunnenlinjen at nye battericeller blir dyrere å produsere. Generelt har kostnaden av battericeller bare gått nedover de siste 10årene, mye på grunn av mer effektiv produksjon og derav lavere produksjonskostnader. Det betyr at råvarekostnaden blir en økende faktor i den totale kostnaden av battericellen, hvilket fører til at prisen på battericeller svinger mer i takt med råvaremarkedet. Figuren 74 viser at selv om kapasiteten er redusert såpass mye at rekkevidden for en elbil blir for kort som er et særlig problem på tidlige elbiler som hadde relativt små batterier helt ned i 28kWh. Kan det fortsatt brukes i et **EBS** der det i de fleste tilfeller ikke så stor praktisk betydning om effekt/kg eller per l øker noe mot at kostanden av pakken reduseres.[42]

Oppsummert gir *Tesla Powerwall* med 13.5kWh som bruker nye 2070 litium-ion celler og *RePack (Evyon)* på 36kWh som bruker brukte litium-ion celler fra Mercedes elbiler, et godt og representativt spekter innenfor de mulighetene som finnes idag på privatmarkedet. Det finnes andre alternative batteriteknologier [43], som *VRFB* [9] og *Solid-State* [40] nevnt introduksjonsmessig i 1.2 som kan brukes, men disse blir ikke modellert i denne oppgaven. Dog skal en tilleggsstudie utføres for å undersøke hvilken kapasitet på batteriet mellom 10 og 100kWh med et intervall på 10kWh som er optimal gitt begrensingene som blir brukt i modellen.

2.3 Inverter

For at batterier skal kunne brukes på strømnettet som er vekselstrøm på 230V 50HZ, kreves det en inverter. Både når batteriet skal lades opp og når det skal lades ut. Det er et tap av energi gjennom inverter og strømtapet avhenger av hvilken type inverter som brukes. Inverteren er derfor en viktig komponent å ta hensyn til i modellen. Effektiviteten til inverteren begrenser størrelsen på prissvingingene **EBS** kan utnytte.

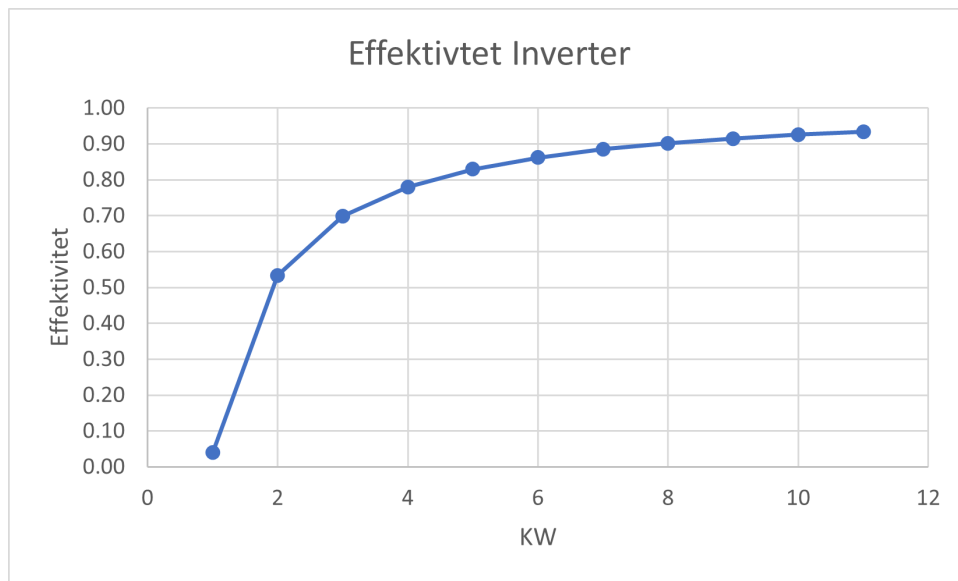
Effektgraden mellom ulike invertere varierer med hvilken teknologi som blir brukt, og kvaliteten på komponentene . Prisen varierer avhengig av kvalitet på inverteren. Høyest effektgrad og båndbredde for lavest mulig kostnad vil produsere størst avkastning.

Generelt kan tapet gjennom inverteren beskrives som [50]:

$$\eta(inv) = f(p = \frac{P_{out}}{P_{inveter}}) = \frac{p}{kp^2 + p + p_0} \quad (3)$$

$$k = 0.0345p_0 = 0.0072 \quad (4)$$

Grafisk fremvisning av likning 3



Figur 11: Graf som viser effektiviteten til inverter basert på ligning 3

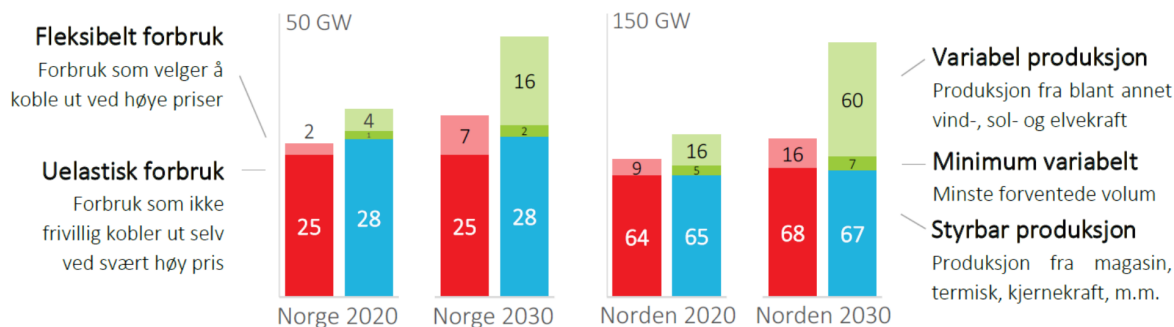
Invertere har stabil høy effektgrad i den øvre halvdel av arbeidsområdet. Inverteren på oppladningsiden av batteriet skal kunne holde seg i det øvre arbeidsområdet under oppladningen, og dimensjoneres ved at maksimal tillatt effekt gjennom inverteren settes lik tillatt effekt inn til batteriet. Inverteren på utgangssiden av batteriet er vanskeligere å dimensjonere for å begrense energitap, da en boligs strømforbruk avhenger blant annet av energikarakter, vær og vind og forbruksvaner.

Det viktigste er å ha inverter som klarer ta under forbrukstopper men som ikke er overdimensjonert, fordi det fører til lav effektgrad i perioder med lavt forbruk. Systemet settes opp slik at en kan ta strøm direkte fra nettet og ikke kjøre det gjennom batterisystemet i tilfeller der det er mest lønnsomt.

Tesla Powerwall har en innebygd inverter og den blir derfor brukt i modelleringen av 13.5kWh pakken. Den innebygde inverteren har en rundtur effektivitet på 90% som betyr et gjennomsnitt tap på 5%, både for oppladning og utladning. *Tesla Powerwall* har en maks effekt på 9.6KW og systemet kan kobles direkte til solceller. *RePack (Evyon)* sitt 36kWh batteri har ikke inverter men de kommer med anbefalinger på ulike inverter typer som batteriet er kompatibel med. Basert på anbefalingene brukes *Growatt SPA4000 10000TL3 BH* som har en maks effekt på 10kW og en rundtur effektivitet på 95%, systemet kan kobles direkte til solceller.

2.4 Strømmarkedet

Statnetts prognoser anslår at kraftbehovet i Norge kommer til å vokse mest av alle landene i Norden med en anslått vekst på 19TWh innen utgangen av 2026. Største bidragsyttere til veksten er elektrifisering av petroleumsindustrien, ny industri (bla. batterifabrikker) og datasentre [44]. Norges kraftoverskudd vil nærme seg null ved en forbruksvekst på 19TWh over en 5 års periode, og flere kraftverk må bygges ut.



Figur 12: Statnetts prognose på kraftsammensetningen [44]

Søylediagrammet 12 viser at variabel produksjon i Norge forventes å øke fra 4GW til 16GW. Styrbar produksjon er uendret da de fleste store ufredete elver allerede er bygget ut. Balansen mellom tilgjengelig, styrbar produksjon, og uelastisk forbruk har blitt svekket i de siste 5-6 årene. ”Fleksibelt industriforbruk, batterier og smart ladning blir sentralt for å kunne møte utfordringene”, ifølge *Statnett* [44]. Fornybar kraftproduksjon kan ikke styres, men avhenger av vær og vind. I områder og land med store lokale klimavariasjoner vil effektytelsen variere følgelig. Variasjonene i effekt bidrar til å øke strømprisenes volatilitet. Eksempelvis kan en sol- og vindfull sommerdag sende prisen ned i det negative, mot at prisen femdobles mot årlig gjennomsnitt en høytrykksperiode om vinteren.

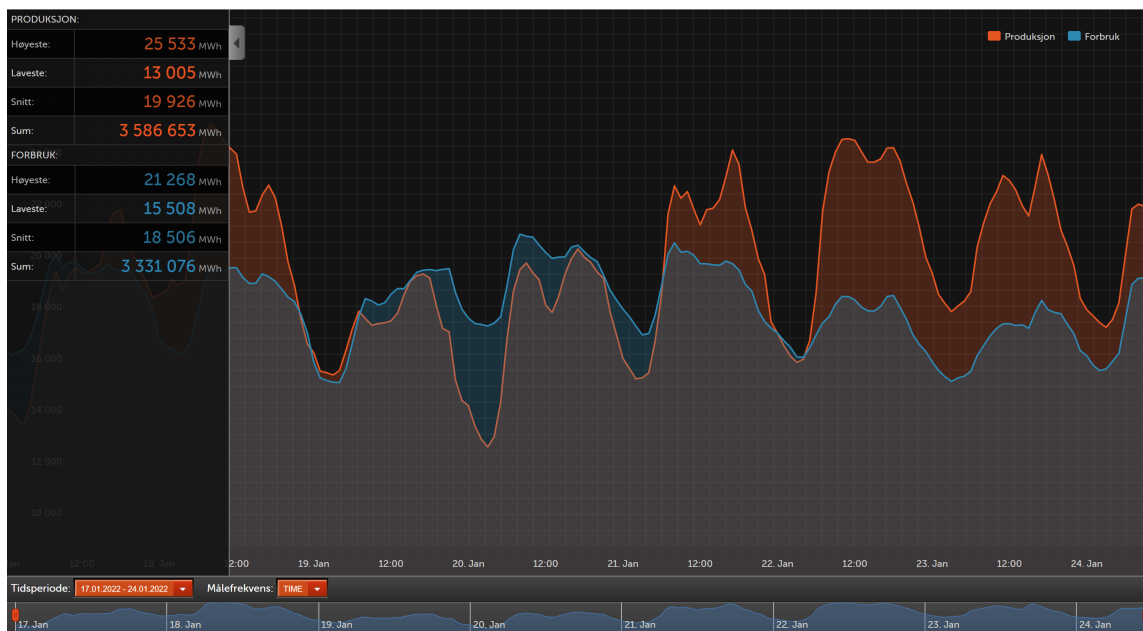
Strømmarkedet som vi kjenner det idag er et nytt system i historisk perspektiv. Det første landet som omstrukturerte og privatiserte strømmarkedet var Chile på slutten av 1970-tallet. Statens monopol klarte ikke å drifte strømmarkedet på en god måte. Hyppige strømbrudd og mangel på konkurranse førte til høyere produksjonskostnad. Løsningen ble en privatisering av strømmarkedet for å finansiere oppgraderinger og utvide kapasiteten til strømmettet.

En rekke land restrukturerte strømmarkedet på 1990-tallet. En av de mest vellykkete restruktureringene var Storbritannias privatisering av strømmarkedet i 1990, der en pool basert løsning ble utviklet og brukt som standard i andre markeder. I 1994 ble det nordiske strømmarkedet startet, dette ble det første multi-nasjonale spotpris markedet som idag styres av strømbørsen Nord Pool. I et pool-marked melder produsentene inn hvor mye de forventer å produsere den kommende dagen, og konsumentene kan legge inn hvor mye de ønsker å kjøpe. Nord Pool har to

strømmarkeder; *day-ahead* og *intraday*. *Day-ahead* markedet er hvor store volum handles. Markedet stenger kl 12.00 hver dag, då må alle parter ha lagt inn ordre. Det er dette markedet som fastsetter prisen til private forbrukere.

Intraday markedet er åpent hele døgnet alle dager i året. Her kan endringer som har kommet etter at *day-ahead* markedet stengtes legges inn i prisberegningene til strømmen. Dersom det er forventet sol men blir overskyet, kommer produksjonen fra solcelle-parker til å være mindre enn det som ble meldt inn til *day-ahead* markedet. Det kan komme mer vind enn forventet, som gjør at produksjonen fra vindmøllene overstiger forventet produksjon. Markedet er derfor viktig slik at deltakeren i markedet kan handle nærme leveransepunktet av strømmen ettersom ny informasjon blir tilgjengelig.

Vi ser i markeder som Tyskland hvor fornybar strømproduksjon er på 61,1% [7], at *intraday* markedet har vokst raskt og stabilt de siste 5 årene. Det er større usikkerhet knyttet til produksjonen og derav større svinginger i strømprisen. I Norge forventes ikke solkraft å stå for mye av total strømproduksjon. Sol-innstrålingen er størst om sommeren, men forbruket er lavere da det kreves mindre strøm til oppvarming. Vind er også satt på vent grunnet politisk motstand, og det forventes at vindproduksjonen i Norge skal gå fra 15TWh til 18TWh [44]. Norge har et forbruk med store daglige svingninger, noe som skaper prissvingninger selv om at produksjonene i stor grad kan reguleres 12, vist på figur 13. Norge er også koblet til et internasjonalt strømmarked som gjør at prisen blir påvirket av energisituasjonen i andre nærliggende land.



Figur 13: Graf som viser produksjon mot forbruk i Norge i perioden 17.januar kl 02.00 til 24.januar kl 11.00 hentet fra [45]

Ser på figur13 at forbruket og produksjonen svinger mellom en maksimal produksjon på 25533 MWh til laveste på 13005 MWh som viser de store svingingen intra-dag. I perioden 19.januar kl 07.00 til 21.januar kl 06.00 er produksjonen i Norge hele tiden lavere enn forbruket. Prisen på strøm var lavere i Europa i denne tidsperioden, som førte til at vi importerte strøm med en import-topp på 20.januar kl 04.00 der differansen mellom produksjon og forbruk er på 4693 MWh. Det betyr at den mengden strøm ble importert fra utlandet, vist i figur 14.

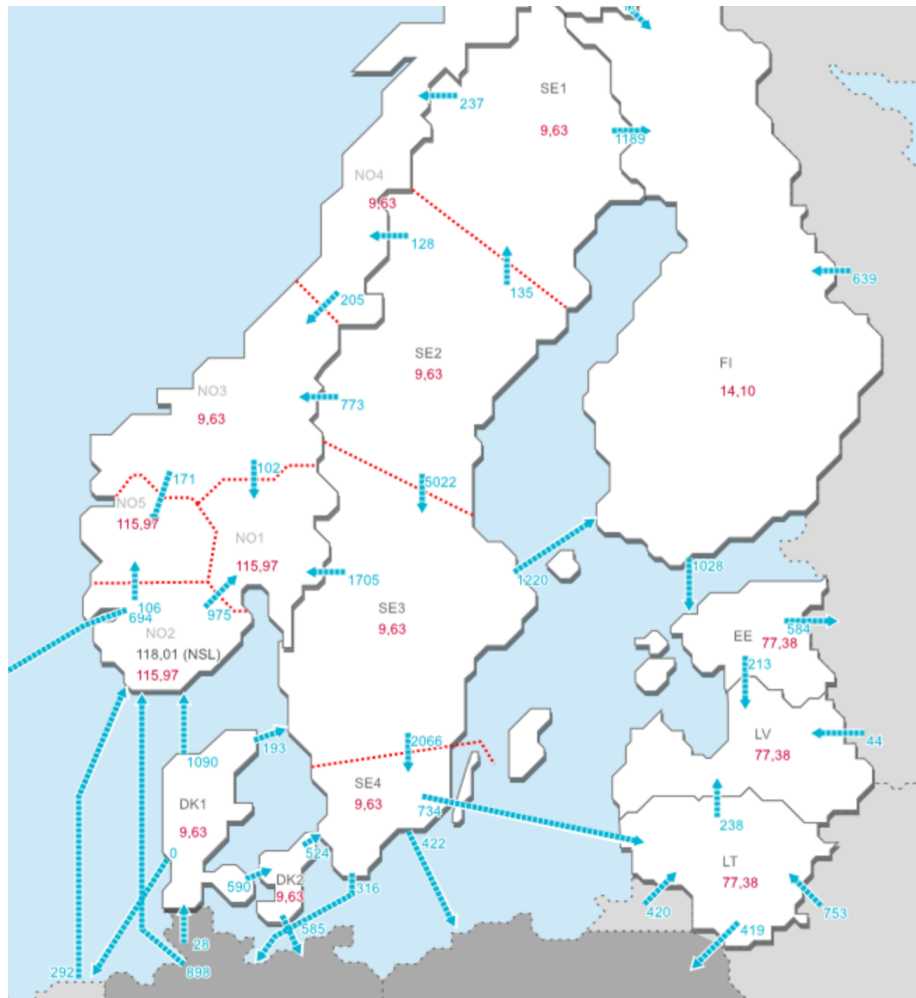


Figur 14: Graf som eksport og import til Norge i perioden 17.jan kl 02.00 til 24.jan kl 11.00 hentet fra [45]

Figuren 14 viser at i det punktet importen var høyest, ble 3082,1 MWh tilsvarende 65,6% av total import kl 04.00 20.januar importert fra Sverige. Grunnen til at så mye strøm kommer fra Sverige er fordi prisen per MWh er 9,63€ der mens den er 115,97€ i de 3 sørligste sonene av Norge NO1, NO2 og NO5 vist i figur 15. Det gir en prisforskjell på 1204% fra Sverige til Norge i denne timen. Legg merke til at prisen i nordlige deler av Norge er den samme som i Sverige. Dette er skyldes begrenset kapasitet på linjene som knytter sonene sammen. Kapasitet på linjene mellom Norge og Sverige er på totalt 3200 MW - 3600 MW fordelt på 9 kabler. 20.januar kl 04.00 ble det totalt importert 2843MW fra Sverige tilsvarende ca 83,6% av maks trekk.

Land	Kapasitet i MW	Antall kabler
Sverige	3200-3600MW	9
Danmark	1700MW	4
Finland	70-120MW	1
Russland	50MW	1
Nederland	700MW	1
Tyskland	1400MW	1
England	1400MW	1
Total	8500MW	18

Tabellen viser maksimal utvekslingskapasitet mot utlandet, data hentet fra [24].



Figur 15: Kart som viser strømpris og eksport i ulike soner koblet til Norge 20.jan kl 04.00 hentet fra [45]

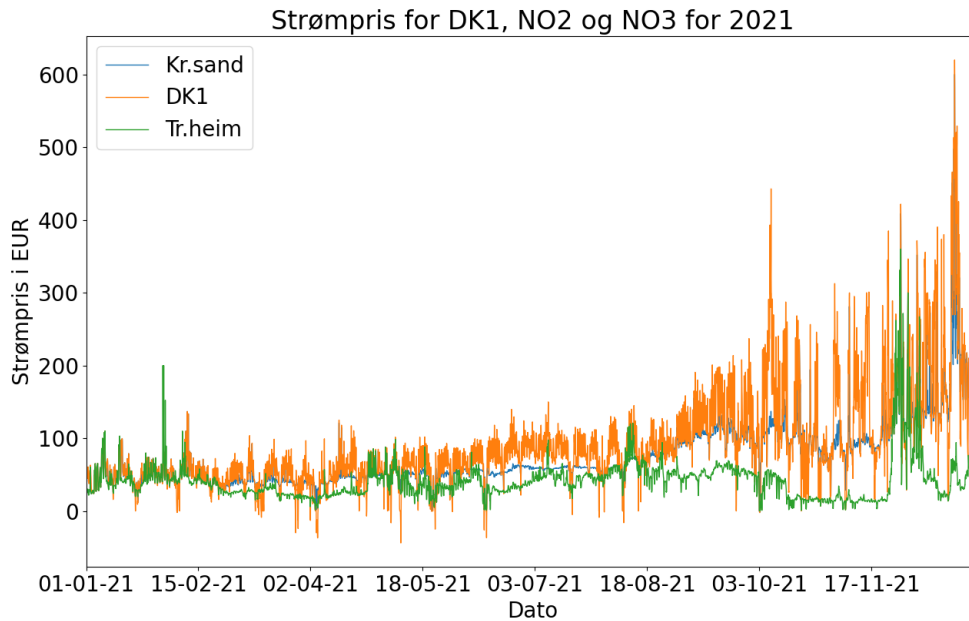
Endringene i strømmarkedet skaper et behov for balansering av strømmettet for å kunne integrere nye fornybare energikilder inn på dagens system. Fra 2020 og utover har energisituasjonen vært unormal i historisk perspektiv. Først var prisene unormalt lave, for så å bli unormalt høye og svinge mer. Dette skyldes delvis koronapandemien som reduserte energibehov over natten, og energikrisen som etterfulgte av at verden åpnet opp igjen og produksjon og energiforbruk økte. Energilagring vil i økende grad være en sentral faktor for å holde strømmettet stabilt og prisene forutsigbare. Da strømmettet sammenkobles mellom nasjoner og fossile energikilder fases ut, blir behovet for energilagring styrket.

3 Relevant data

I dette kapitlet dekkes og beskrives strømpris, forbruk, solceller, inverterkarakterstikk og batterikarakteristikk. Formålet er å undersøke hvilke betingelser som påvirker **EBS**

3.1 Strømpris

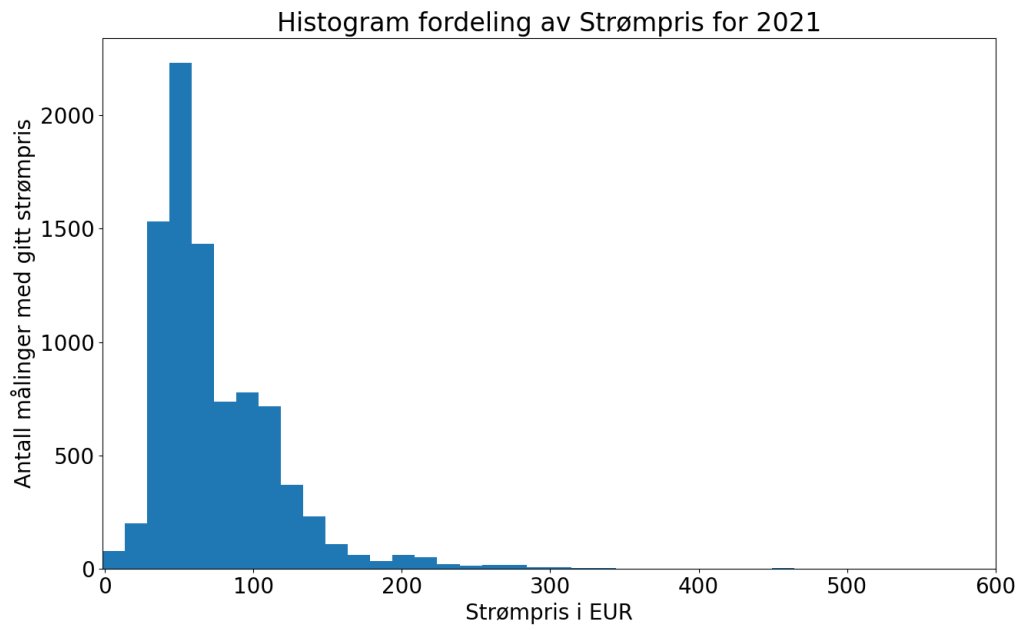
Strømpriser hentes for 3 ulike prissoner. Dataen er hentet fra *Nord Pool time for time* [37]. *Day-ahead* prisene brukes fordi det er prisen som private strømforbrukere betaler. Prisen fastsettes kl 13 hver dag for den kommende dagen. Det gir potensiale til å kjøre **EBS** i sanntid med litt modifikasjoner. Bakgrunnen for det er at ulike prissoner blir påvirket av ulike hendelser. NO3 er området rundt Trondheim vist i figur 15. NO3 er mindre påvirket av krafteksporten til Danmark, Tyskland, England og Nederland enn de 3 sørligste sonene i Norge. Dette er fordi det er relativt mye kraftproduksjon i området og begrenset kapasitet på strømmettet som går mellom sone NO3 og sone NO1 og NO5. NO3 er koblet på Sverige SE2. NO2 er den sørligste sonen i Norge, hvor strømprisen påvirkes i stor grad av import og eksport gjennom internasjonale undersjøiske kabler. Dette fører til større svinginger, særlig ettersom andelen ikke-fleksibel fornybar energi i Europa øker. Likevel har NO2 fortsatt et kraftoverskudd, som betyr at det er en netto eksporterende sone. DK1 er en netto-importerende sone som i 2021 importerte 4869GWh, og blir sterkt påvirket av vind i området samt strømprisen i Tyskland og Norge. Prisen i DK1 har gått under 0 ved flere anledninger, og drives ned av vindproduksjon når det blåser mye. DK1 har begrenset mulighet for å eksportere, da sonen går i overskudd samtidig som nærtliggende land med sin egen vindproduksjon.



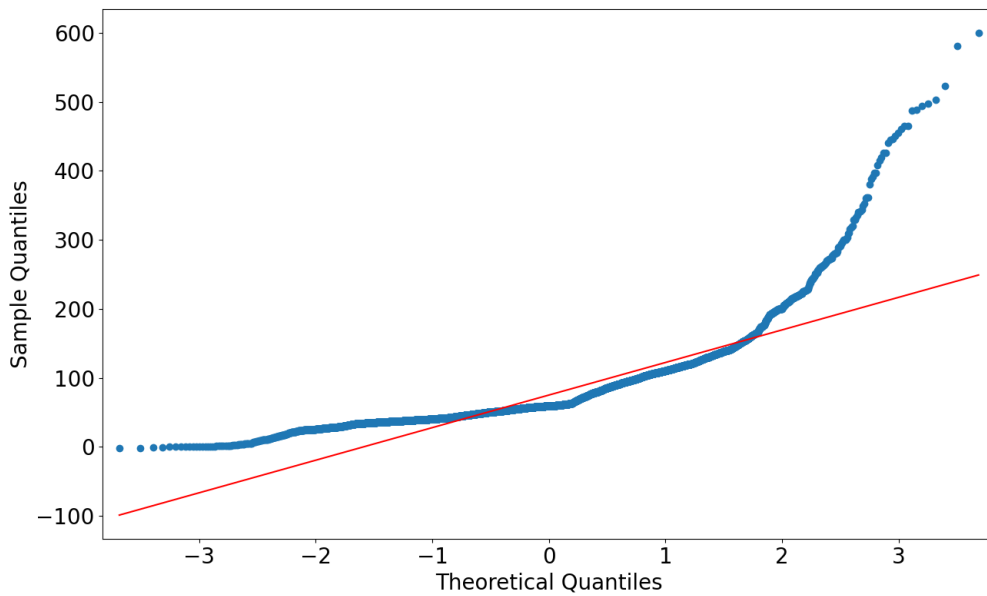
Figur 16: Oversikt over strømprisen gjennom 2021 for sonene DK1, NO2 og NO3

Strømprisen i sone NO3 (Trondheim) har lavest gjennomsnittspris gjennom 2021 på 41,1 EUR/MWh, 54% av gjennomsnittet for NO2(Kristiansand). Svingningene er også lavest i NO3 målt i standardavvik på 27,2. Standardavvik er et godt mål på svingninger og forteller hvor langt fra gjennomsnittet hver enkelt verdi ligger. Ulempen er at normalfordeling fra gjennomsnittet antas, som betyr at ekstremalverdier verdsettes høyere enn verdier som ligger nærme gjennomsnittet. Strømprisen er ikke normalfordelt, vist i histogram 17 og kvantilplot 18.

Standardavvik kan fortsatt være et godt mål på svingninger, selv om dataen ikke er normalfordelt. Kvadratroten av variansen er et mål på hvor mye hver enkelt måling avviker fra gjennomsnittet.



Figur 17: Histogram som viser fordelingen av strømprisverdier NO2 2021



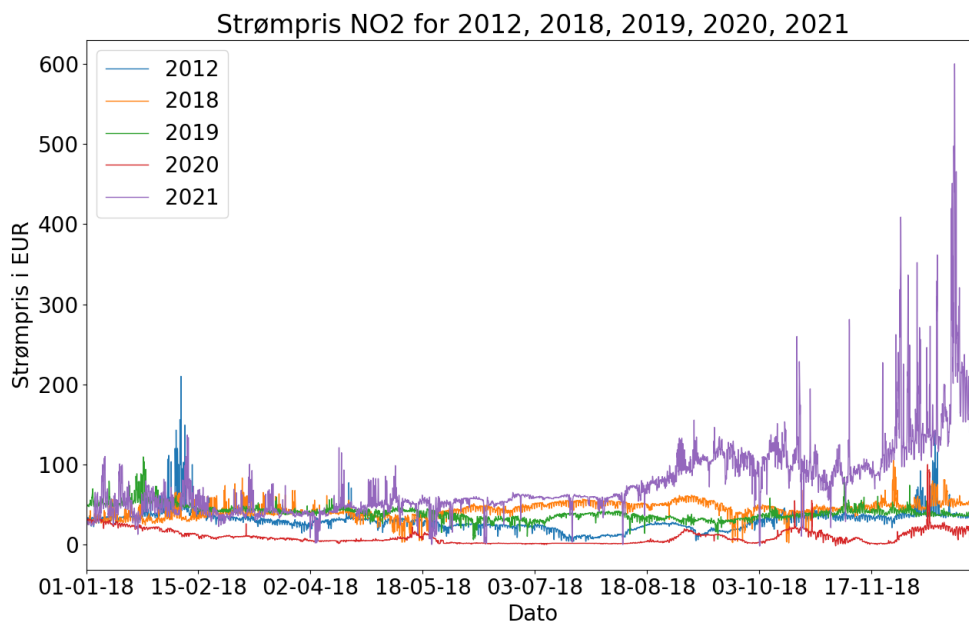
Figur 18: Q-Q plot for NO2 2021

Verdiene i den lave delen rundt 75 EUR/MWh er tilnærmet normalfordelt. 75,1 EUR/MWh er gjennomsnittet og på q-q plottet ser en at dataen som ligger ± 1 kvantil fra gjennomsnittet er tilnærmet normalfordelt. Når verdiene svinger mer fra gjennomsnittet forsvinner normalfordeling.

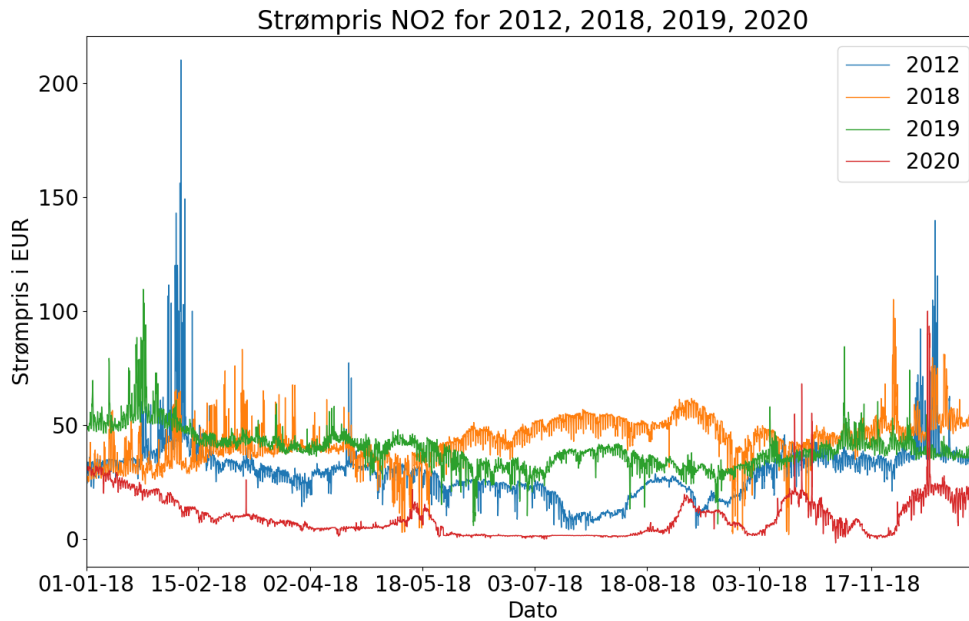
Det er en tydelig trend på at standardavviket øker jo lenger sør en kommer. For et **EBS** er større svinginger og høyere snittpris essensielt for hvor mye strømprisen kan reduseres med. Dersom strømpris var konstant, taper en uansett penger. **EBS** må utnytte prissvingningene og produsere en avkastning større enn kostnaden for tapet av energi gjennom systemet, og nok til å dekke initial investering i **EBS** i løpet av batteripakkens levetid. En uendelig batterikapasitet med **EBS** installert ville til slutt ført til en flat strømpris, da *peak-shaving* er en nettstabiliserende funksjon til **EBS**.

Sone	Kristiansand	Trondheim	DK1
Gjennomsnittspris EUR	75,1	41,1	88,1
Standardavvik EUR	47,2	27,2	64,8

Strømpris for sone NO2(Kristiansand) har endret seg over tid som en tydelig ser på figur 20. 2021 er i historisk perspektiv et avviks-år. Tabell viser gjennomsnitt strømpris og standardavvik for NO2, i følge plot 21 er strømprisen i 2018 tilnærmet normalfordelt.



Figur 19: Historisk strømpris for NO2

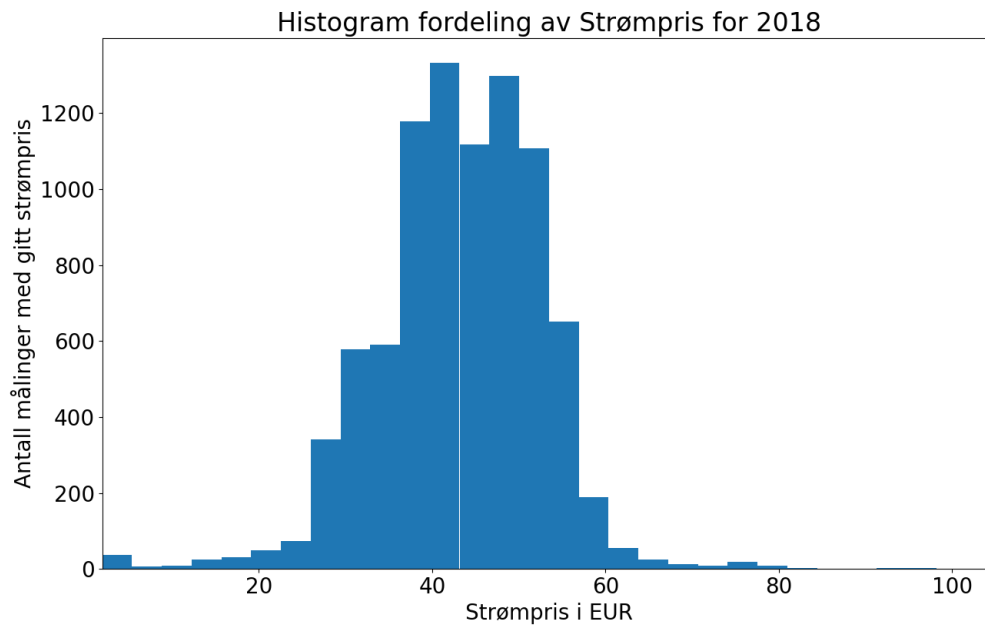


Figur 20: Historisk strømpris for NO2 uten 2021

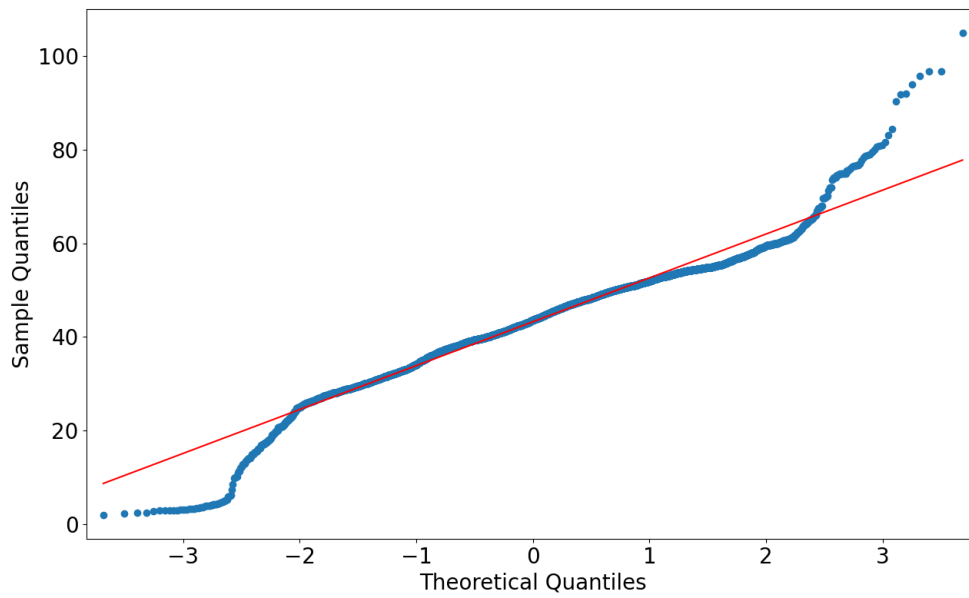
Det er typisk størst prissvingninger om vinteren, da temperaturen i Norge fører til at mye energi brukes til oppvarming. Det betyr at det er størst energibehov om vinteren, vist i figur 30. Det at volum er størst når svingningen er størst er positivt for **EBS**, fordi en fullt ut kan dra nytte av batterikapasiteten.

ÅR	2012	2018	2019	2020	2021
Gjennomsnittspris EUR	29,1	43,2	39,3	9,2	75,1
Standardavvik EUR	11,8	9,4	8,2	8,2	47,2

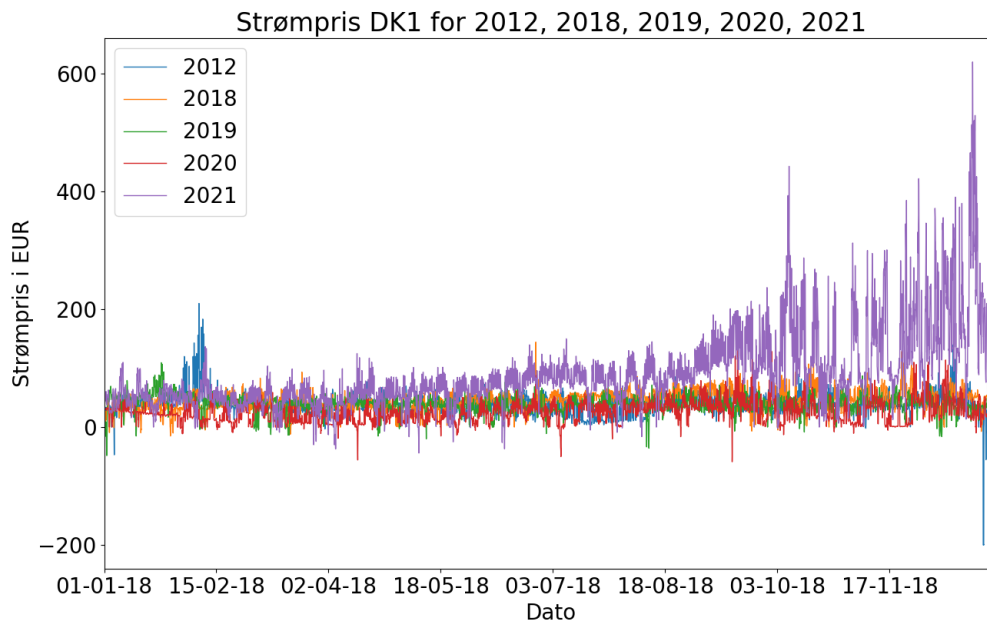
Strømprisen stiger ikke jevnt år for år i takt med inflasjonene. Den blir i økende grad påvirket av det globale energimarkedet særlig etter at kullkraftverk legges ned og erstattes med gass og fornybar energi. Svingningene målt ved standardavvik falt gradvis helt frem til 2020. I 2021 særlig mot slutten av året har verden opplevd en energikrise som drar opp både snittpris og svinginger. Det er umulig å si hvordan strømmarkedet blir i fremtiden, men prognosene til *Statnett* anslår en strømpris på 45 EUR de neste 5 årene og økende svingninger sett i historisk perspektiv [44].



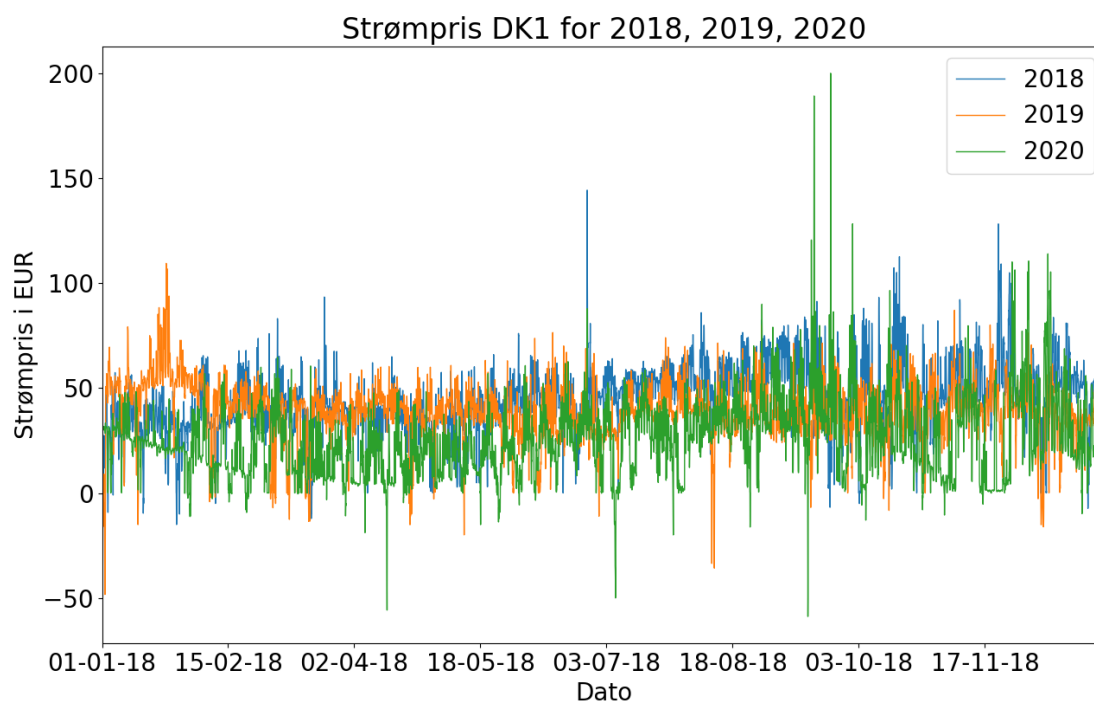
Figur 21: Histogram som viser fordelingen av strømprisverdier NO2 2018



Figur 22: Q-Q plot for NO2 2018



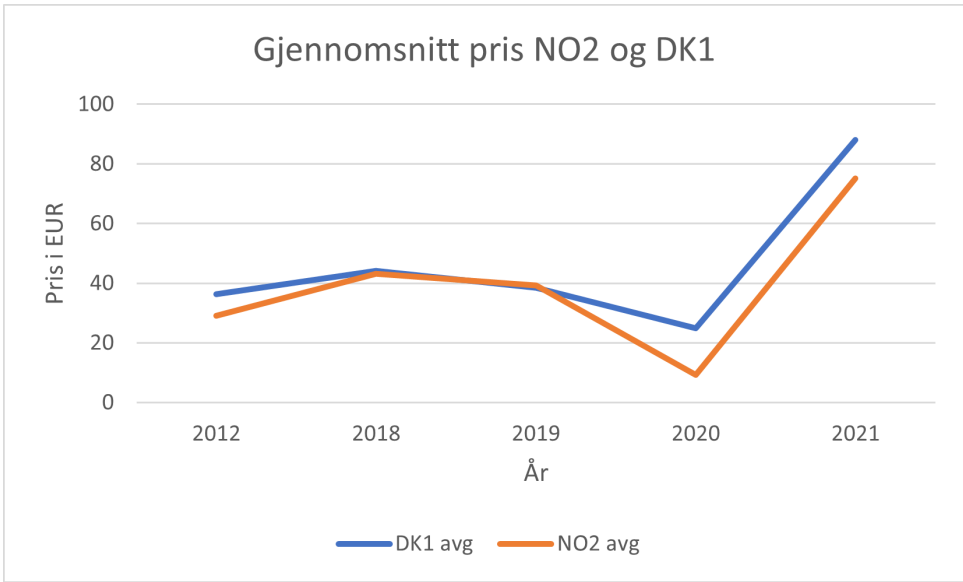
Figur 23: Historiske strømpriser for DK1



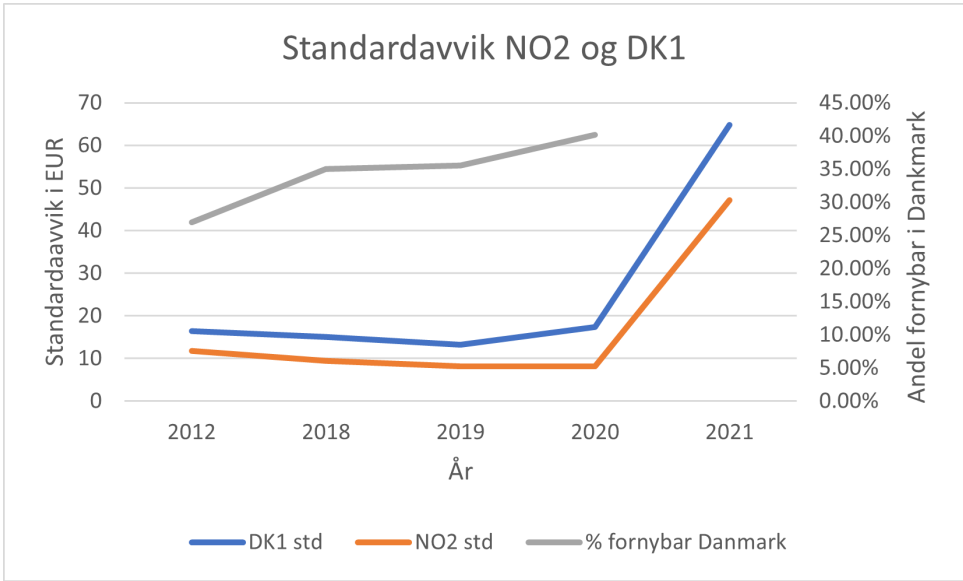
Figur 24: Historiske strømpriser for DK1 uten 2012 og 2021

ÅR	2012	2018	2019	2020	2021
Gjennomsnittspris EUR	36,4	44,1	38,5	24,9	88,1
Standardavvik EUR	16,4	15,1	13,2	17,4	64,8

Strømprisen for DK1 ligger stort sett over NO2 i hele måleperioden fra 2012-2021. I 2018 og 2019 er prisen tilnærmet lik, men ved unormalt lave priser i 2020 og unormalt høye priser i 2021 ligger prisen NO2 betydelig under DK1. **EBS** drar nytte av høyere strømpriser, da 1% fortjenste av 100EUR er mer en 1% av 20 EUR. Årene 2020 og 2021 regnes for å ikke være representative for fremtiden på lang sikt på grunn av det store avviket fra historisk gjennomsnitt . Data fra 2019 er relativt gammelt, særlig fordi hele energisektoren går gjennom en stor overgang fra fossilt til fornybart. Figur 26 viser ikke tydelig at standardavvik til strømprisen direkte kan kobles opp mot andel fornybar energi. Data om andel fornybar energi i Danmark er hentet fra [16]. Offisiell statistikk og data for 2021 var ikke tilgjengelig under forsøksperioden. I perioden fra 2012 og frem til 2020 synker standardavviket og derfor også svingningene. I 2021 stiger stiger standardavviket betydelig delvis drevet av høyere gjennomsnittspris.



Figur 25: Historiske strømpriser årlig snitt



Figur 26: Historiske standardavvik mot fornybar andel i DK

Merverdiavgift (MVA)

Merverdiavgift legges på strømprisen for å representere **EBS** installert i privat husholdning. Det betyr at alle strømprisene multipliseres med 1,25 for at det skal være 25% MVA. Strømprisen til forbruker består ikke kun av moms, men også nettleie, forbruksavgift, Enova-fondet, eventuelt påslag til kraftselskapet. Oppgaven tar utgangspunkt i spotpris som du for eksempel kan få hos *Tibber* [48], der er det en fast sats hver måned på 39kr, som må betales uavhengig om du har installert batteripakke eller ikke. Tas derfor ikke hensyn til. Det blir heller ikke tatt hensyn til de ekstraordinære strømstøtte-ordningene, fordi det ikke kan antas at de står i behold over enn 10års periode.

Nettleie

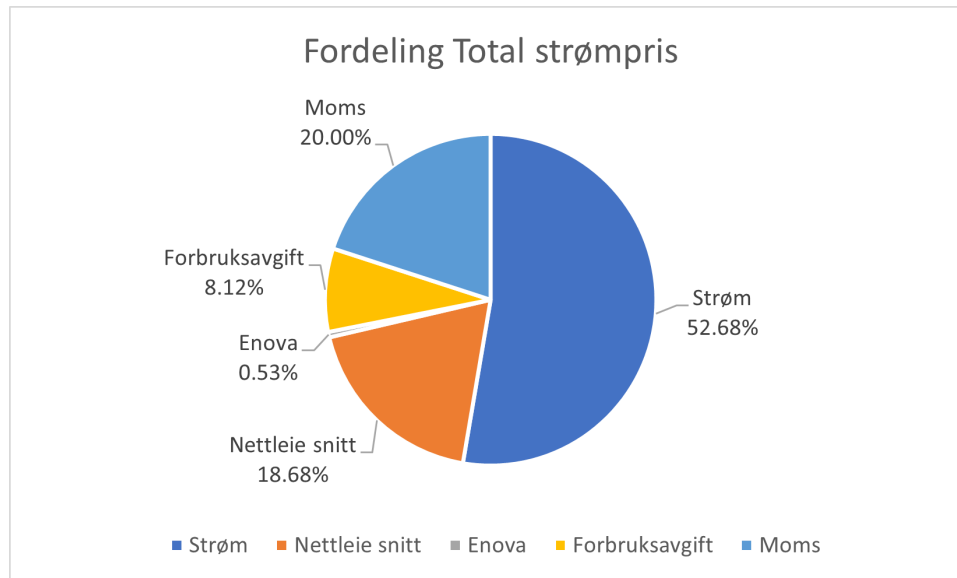
Nettleie er det stor usikkerhet knyttet til da den skal endres, men forsinkes av energikrisen. Opprinnelig skulle fastleddet avhenge av hvor mye kraft som ble trukket i løpet av en time. Da ville den timen du bruker mest i løpet av en måned brukes for å sette fastleddet. Det har dog i ettertid vært sterk motstand mot forslaget og ser nå ikke ut til å gå gjennom [35]. Det gjør det vanskelig å velge en nettleiemodell som skal brukes i oppgaven. En nettleiemodell som baserer seg på et maks trekk vil redusere besparelsen av et **EBS**, men gitt den store usikkerheten blir ikke den nye modellen brukt som utgangspunkt i oppgaven. Heller brukes den gamle fastledd-modellen som er konstant avhengig av forbruk og er dermed lik om **EBS** brukes eller ikke.

Det variable nettleie-leddet skal gå fra å være et konstant tillegg per kWh til å variere med hensyn på dag og natt. Det skal være dyrere å bruke strøm om dagen når det er mer belastning på nettet enn om kvelden. Dette tas hensyn til i modellen fordi det er meget sannsynlig at det innføres fordi det er store fordeler med å fordele forbruket utover slik at nettet ikke trenger bygges ut unødvendig. Satsene på det brukt i oppgaven er 36,5 øre inkludert moms om kvelden som er mellom 22 og 06 og 52,1 øre om dagen som er mellom 06 og 22, inkludert moms, tar utgangspunkt i foreslåtte priser [36]

Forbruksavgift og Enova-fond

Forbruksavgiften og Enova-fondet er statlige avgifter som må betales per brukte kWh. Forbruksavgiften er på 19,26 øre er kWh inklusiv MVA og Enova-avgiften er på 1,25 øre per kWh inklusiv moms. Disse er faste avgifter, men bidrar til å øke totalkostnaden av strømmen, noe som igjen bidrar negativt for muligheten til å selge tilbake strøm, da avgiften ikke returneres ved salg. Det at totalkostnaden øker betyr normalt økt besparelse, siden det er snakk om % reduksjon og 1% av 100 er mer enn 1% av 50. Det har likevel ingen praktisk betydning da avgiften ikke avhenger av nivå på strømprisen.

Salg tilbake på nettet tiltates, og det tas utgangspunkt i *tibber* sin løsning [49] der strøm kan selges tilbake på nettet for spot-pris hos *Nord Pool*. En får da ikke nettleie, fordi kjøper av strøm betaler denne. Det betyr at dersom en med batteriet klarer forskyve produksjonen utover dagen slik at en ikke trenger bruke strøm fra nettet, sparer en både ved at strøm kun koster investeringen i solcellene, og ved at en ikke får nettleie for strøm som blir kjøpt om kvelden.



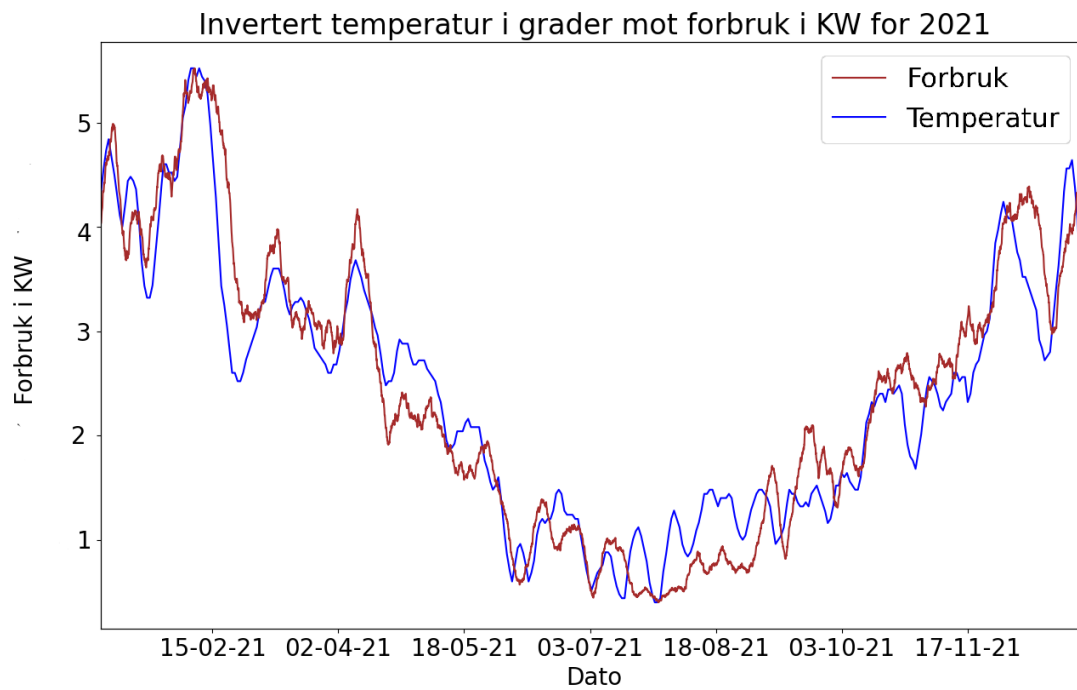
Figur 27: Fordeling av strømpris i % brukt i oppgave

Faktor	Pris i øre per kWh
Strøm	100
Nettleie snitt	35,45
Enova	1
Forbruksavgift	15,41
MOMS 25%	37,97
Total	189,83

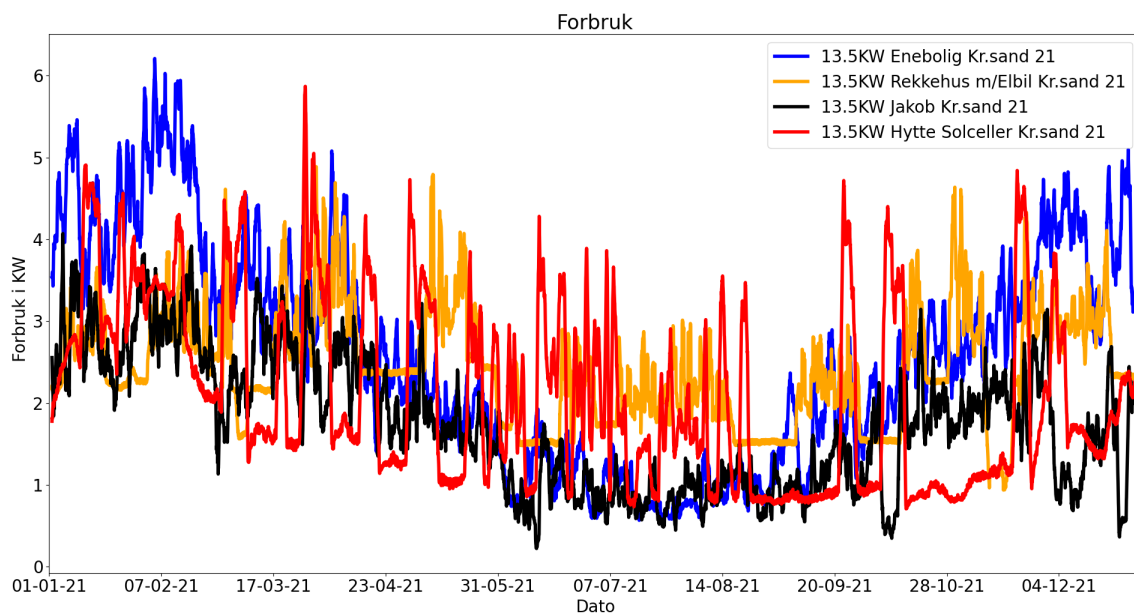
Strømprisen kan ikke styres, men har betydelig innvirkning på hvordan **EBS** opererer. Det kreves derfor gode kunnskaper om strømmarkedet i området **EBS** tenkes installert for å kunne vurdere om det er mulig å redusere strømprisen. En begrensende faktor for besparelsen til **EBS** er alle avgiftene som kommer i tillegg til strømprisen vist i figur 27. Dette betyr at det er nærmest umulig å tjene penger på å selge tilbake til nettet fordi en da kun får igjen spotprisen. Særlig når tap gjennom systemet også skal modelleres. Fordelen er betydelig større hvis egenproduksjon fra solceller kan brukes til salg grunnet ingen avgifter på egenprodusert strøm, i tillegg til å forsyne selvkonsum av strøm.

3.2 Forbruk

Forbruket varierer fra husstand til husstand avhengig av en rekke faktorer, som elbil og varmepumpe, eller hvor mange som oppholder seg i boligen og hvor ofte. Den viktigste faktoren for boliger i Norge er utetemperatur, ettersom den største bidragsyteren til strømforbruket sett bort ifra elbil-lading er oppvarming. Sammenhengen er illustrert i figur 28.

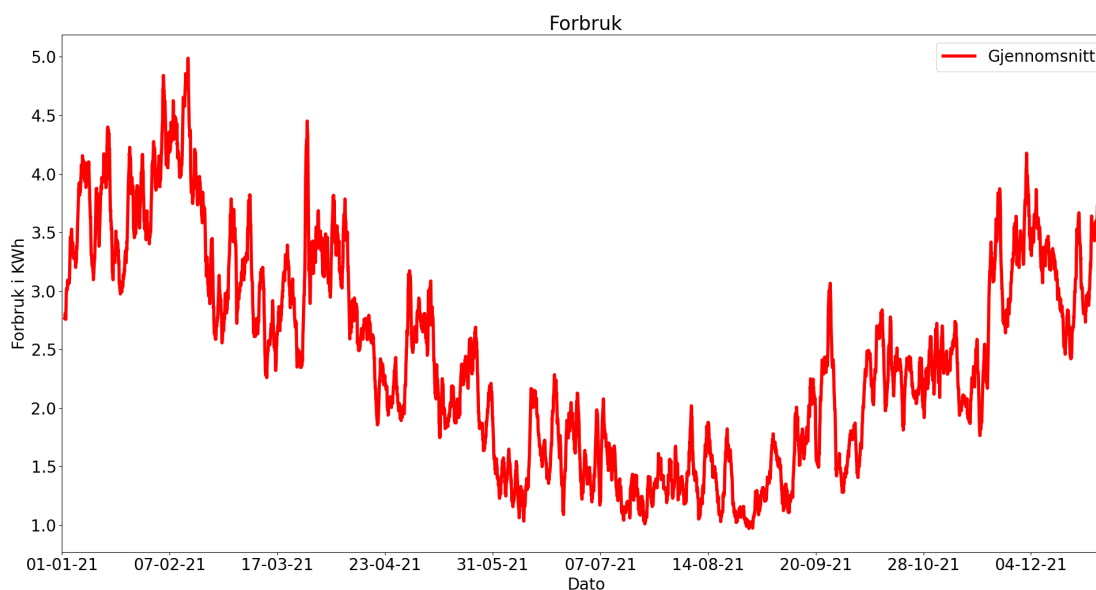


Figur 28: Temperatur i stavanger mot forbruk i enebolig 2021



Figur 29: Forbruk i KW for ulike hustander

Figuren 29 viser forbruk til 4 ulike boliger. Figuren 30 viser gjennomsnittet av de 4 boligene gjennom hele 2021. Noen av boligene bruker strøm relativt jevnt gjennom året mens andre følger temperaturen slavisk. Summen av de 4 forbrukene har en tydelig sammenheng med temperaturen. Forbruket fra grafen "13,5KW Rekkehus m/elbil" har en unormal syklus fordi eier arbeider skift i nordsjøen, og fører til at forbruket ligger lavt i perioden huset står tomt. Dog er det høyt forbruk særlig på grunn av elbil-ladning når eier er hjemme. I tillegg er det installert en krypto-miner som også blir brukt til å varme opp huset. Det fører til et relativt stabilt høyt strømtrekk gjennom hele året og blir derfor mindre påvirket av temperatursvinginger.



Figur 30: Forbruk gjennomsnitt for de 4 ulike hustander

Bolig	G.snitt kWh/døgn	Standardavvik kWh/døgn	Standardavvik KW/time
Enebolig	62,41	32,30	1,70
Rekkehus m/elbil	58,81	16,90	1,37
Leilighet	41,98	19,32	1,17
Hytte	50,64	26,29	1,29
Sammendrag	58,57	21,68	1,10

Det er ikke gunstig å bruke snittet av de fire strømførbukene fordi som tabellen 3.2 viser så jevnes svvingen ut når et snitt av ulike forbruk brukes. Dette er fordi de ulike hustandene bruker strøm på ulike tidspunkt og når det legges sammen så jevnes gjennomsnittet ut. Det er derfor ikke hensiktsmessig å bruke et gjennomsnitt for å si hvor mye strømprisen kan reduseres med. Det er det unike forbruket fra hver enkelt som skaper muligheten for å redusere strømprisen.

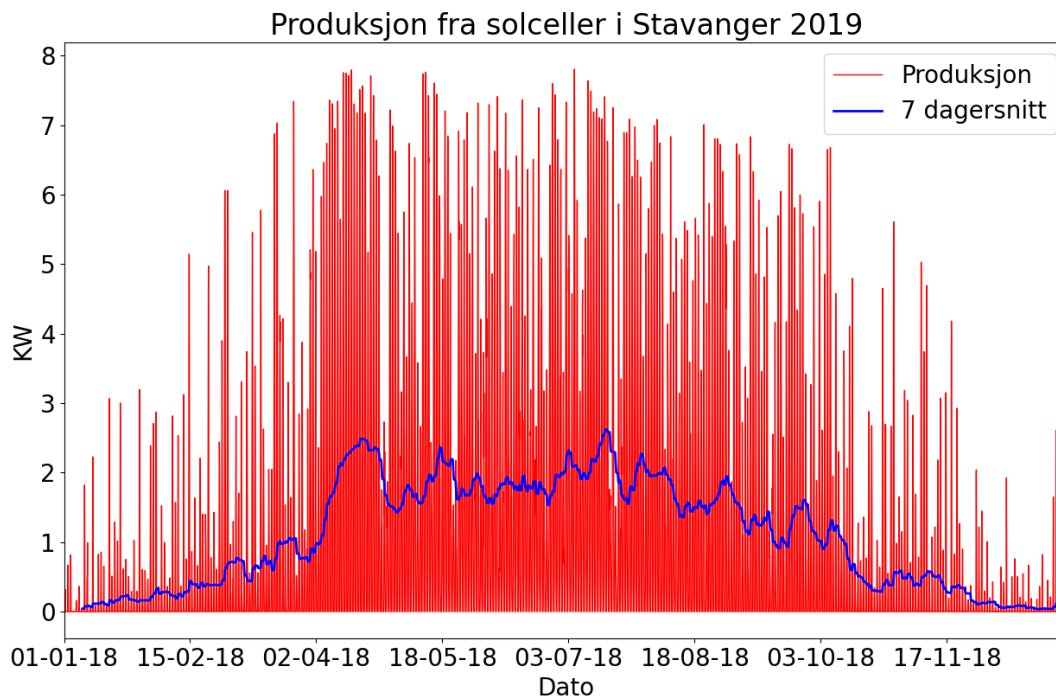
En ulempe med at alle boligene er lokalisert i Stavanger området betyr at forbruket ikke er direkte overførbart til andre prissoner. Dette fordi prisen i andre soner avhenger av lokal faktorer som temperatur.

Det vil føre til at resultatet blir usikkert fordi det kan tenkes at forbruket i andre regioner var annerledes. Time for time bevegelser er det som skaper besparelsen. Faktorer som når solen går ned og lokal temperatur skaper en forskyving mellom gjennomsnittsforbruk for den regionen og

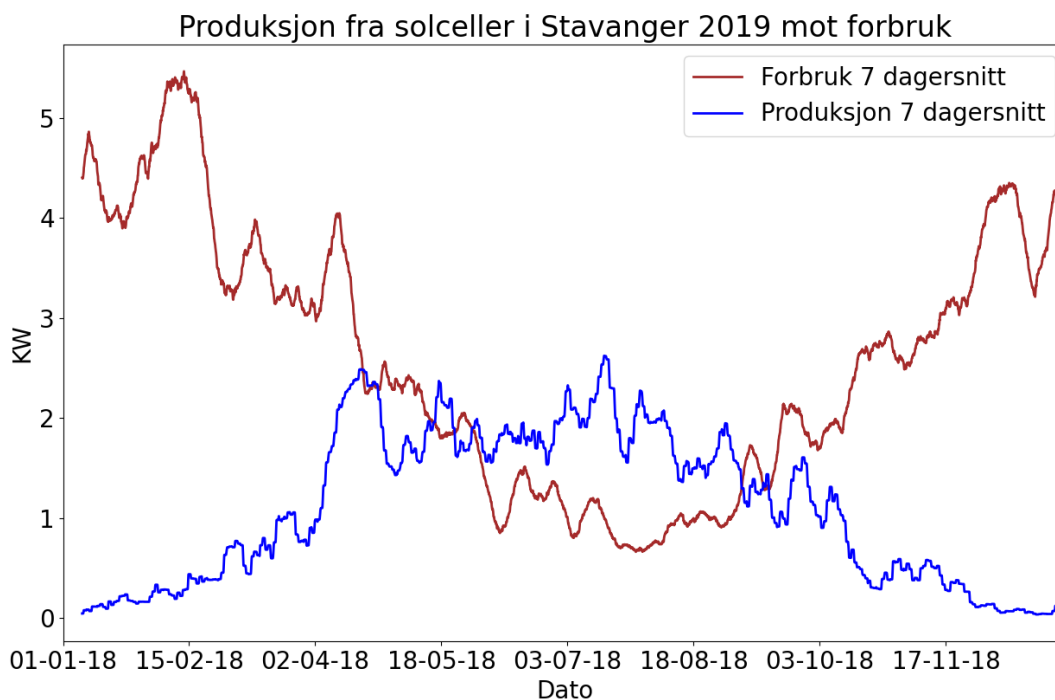
dataen som er blitt samlet i stavanger området.

3.3 Solceller

For modellene brukes data fra stavanger hentet på Renewables ninja [34]. Her settes KW_p til 10KW som er maks effekt solcellene kan levere. Det kan enkelt oppnåes ved å sette sammen 25, 400watt paneler. Tap for solcellen settes til 0.1, tilt settes til 35grader og retningen er 180 grader altså rett nord. Dataen fra 2019 kommer fra datasettet MERRA-2.



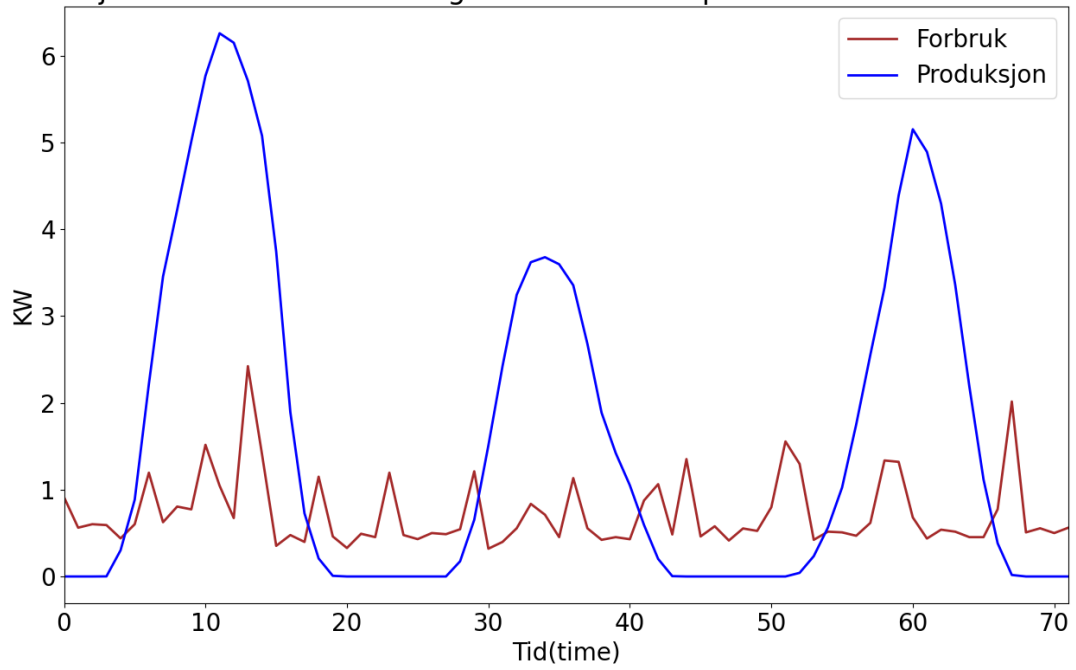
Figur 31: Produksjon fra solcellepanel i Stavanger



Figur 32: Produksjon fra solcellepanel i Stavanger mot forbruk

Det er mest produksjon i Norge om sommerhalvåret. Data fra 2019 vist i figur 31 viser at det er mest produksjon mellom Mai og Aug. Hvilket er naturlig siden solinnstrålingen er sterkest på denne tiden. Det betyr at det er størst produksjon av energi når forbruket er minst. Figur 32 viser at i en periode om sommeren fra juni til starten av september overstiger produksjonen forbruket som fører til en blir tvunget til å selge energi for lav pris til netteier. Her kan en bruke batterisystemet for å selge overskuddstrømmen når prisen er høyest. Problemet med solceller dersom mange i et område har det så blir strømtilbudet såpass høyt at prisen går ned. Dersom alle i et nabolag har solceller og alle disse har overskudd må enten strømmen fraktes til forbrukere som feks industri som bruker strøm jevnt gjennom hele året eller så må en strupe produksjonen og då ikke få så mye strøm igjen som mulig. Dog svingen produksjonen i takt med skydekket, en kan få mer produksjon når det er delvis skyet i de perioden det er sol på grunn av refleksjon fra skyene. Mens produksjonen faller de perioden skyene dekker solen. Denne oppførselen kan jevnes ut ved hjelp av batterier. En kan også forskyve trekket til kvelden då forbruket særlig om sommeren er høgest om kvelden når produksjon er lavest vist i figur 33.

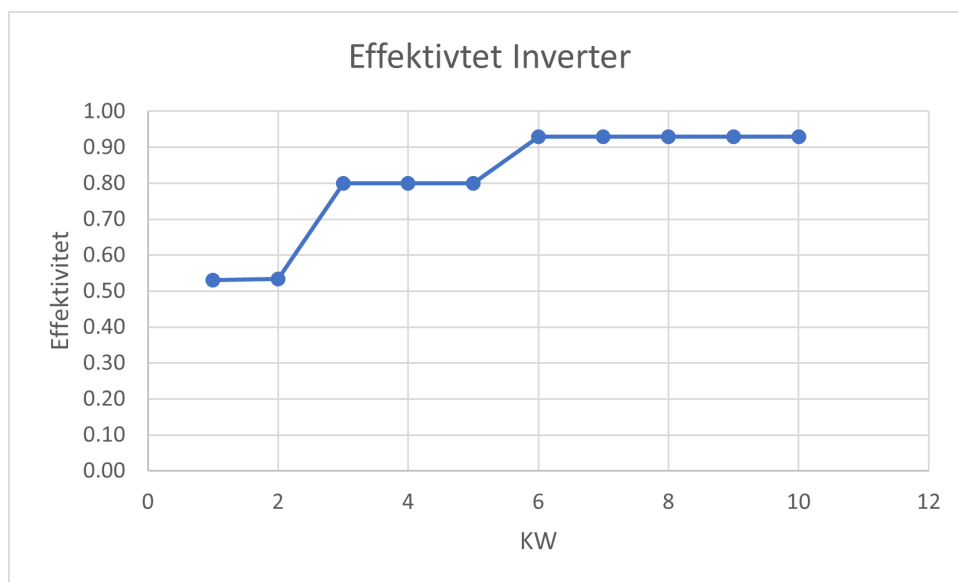
Produksjon fra solceller i Stavanger mot forbruk i perioden 29-07-19 til 31-07-19



Figur 33: Produksjon fra solcellepanel i Stavanger mot forbruk i perioden 29/31-07-19

3.4 Inverter karakteristikk

Tap gjennom inverter kan beskrives med likning 3. Karakteristikken forenkles med at effektiviteten beskrives med trinn i motsetning til en kontinuerlig graf. Dette fører til en potensiell feilkilde med at modellen holder oppladningen eller utladningen rett over der grensen mellom ulike effektiviteter går, men over tid skal det ikke ha stor betydning. Så lenge grensene settes på rimelige nivåer så er det uansett fordel å kjøre en oppladning nær maksgrensen for inverteren der den er mest effektiv. Det tas utgangspunkt i to ulike batterier og derav 2 ulike invertere. **Tesla Powerwall** har en innebygd inverter mens **RePack (Evyon)** kan kobles til en valgfri inverter. For oppgaven settes karakteristikkene likt Tesla, som oppgir et maksimum på 9.6KW, mens **RePack (Evyon)** kan ta 10KW. For oppgaven settes derfor maks effekt til 10KW. Figur 34 viser at grensene settes til 3KW og 6KW. 3 eller mindre .



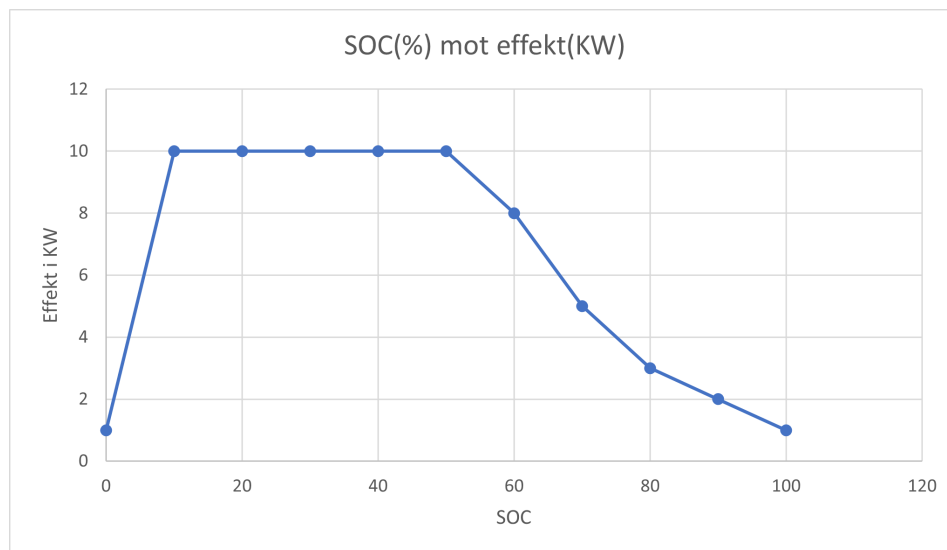
Figur 34: Inverter karakteristikk for oppladning brukt i modell

Tapet gjennom inverteren på utgangen settes konstant til 95% effektgrad. Forenklingen kan føre til feilkilder, men kompenseres mot ved å sette effektgrad under maks effektivitet for å best mulig representere det faktiske tapet.

Forbruket svinger betydelig mindre enn oppladningen og er med på styrke at inverteren settes konstant. Summen av inverter utgang og inverter inngang skal være 10% for å oppnå samme effektivitet som **Tesla Powerwall**. For oppgaven blir maks effektivitet 12% for at det anses som et mer realistisk verdi fordi tesla oppgir effektiviteten for en ideelle situasjon og også for å forsikre at ikke tapet settes for optimistisk siden det har stor betydning på hvor mye strømprisen kan reduseres.

3.5 Batteri karakteristikk

Et litium-ion batteri kan lades med høyere effekt når State of Charge (SOC) er lav enn når SOC er høy. Dette har blant annet med temperatur utvikling i batteripakken. Mer info her om hvorfor. Figuren 35 viser dataen som det blir tatt utgangspunkt i for modellen. Tidsteget på modellen er 1 time noe som betyr at den kjører samme som eller under maks tillatt effekt i en time frem til neste punkt.



Figur 35: SOC mot effekt for oppladningsinverteren

Kurven beskrives i modellen ved hjelp av likningen 5 som er en 3.grads regresjon av figur 35

$$\text{Effekt}_{max} = 6,702 \cdot 10^{-5} \cdot \text{SOC}^3 - 0,0131 \cdot \text{SOC}^2 + 0,6308 \cdot \text{SOC} + 2,308 \quad (5)$$

Degradering beskrives i modellen kun som at 0.0000345% av maksimal kapasitet forsvinner for hver syklus. Siden maksimum og minimum **SOC** settes til 90% og 10% så er det bygget inn en 10% buffer på oppsiden før en marker at kapasiteten har gått ned. For at nedgraderingen skal være representativ må modellen kjøre igjennom data for hele perioden det er tenkt at systemet skal være operativt. For at degradering skal tas hensyn til eventuelt må degradering kalkuleres inn med at maks besparelse reduseres med en bestemt verdi over tid avhengig av degradering. Formel som blir brukt er 6

$$\text{Degradering} = \text{kapasitetbatteri} - \left(\frac{0,0000325 \cdot \sum(\text{oppladning}[t] + \text{produksjon}[t] \cdot \text{sol})}{\text{kapasitetbatteri}} \cdot \text{kapasitetbatteri} \right) \quad (6)$$

4 Optimaliseringsproblem

For å kunne finne et optimum, må en likning konstrueres med tanke på ønsket objekt. Objektet vi ønsker å minimere i størst mulig grad er total kostnad for strøm. Den totale kostnaden for antall timer t , er gitt som summen av produktet til strømpris·oppladning + direkteforbruk - salg addert med summen av nettleien over samme tidsrom.

Likningen er som konsekvens av dette lineær. Det kan tyde på et optimaliseringsproblem av typen *Mixed-Integer Linear Programming*, da variablene er *continuous* og heltall, og likningen lineær. For at problemet skal kunne klassifiseres som av typen *MILP*, må også begrensningene være lineære.

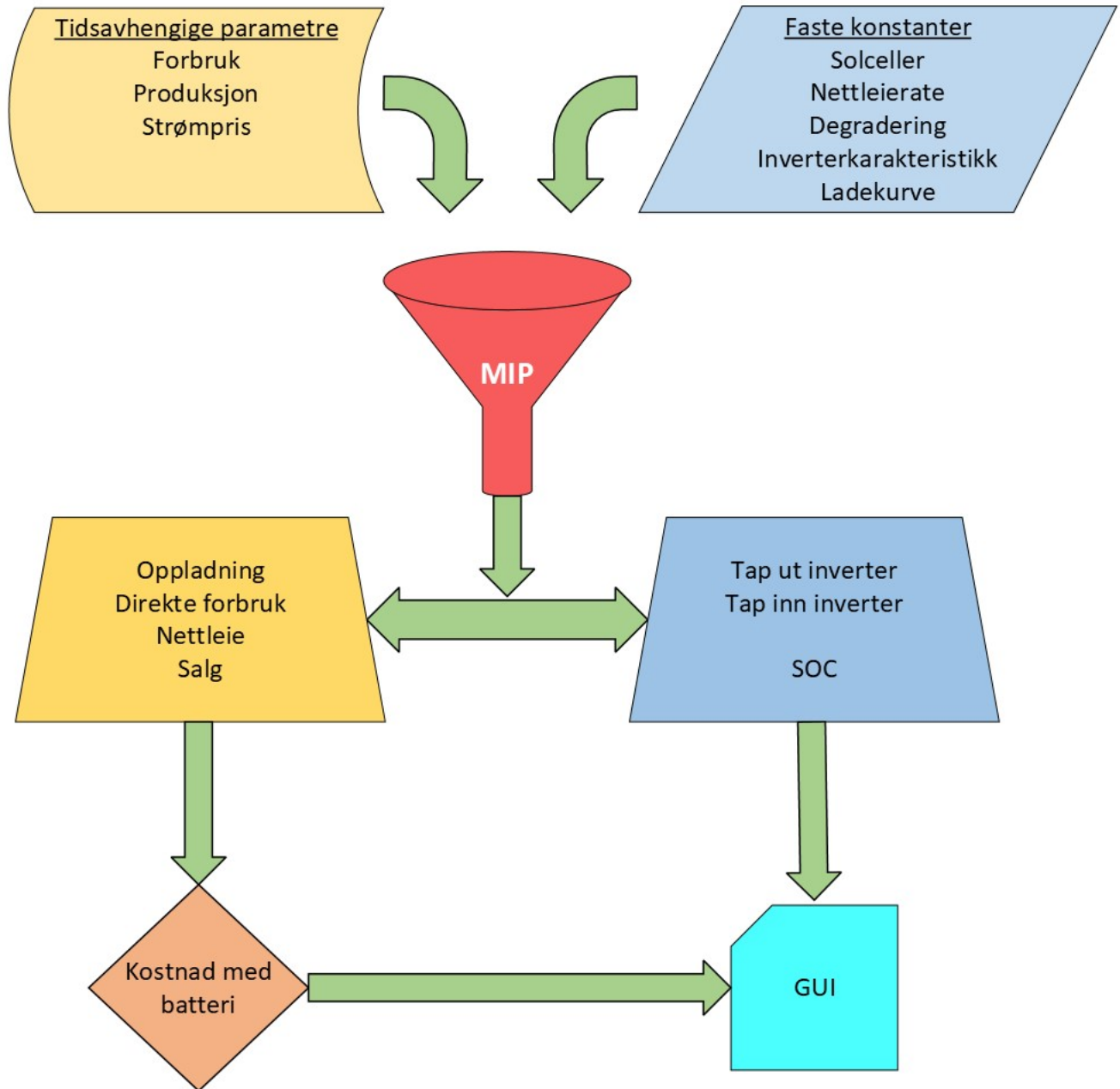
4.1 Valg av algoritme

Mixed Integer Programming (MIP) er samlebetegnelsen for en problemtype flere kommersielle løsere eksisterer for. Blant annet finnes *IBM* sin *CPLEX Solver*, og *Gurobi Optimizer*. Ikke-kommersielle tilbud finnes også, mest kjent er *COIN-OR Branch-and-Cut (CBC)* og nå nylig *SCIP*.

Valg av algoritme er viktig med tanke på fremtidig bruk, modularitet og kompatibilitet med eksisterende grensesnitt. Python-MIP er en ressurspakke som inneholder verktøy til å kunne modellere og optimere MIP-problem med programkode. Løseren python-MIP bruker som standard er CBC. CBC er open-source, og har vært i utvikling i tiår. Gjennom python-MIP pakken blir hele CBC-biblioteket oppkallbart. CBC velges dermed som løser-algoritme. Forbehold om at betalte, akademisk utviklede løsere kan finne bedre optimum foreligger, men tilgjengelighet og bevis av konsept kan likeså godt demonstreres ved hjelp av CBC.

4.2 Generell formulering

For å modellere et batterisystem må de fysiske begrensningene tas hensyn til. Målet er å minimere strømkostnadene for en forbruker. Strømkostnaden svinger gjennom dagen for hver time og opplading ønskes på batteriet når strømprisen er lavest. Variabelen oppladning målt i KW introduseres dermed til likningen. Dersom det ikke er store nok svingninger på strømprisen til at det skal lønne seg å bruke batteriet, må forbruket kunne dekket med strøm hentet direkte fra nettet. Dermed introduseres direkte forbruk målt i KW, som er den strømmen som ikke går gjennom batteriet, og dermed unngår energitap gjennom inverter og batteri. For å ta hensyn til kostnaden av nettleien som svinger med tid på døgnet, introduseres variabelen nettleie målt i EUR. Oppladning og direkteforbruk multipliseres med strømprisen, som er en fast parameter for hver time, for at alle enhetene skal være i EUR. For å kunne oppdatere tilstanden på batteriet introduseres bestemmelsevariabelen batteri målt i kWh. Denne oppdateres hver time etter hvor mye strøm som forsvinner som tap og hvor mye som blir utladet og oppladet. Det er ikke behov for en egen variabel for utladning da forbruket i boligen målt i kW som kalles forbruk i likningen kan trekkes fra direkte forbruk. Da må det forbruket som ikke blir dekket direkte fra strømmen komme fra batteriet. Det introduseres en variabel for inverter til oppladning kalt inverteroppladning i likningen. Denne er beskrevet i detalj i kapittel 3.4. Inverteren på utgangen kalt inverterforbruk i likningen brukes når strømmen fra batteriet skal gjøres om fra likestrøm til vekselstrøm som kan brukes i boligen. Solceller montert på boligen er beskrevet med en binær parameter kallet solceller og er 0 når boligen ikke har solceller og 1 når boligen har solceller.



Figur 36: Grafisk fremvisning av optimaliseringsproblem

Parametre

Tidsavhengige	Verdi	Enhet
Forbruk	Variereer for hver time	KW
produksjon	Variereer for hver time	KW
Strømpris	Variereer for hver time	EUR/kWh
Faste		
Solceller	0 1	Binær verdi
Nettleierater	Dag:0.0292 Kveld:0.0417	EUR
degradering	3.grad polynom ref.6	kWh
Inverterkarakterstikk	Likning ref.3	KW
Ladekurve	3.grad polynom ref.5	KW
Strømvavgifter	MVA:25% Avgifter:0.02051	EUR

Bestemmelsesvariabler

Bestemmelsesvariabler	Egneskaper	Enhet
batteri	Kontinuerlig, begrenset mellom 10% og 90% av full kapasitet	kWh
oppladning	Kontinuerlig, begrenset mellom 0 og 10	KW
direkteforbruk	Kontinuerlig, begrenset mellom 0 og 11	KW
nettleie	Kontinuerlig	EUR
tapinverteroppladning	Kontinuerlig, begrenset mellom 0 og 0.9	KW
tapinverforbruk	Kontinuerlig	KW
salg	Kontinuerlig, begrenset mellom 0 og 20	KW

Objektfunksjon *Totalkostnad* er objektfunksjonen og den økonomiske kostnaden vi ønsker å minimere. Parametre har stor forbokstav, variabler står i fet tekst.

$$\begin{aligned}
 \text{min. Totalkostnad[EUR]} &= \sum_{n=t}^T (\text{Strømpris[EUR/kWh]} \cdot (\text{oppladning[KW]} \\
 &+ \text{direkteforbruk[KW]} - \frac{\text{salg[KW]}}{1,25}) \\
 &+ \text{nettleie[EUR]})
 \end{aligned} \tag{7}$$

Begrensinger Dette er betingelsene objektfunksjonen skal minimeres innenfor. Hvilke begrensninger som tas med og kvaliteten av dem, har mest å si for kvaliteten til hele programmet.

(a) **Energibalanse**

Mikronettets energibalanse må overholdes til enhver tid. Det sikrer at positiv energi på venstresiden er samme altså dekker forbruket og tapet av energi på høyresiden. LP-problemet kan ikke ha negative verdier derfor er negativ energi representert med positive tall.

$$\begin{aligned} & \mathbf{oppladning}[\text{KW}] + \mathbf{direkteforbruk}[\text{KW}] + \text{Produksjon}[\text{KW}] \cdot \text{Solceller}[0|1] \\ & = \text{Forbruk}[\text{KW}] + \mathbf{tapinverteroppladning}[\text{KW}] + \mathbf{tapinverterforbruk}[\text{KW}] + \mathbf{salg}[\text{KW}] \end{aligned} \quad (8)$$

(b) **Oppdatere tilstand på batteri** Tilstanden til batteriet må oppdateres for hver time. Oppladning blir addert mens utladning blir trukket fra. Det er de samme bestemmelsevariablene og parametrene som i begrensing (a). Der det som står på venstre side i likning 8 blir addert og det som står på høyreside blir subtrahert.

$$\begin{aligned} \mathbf{batteri}[\text{kWh}] = & \mathbf{oppladning}[\text{KW}] - \text{Forbruk}[\text{KW}] + \text{Produksjon}[\text{KW}] \cdot \text{Solceller}[0|1] \\ & + \mathbf{direkteforbruk}[\text{KW}] - \mathbf{tapinverteroppladning}[\text{KW}] \\ & - \mathbf{tapinverterforbruk}[\text{KW}] - \mathbf{salg}[\text{KW}] \end{aligned} \quad (9)$$

(c) **Energitilførsel**

Batteriets oppladning tar hensyn til den spesifikke ladekurven til batteriet i bruk. Kurven er et tredjegrads polynom. Det virker med at oppladningen til en hver tid skal være under eller lik maksimal strømtrekk avhengig av tilstanden på batteriet (SOC). Se likning 5

$$\mathbf{oppladning}[\text{KW}] \leq \text{Effekt}_{max} \quad (10)$$

(d) **Kapasitet tap som følge av tid:**

Kapasitet til batteriet degraderes over tid avhengig av bruksmønster. Etter forenklinger brukes 6 som kun tar hensyn til hvor mye kapasitet som forsvinner avhengig av antall sykluser på batteriet mellom 10% og 90% av total batterikapasitet. På grunn av at maks oppladning er 90% av total kapasitet skal batteriet degraderes 10% før effekten blir gjeldende då bufferen er spist opp.

$$\mathbf{batteri}[\text{kWh}] \leq \text{Degradering} \quad (11)$$

(e) **Energitap som følge av inverter:**

Konstantene som ganges inn er effektiviteten fra figur 34 modellert trinnvis i tre trinn. Under 3KW er Basert på ligning 3. Det første trinnet har en effektivitet på 0,53, det andre 0,80 og det siste 0,93. 1 - effektivitet gir tallet vi ganger inn med oppladning[t].

- **Tap gjennom oppladning**

- for $\text{oppladning}[KW] \geq 6$

$$\text{tapinverteroppladning}[KW] \geq \text{oppladning}[KW] \cdot 0.07 \quad (12)$$

- for $3 \leq \text{oppladning}[KW] \leq 5.99$

$$\text{tapinverteroppladning}[KW] \geq \text{oppladning}[KW] \cdot 0.2 \quad (13)$$

- for $\text{oppladning}[KW] \leq 2.99$

$$\text{tapinverteroppladning}[KW] \geq \text{oppladning}[KW] \cdot 0.47 \quad (14)$$

- **Tap gjennom utladning**

$$\text{tapinverterforbruk}[KW] \geq (\text{Forbruk}[KW] - \text{direkteforbruk}[KW]) \cdot 0.05 \quad (15)$$

(f) **Direkte strømforbruk (som forbiplaner batteriet), skal ikke overstige totalt strømforbruk.**

$$\text{direkteforbruk}[KW] \leq \text{Forbruk}[KW] \quad (16)$$

(g) **Utrekning av nettleie kveld og dag:** Nettleien for det variable leddet variere mellom dag og natt. Det avhenger av strømtrekking i KW. Derfor multipliseres summen av oppladning og direkteforbruk med nettleieraten per KW.

- **Kveld**

$$\text{nettleie}[EUR] = (\text{oppladning}[KW] + \text{direkteforbruk}[KW]) \cdot 0.0365 \quad (17)$$

- **Dag**

$$\text{nettleie}[KW] = (\text{oppladning}[KW] + \text{direkteforbruk}[KW]) \cdot 0.0521 \quad (18)$$

4.3 Formel for bruk av Bestemmelsevariabler

$$\begin{aligned} \text{KostnadBatteri[EUR]} &= (\text{Strømpris[EUR]} + \text{avgifter[EUR]}) \cdot \text{oppladning[KW]} \\ &+ \text{nettleie[EUR]} \\ &+ (\text{Strømpris[EUR]} + \text{avgifter[EUR]}) \cdot \text{direkteforbruk[KW]} \\ &- \text{Strømpris[EUR]} \cdot \text{salg[KW]} \end{aligned} \tag{19}$$

If (forbruk[t] - produksjon[t] · sol) > 0:

$$\begin{aligned} \text{KostnadUtenBatteri} &= (\text{Strømpris} + \text{avgifter}) \cdot \text{Forbruk} + \text{nettleie} \\ &- (\text{Strømpris} + \text{avgifter}) \cdot (\text{Produksjon} \cdot \text{sol} \cdot 0.95) \end{aligned}$$

Else:

$$\begin{aligned} \text{KostnadUtenBatteri} &= (\text{Strømpris} + \text{avgifter}) \cdot \text{Forbruk} + \text{nettleie} \\ &- \text{Strømpris} \cdot (\text{Produksjon} \cdot \text{sol} \cdot 0.95) \end{aligned} \tag{20}$$

5 Resultat

Optimaliseringsproblemet er testet under ulike situasjoner. Som vist i figur 16 varierer strømprisen fra sone til sone, og betyr at boligens plassering har innvirkning på ytelsen til **EBS**. Forbruksmønsteret til den enkelte bolig har også stor betydning på ytelsen, samt produksjonen fra solcellene.

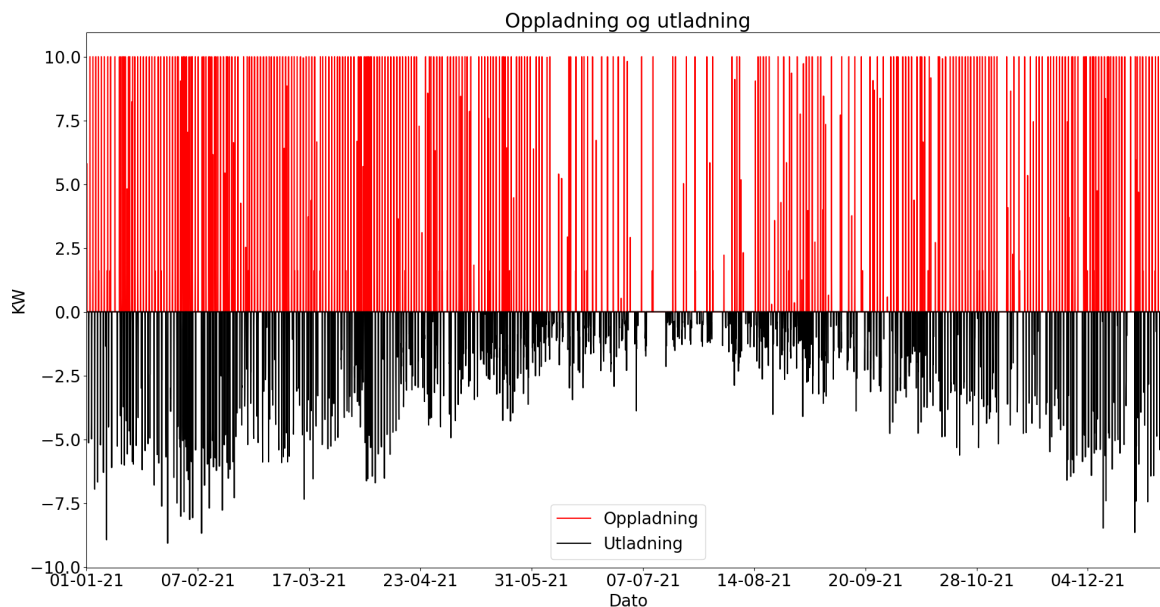
5.1 Ytelse med batteristørrelse på 13.5kWh for enebolig i pris-sone Kristiansand 2021



Figur 37: Oversikt over resultat for 13.5KW enebolig Kristiansand 21

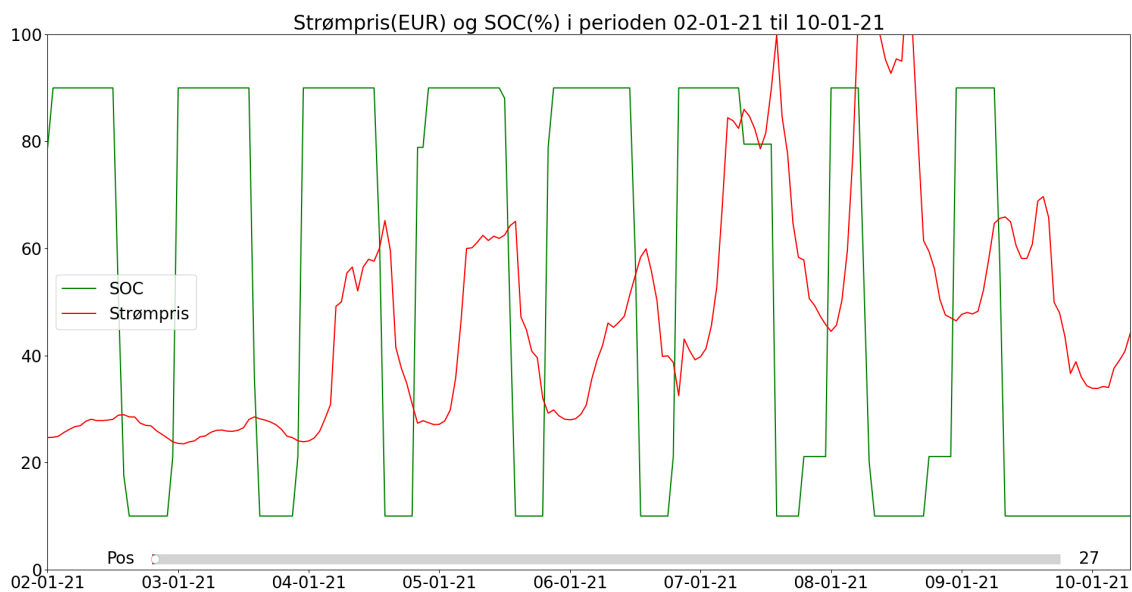
Over i figur 37 simuleres forbruket fra en enebolig i Stavanger-området for 2021, med strømprisen for Kristiansand (**NO2**) 2021. Ved hjelp av programvaren utviklet i denne oppgaven med tilhørende MIP-optimalisering, er det mulig å redusere strømkostnadene med 2,9% tilsvarende 111 EUR i løpet av hele 2021. 19 EUR av 111 EUR kommer fra besparelse fra nettleie, mens resterende kommer fra besparelse i innkjøpsprisen av strømmen. Stigningsraten på akkumulert besparelse er høyest mot slutten av året når strømprisen er høyest og svinger mest. Totalt forbruk for eneboligen i Stavanger var på 22.8 MWh for 2021. Av dette dekkes 19,4 MWh rett

fra strømmettet mens de resterende 3,3 MWh dekkes av strøm som har blitt lagret på batteriet. Batteriet har blitt ladet opp med en total effekt på 3794 kWh som gir en effektivitetsgrad på 88,1% for strøm som har gått gjennom **EBS**. Det var som forventet gitt begrensinger som er brukt for inverter 34.



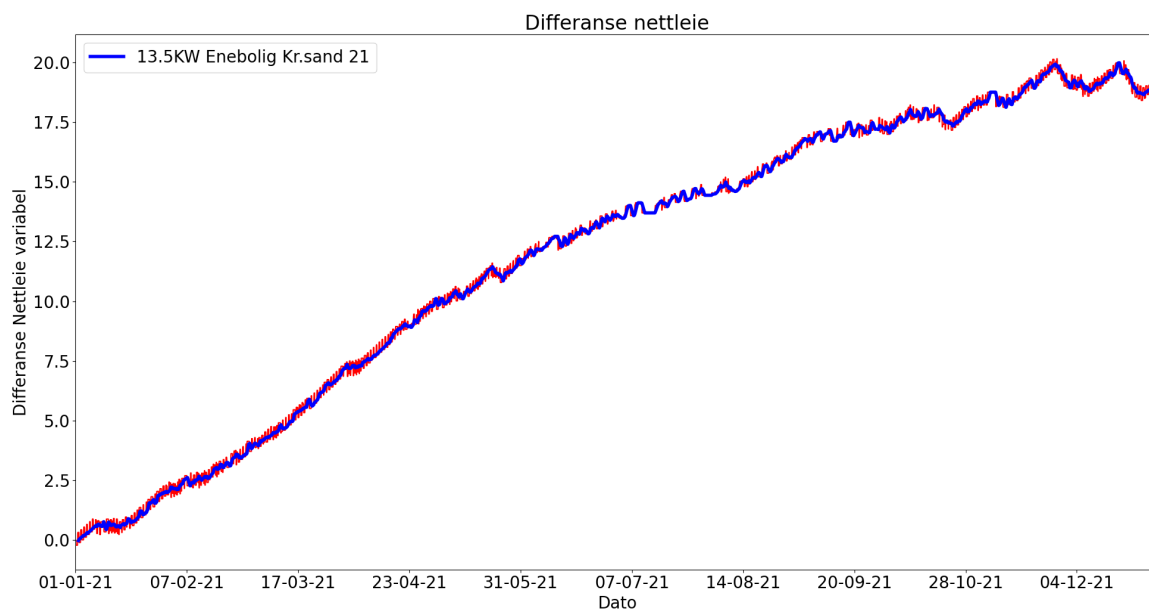
Figur 38: Oppladning og utladning for hver enkelt time gjennom 2021

Oppladningen er begrenset til maksimalt 10KW iløpet av en time. Aktiviteten på batteriet svinger med forbruket. Derfor er det minst aktivitet på batteriet når forbruket er lavest. I dette tilfellet er det om sommeren. Hustanden har ikke elbil, og bruker ikke andre elektriske produkter som trekker strøm med høy effekt. Derfor blir oppvarming største bidragsyter til forbruksmønster. I de fleste tilfeller trekkes full effekt til oppladning, hvilket er naturlig for å kunne lade mest mulig i den billigste timen.



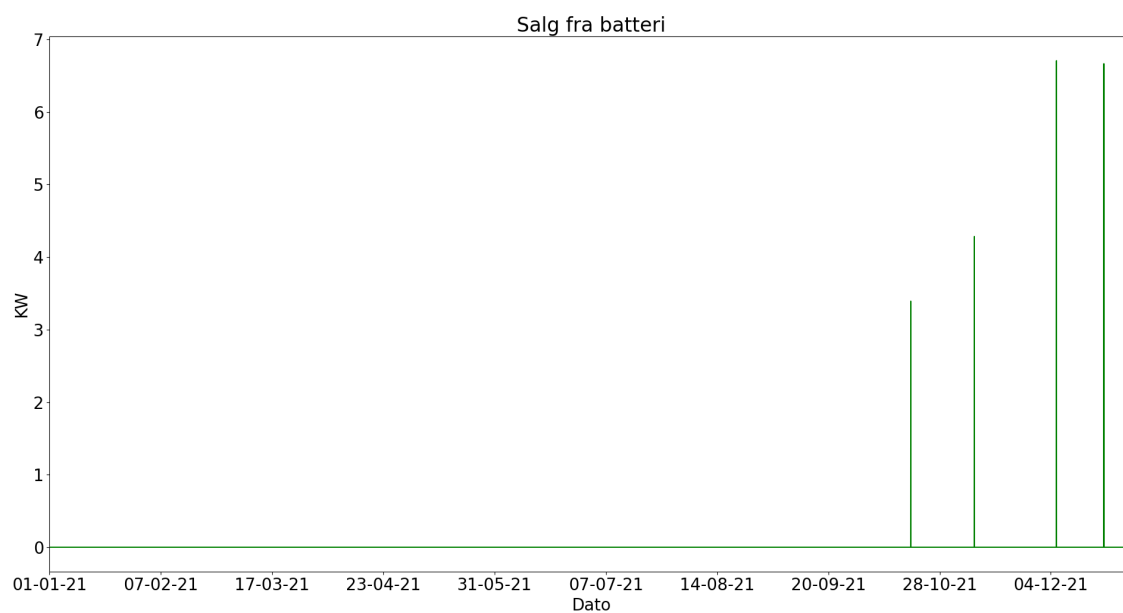
Figur 39: State of Charge mot strømprisen

Gitt begrensinger i optimaliseringsproblemet gjøres kun en oppladningsyklus når det er lønnsomt. Altså at fortjensten overstiger tap av energi gjennom **EBS**. Mye av tiden blir derfor batterisystemet stående uten å verken lade opp eller ut. Batteristørrelsen virker også begrensende, og med 13.5kWh dekkes maksimalt 5t 15min av gjennomsnittlig forbruk gjennom dagen. Forbruket time for time følger ikke gjennomsnittet, men ligger lavt gjennom natten og midt på dagen, og høyt et par timer om morgenen og ettermiddagen når strømprisen typisk er høy. Er sistnevnte tilfellet kan batteriet lades ut på under 2 timer. Generelt ut fra figur 39 lades batteriet opp i den billigste timen og lades ut når strømprisen er høyest.



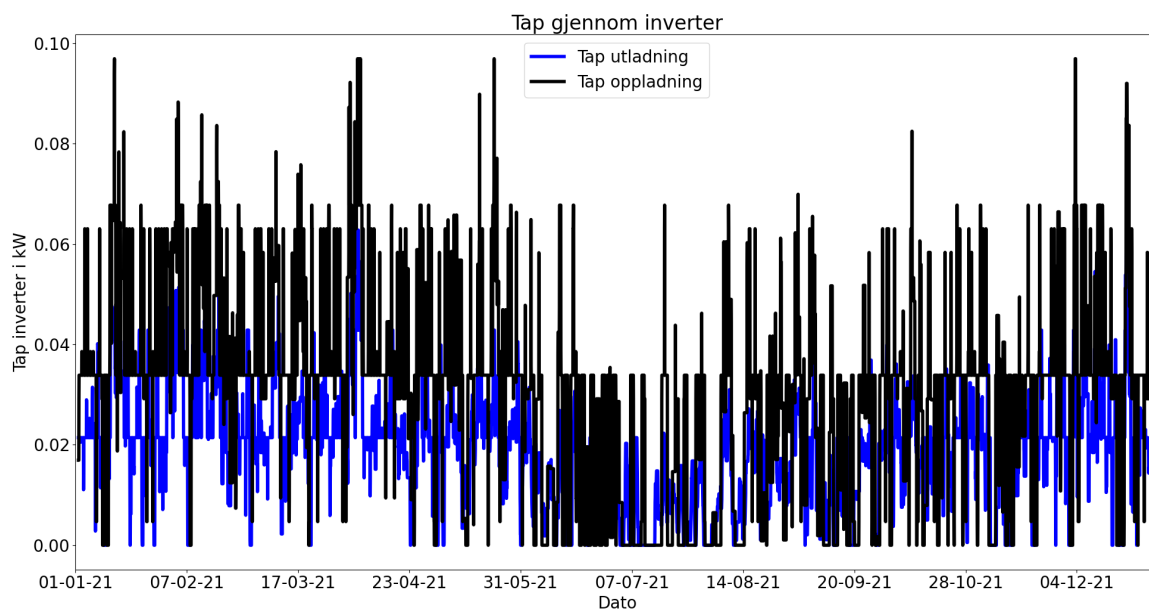
Figur 40: Akkumulert relativ besparelse på nettleie

Nettleie besparelse relativt til å ikke ha et batterisystem stiger jevnt, men avtar litt etter hvert. Det skyldes at mot høsten 2021 var strømprisen mye høyere og optimaliseringalgoritmen har heller utnyttet strømprisens svingninger enn å optimalisere for å få lavest mulig nettleie.



Figur 41: Salg av strøm tilbake til netteier

Det blir nærmest ikke solgt tilbake strøm på nettet på grunn av avgifter 27, fordi strøm selges til spotpris i markedet. Det gjør at det i et normalt marked blir nærmest umulig å kjøpe strøm fra nettet og selge det tilbake for en privat person.



Figur 42: Tap av energi gjennom invertere

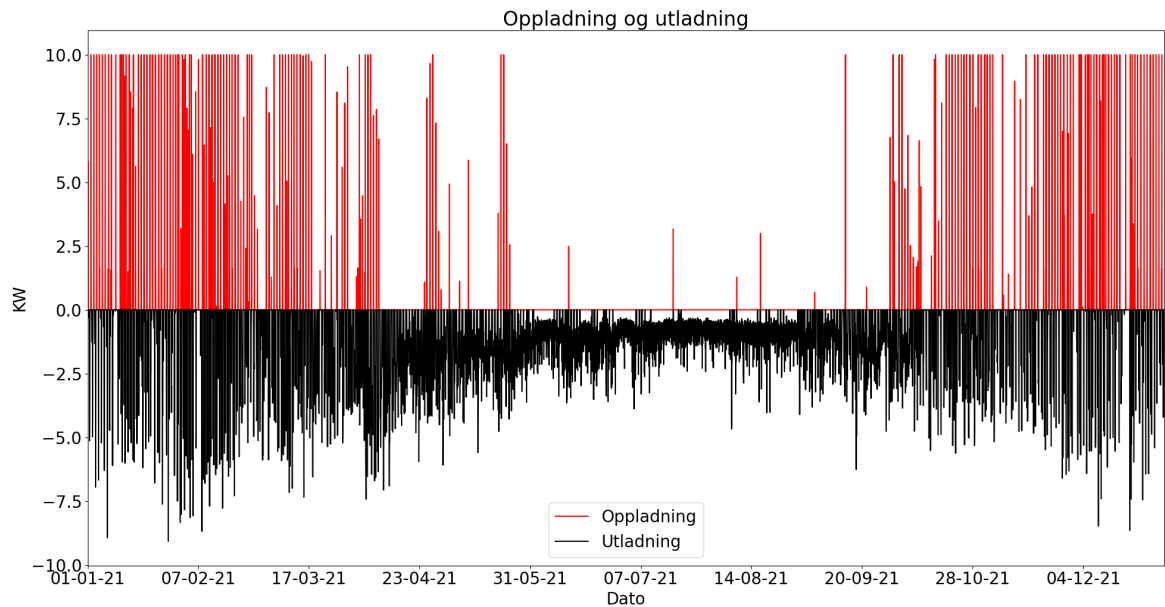
Tapet gjennom inverter til oppladning er litt høyere enn tapet til inverter utladning. Tapet til inverter utladning ble satt til konstant 5% i optimaliseringsproblemet og tap inverter utladning følger karakterstikken bestemt i figur 34. Figuren 42 viser derfor forventet atferd gitt begrensningene.

5.2 Ytelse med batteristørrelse på 13.5kWh for enebolig med solceller i pris-sone Kristiansand 2021



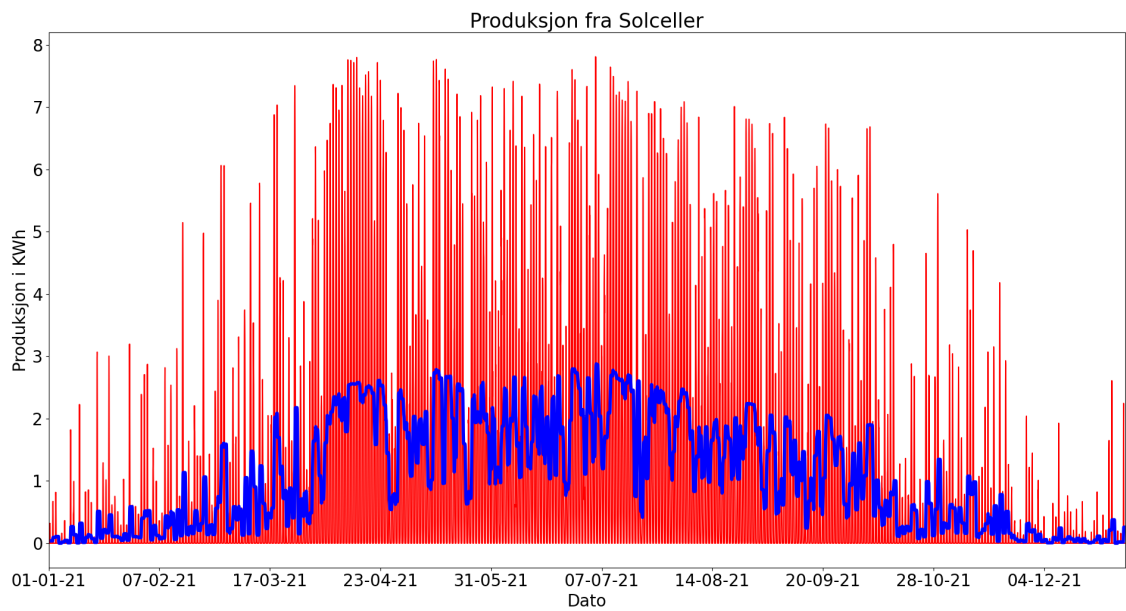
Figur 43: Oversikt over resultat for 13.5KW enebolig med solceller Kristiansand 21

Her tilegnes husstanden diskutert tidligere et solcellepanel. Referansen settes til at huset har solcellepaneler, men også at all overskuddsenergi selges direkte til spotpris. Det gir en reduksjon på 12,1% tilsvarende 360 EUR iløpet av 2021. Dette kommer i all hovedsak fra sommerperioden da solcellene produserer strøm. Da har grafen for akkumulert besparing raskest stigningsrate.

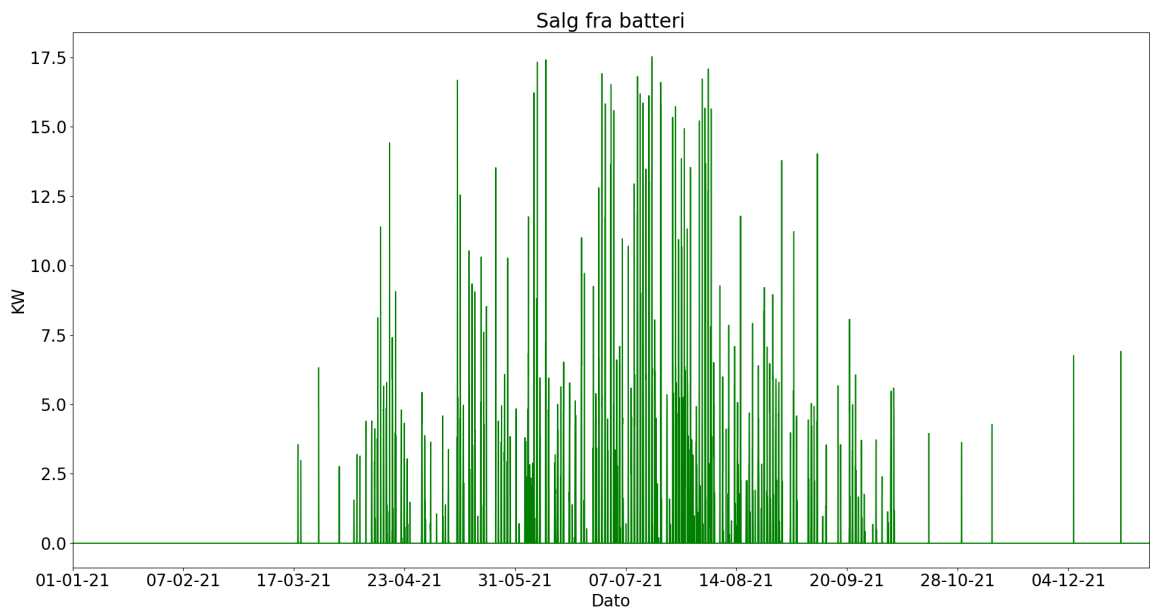


Figur 44: Oppladning og utladning for hver enkelt time gjennom 2021

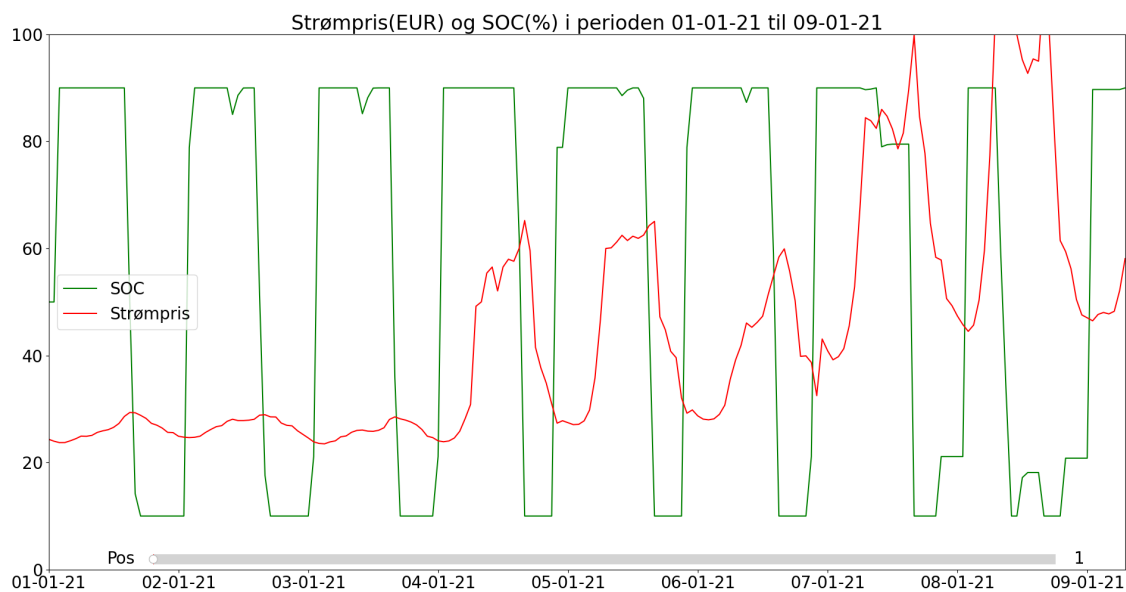
I midten av figur 44 er det svært få røde linjer, som vil si at batteriet ikke opplades med strøm fra nettet. Det er typisk for sommeren, og vi ser lite oppladning i perioden 31.05-30.08. Produksjonen fra solceller 45 overstiger forbruket, og det blir derfor nødvendig å selge strømmen i perioder vist i figur 46. I perioden midt på sommeren blir det trukket strøm fra batteriet kontinuerlig. Da solceller dekker hele det daglige behovet og strømmen kan lagres på batteri, blir det ikke behov for å kjøpe strøm fra nettet.



Figur 45: Produksjon fra solceller

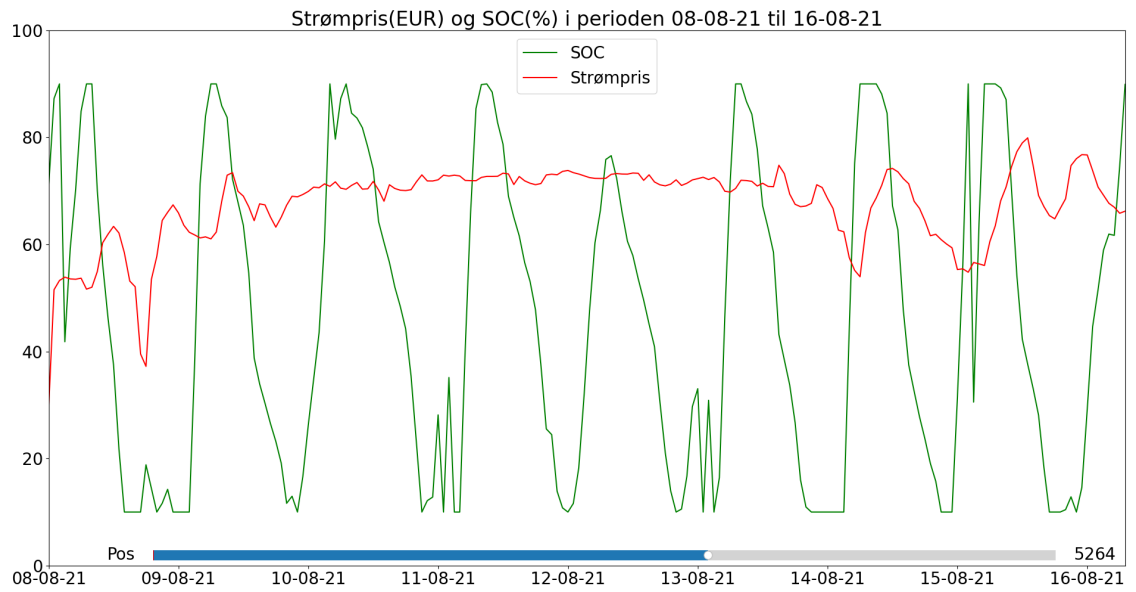


Figur 46: Salg av strøm tilbake til netteier

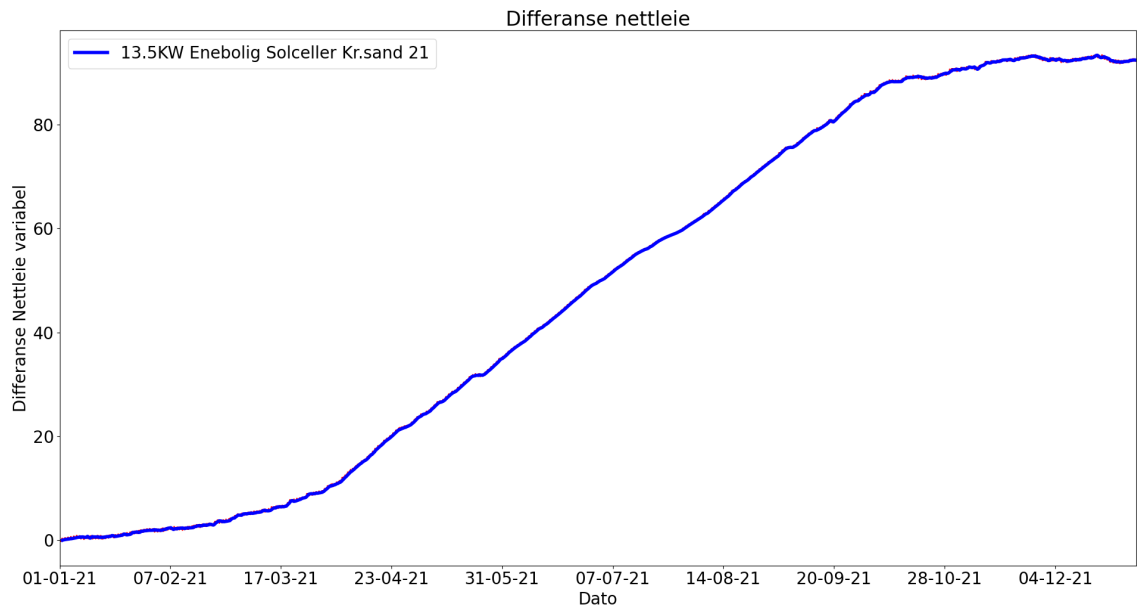


Figur 47: State of Charge mot strømprisen

I vinterperioden er strømforbruket høyest og produksjonen fra solcellene lavest. Oppførselen til systemet er tilnærmet likt med og uten solceller om vinteren. Om sommeren derimot, vist i figur 48, lader solcellene opp batteriet på dagtid som sakte utlades gjennom natten. Den 11.08 og 13.08 er det ikke nok strøm på batteriet til å dekke hele kvelden og natten, derfor blir det nødvendig å lade opp batteriet litt fra strømmettet.

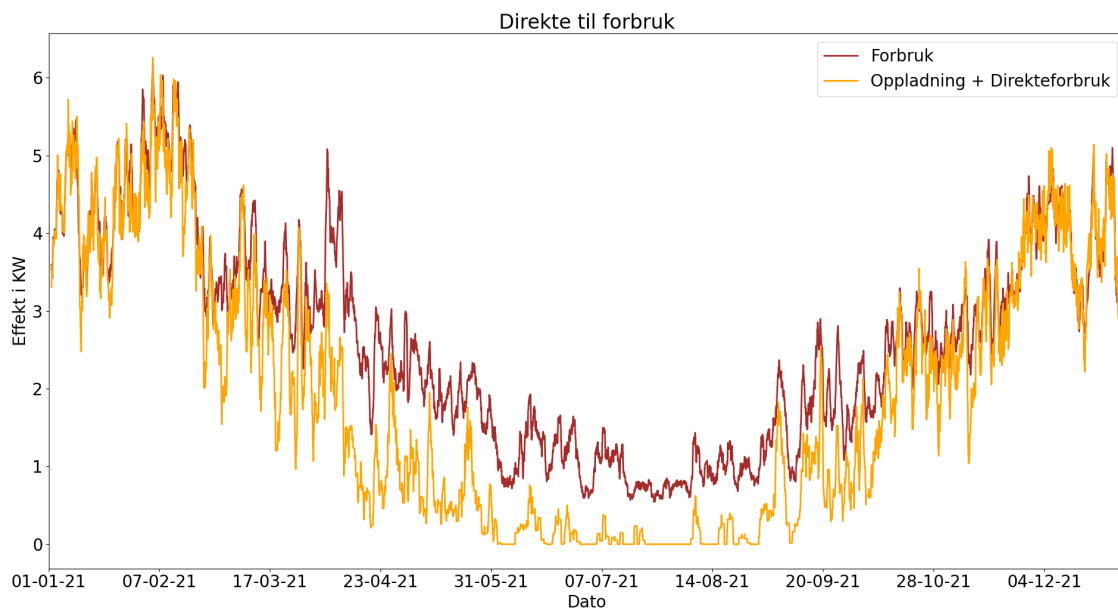


Figur 48: State of Charge mot strømprisen



Figur 49: Akkumulert relativ besparelse på nettleie

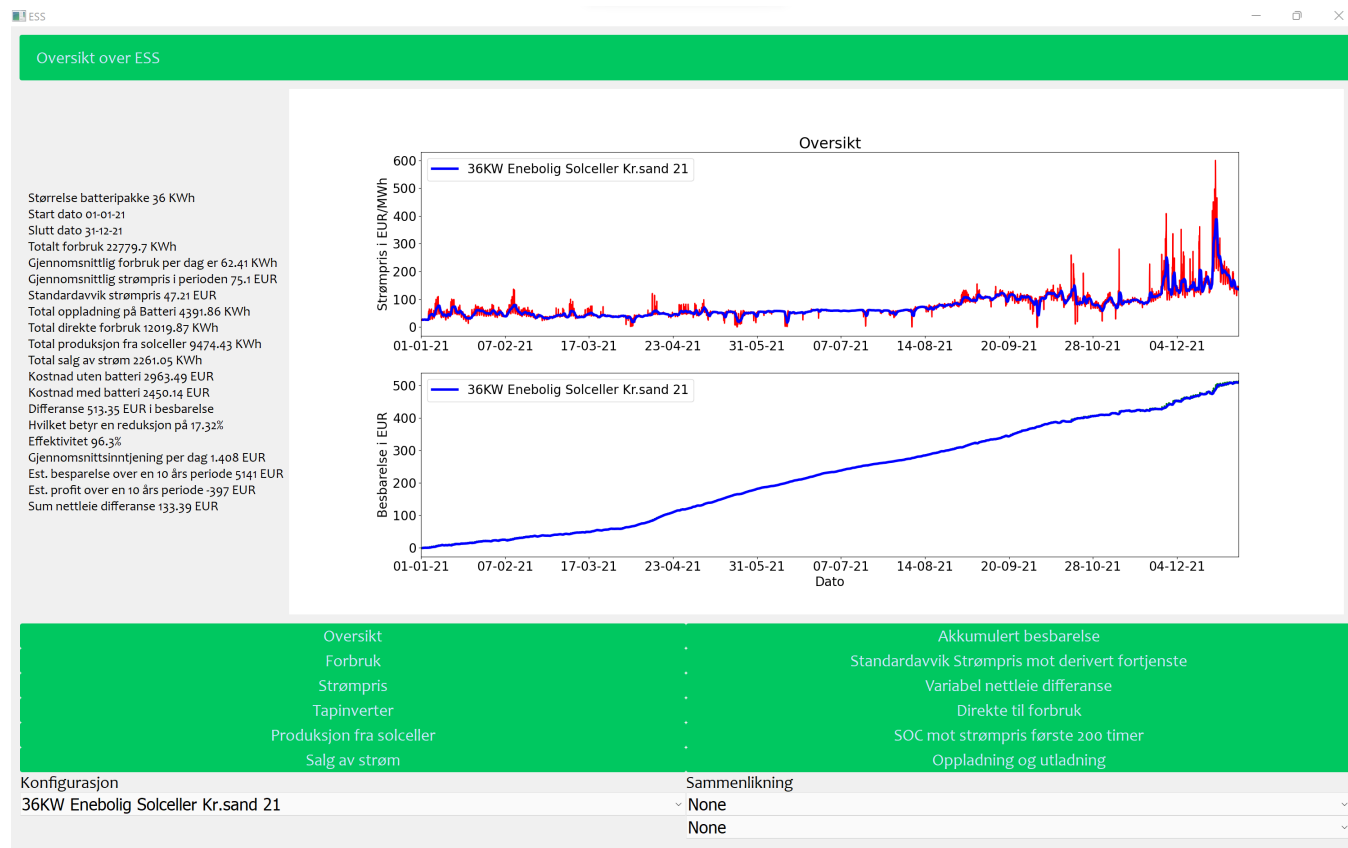
Besparelsen på nettleien er desidert størst i perioden solcellene produserer nok strøm til å dekke det daglige behovet. Gitt at forbruket er lavere enn energimengden på batteriet gjennom natten, kan nettleie elimineres da det ikke trekkes strøm fra nettet. Derav er det ingen nettleie på det variable leddet. Fast-leddet på nettleie må fortsatt betales men dette er ikke blitt tatt hensyn til i oppgaven, fordi det er en kostnad som er uavhengig av om **EBS** er installert.



Figur 50: Forbruk mot oppladning + strøm som ikke går gjennom **EBS**

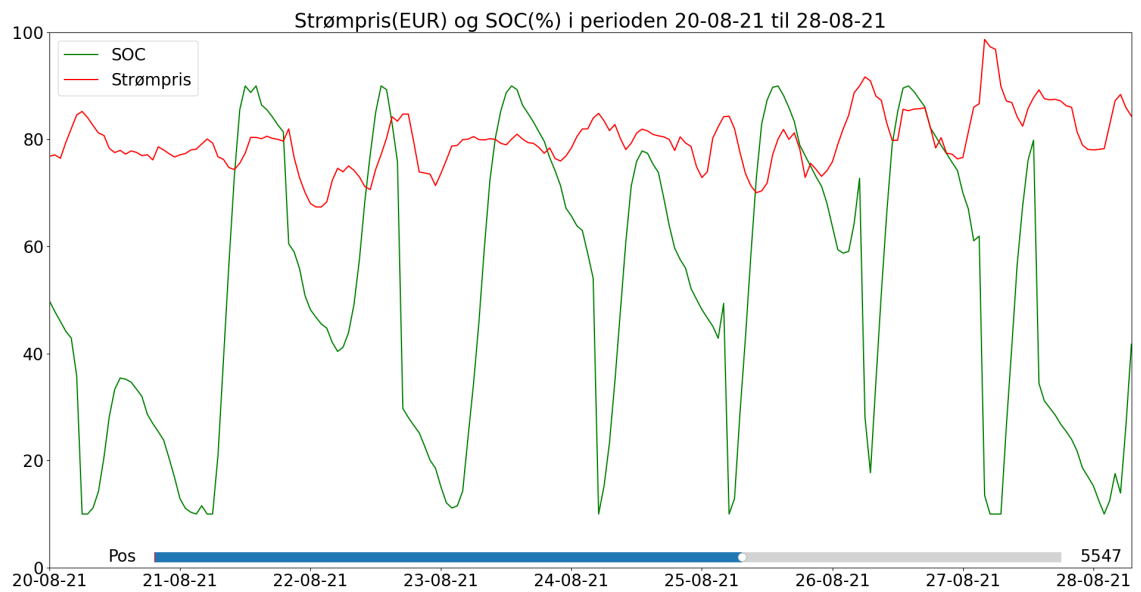
Figur 50 viser at summen av oppladning og forbruk som ikke går gjennom **EBS** er mindre enn forbruket i sommerperioden. Differansen mellom de 2 grafene er produksjonen fra solceller. Det at den gule grafen ligger på 0KW betyr at det ikke blir brukt strøm fra nettet i det hele tatt. Det er i samme periode det spares mest på nettleie 49.

5.3 Ytelse med batteristørrelse på 36 kWh for enebolig med solceller i pris-sone Kristiansand 2021



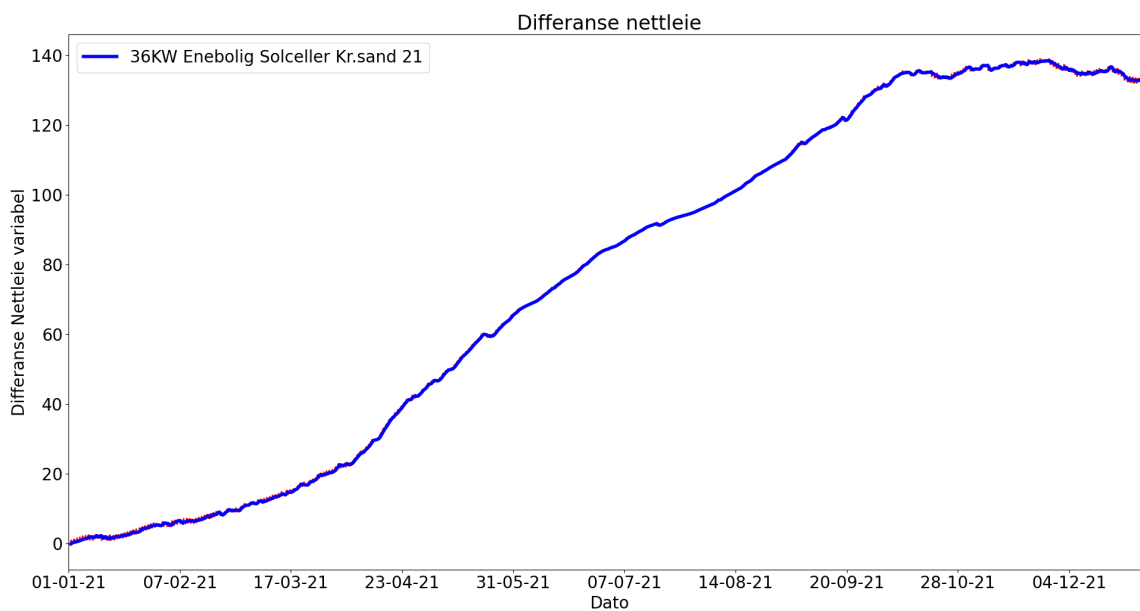
Figur 51: Oversikt over resultat for 36KW enebolig med solceller Kristiansand 21

En økning fra 13,5 kWh batteripakke til 36 kWh medfører økning i besparelsen fra 12,1% til 17,3%, som er 5,2% mer når batteripakken øker med 22,5kW, nesten tredobling av størrelse. Besparelsen i EUR går fra 360 til 513, som er en differanse på 154 EUR. Total oppladning på batteri fra strømmettet er 4392 kWh mot 2021 kWh for batteripakke på 13,5kWh. Det betyr at batteriet går gjennom færre fulle sykluser enn batteriet på 13,5KW. Det er forventet da kapasiteten er større. Det økte volumet har gjort at batteriet kan levere strøm til huset over lengre tid, men også at marginen mellom innkjøpspris synker. Mesteparten av besparelsen kommer fra perioden solcellene produserer strøm, og den økte kapasiteten gjør at enda mindre strøm må kjøpes om sommeren.



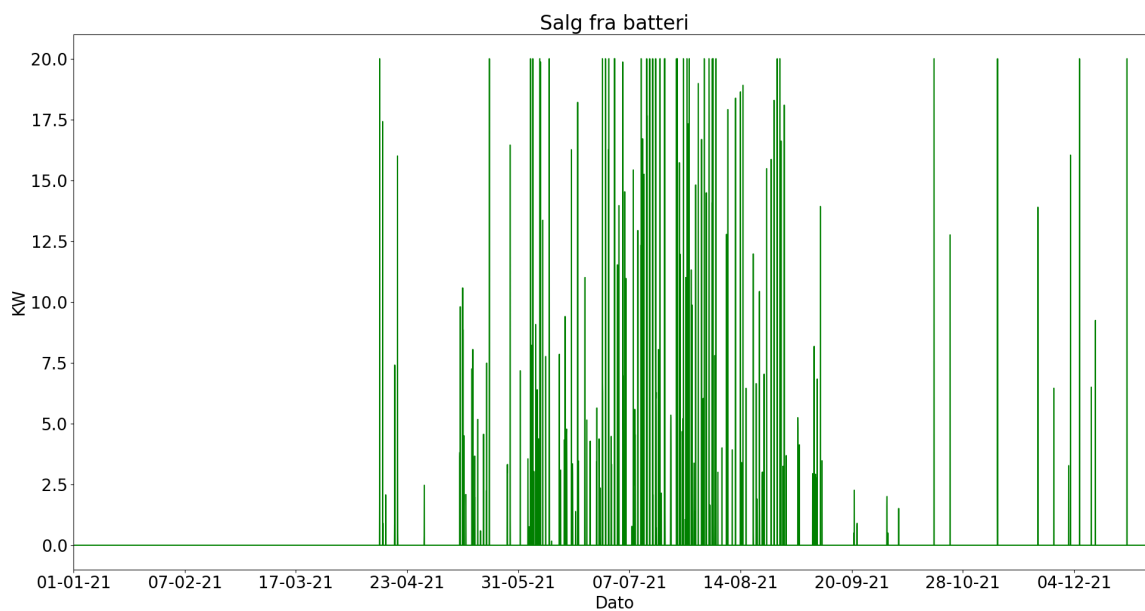
Figur 52: State of Charge mot strømprisen

Figuren 52 viser at batteriet utnytter hele kapasiteten fra 10% State of Charge til 90% State of Charge. Det kan tyde på at batterikapasiteten kan økes ytterligere for å øke besparelsen. Besparelsen som har økt med 5,2% gitt 22,5kWh mer batterikapasitet, illustrerer at besparelsen vokser saktere enn batterikapasiteten, og tyder på at et mindre batteri vil gi høyere avkastning på investeringen.



Figur 53: Akkumulert relativ besparelse på nettleie

Besparelsen på nettleie er 40,9 EUR høyere enn med batteripakke på 13,5kWh. Det er forventet at besparelsen skal være høyere fordi mer strøm kan lades når nettleien er billigst. Den andre fordelen er at mer ladning fra solceller kan lagres på batteriet. Det at økningen ikke er mer kan tyde på at batteripakken er for stor.

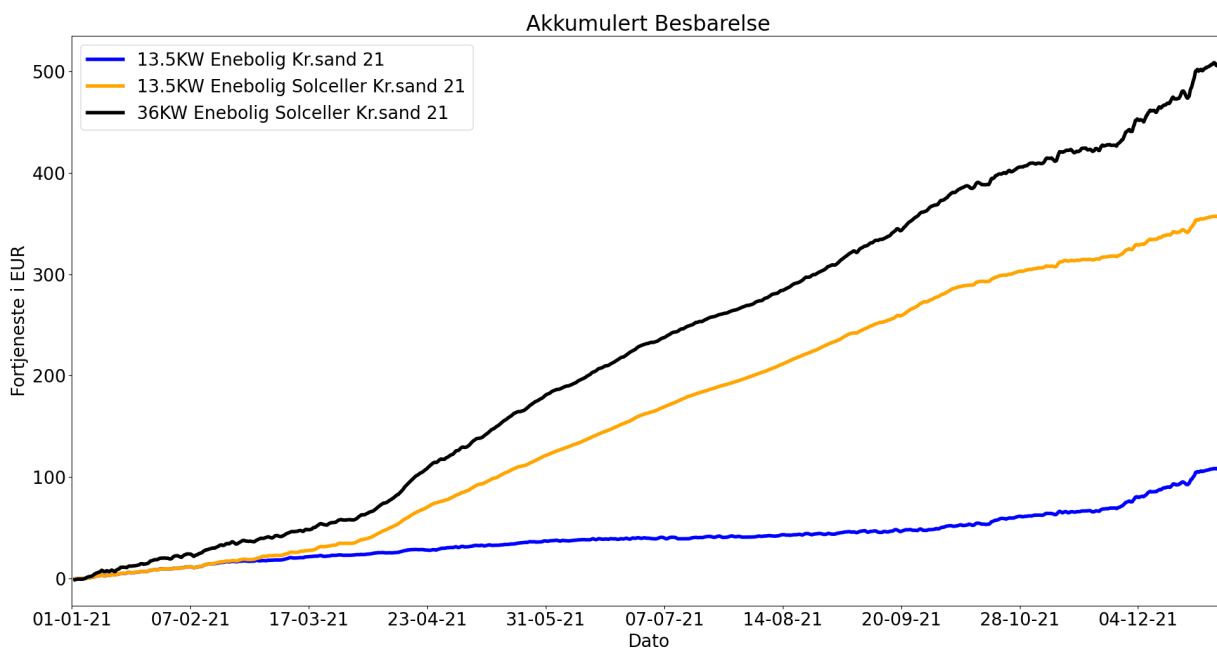


Figur 54: Salg av strøm tilbake til netteier

Det blir solgt 634 kWh mindre strøm tilbake på nettet. Det er strøm fra solcellene som heller har blitt lagret på batteriet og brukt til å dekke eget forbruk. Det er satt en maksimal grense på tilbakesalg på 20KW per time. Dersom denne fjernes kunne utslaget vært større fordi en større mengde strøm kan selges den timen prisen er høyest.

De 3 ulike situasjonene 13.5 kWh enebolig Kristiansand 2021, 13.5 kWh enebolig med solceller Kristiansand 2021 og 36 kWh enebolig med solceller Kristiansand 2021 er oppsummert i tabell og figur 55

Situasjon	13,5 kWh EB	13,5 kWh EB m/sol	36 kWh EB m/sol
Reduksjon strømpris	2,9%	12,13%	17,32%
Total besparelse	111 EUR	360 EUR	513 EUR
besparelse nettleie	19,2 EUR	92,5 EUR	133 EUR
Oppladning på batteri fra nettet	3794 kWh	2021 kWh	4392 kWh
Effektivitet	98,1%	97,6%	96,3%

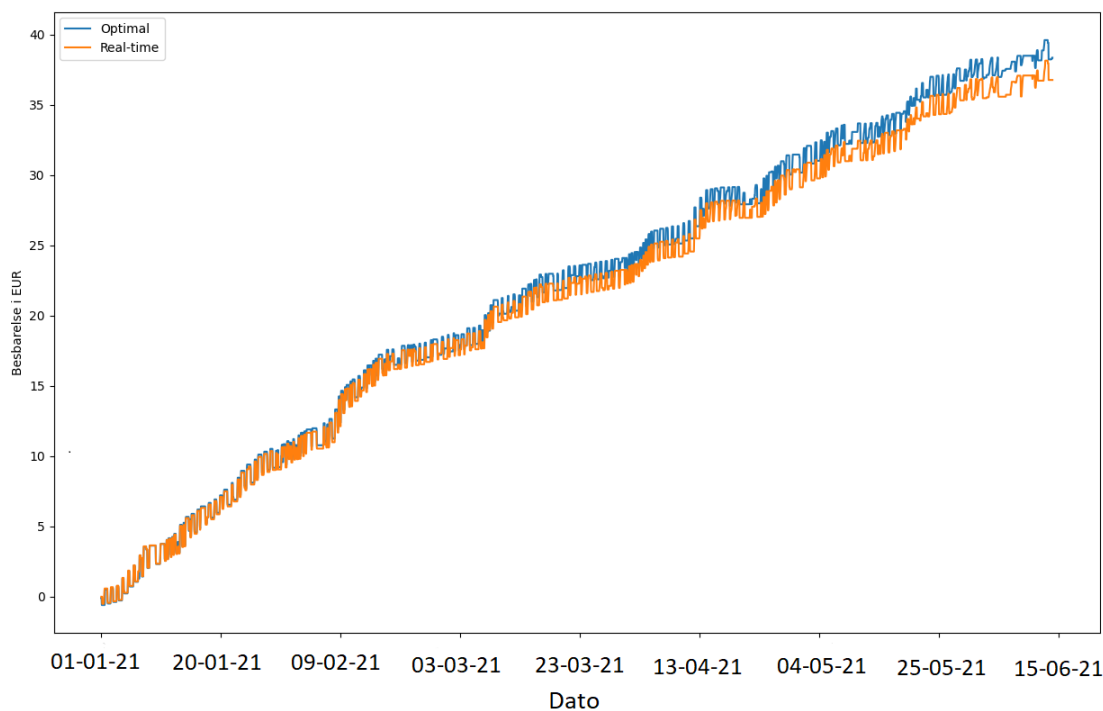


Figur 55: Akkumulert besparelse over hele 2021 for de 3 ulike situasjonene

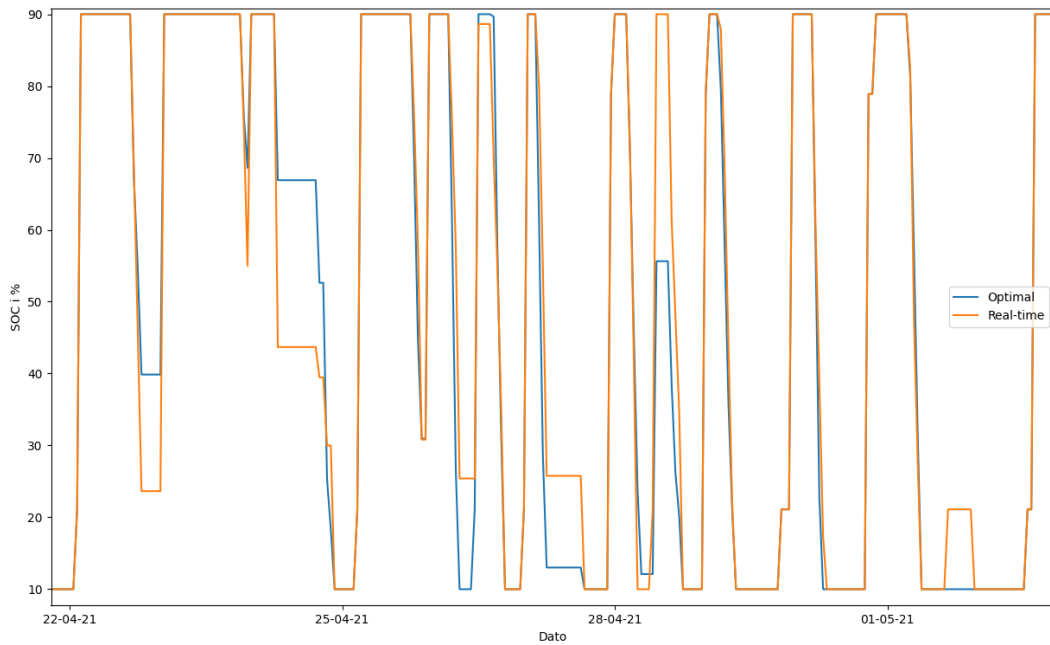
5.4 Sanntidskjøring

Tidligere har det blitt vist hvor mye det er mulig å redusere systemkostnaden med optimal bruk av **EBS** gitt begrensingen som er satt. Når systemet skal kjøres i sanntid vil det være tilnærmet umulig å oppnå samme resultat. Hver feil som blir gjort under sanntidskjøring er umulig å hente inn igjen siden det ikke er mulig å gjøre det bedre enn optimalt resultat som betyr at i beste fall følger sanntid-kjøring den optimale kurven.

For å gjøre sanntidskjøringen kjøres LP-problemet for hver time der en antar at strømforbruket de neste 24 timene blir lik forrige 24 timer. Strømprisen oppdateres hver dag kl 13.00 hos *Nordpool*. Da publiseres strømprisen for kommende dag. Det betyr at en har strømprisen for de neste 36 timene. Rett før kl blir 13 igjen dagen etterpå har en kun igjen 12 timer med strømpris. Det betyr at når Day-ahead strømprisen som er tilgjengelig går under 24timer med data settes timen som ikke har verdi til prisen som er den siste timen som er tilgjengelig. Under kjøring med solcellerpaneler settes forventet produksjon fra solceller de neste 24 timer til snittet av de 3 forrige dagene. Dette for å snitte for dager med tett skydekke og dager med full sol. Det gir en prediksjon som ikke er perfekt men som stemmer bedre enn dersom kun forrige dag brukes. Simuleringen kjøres over et halvt år.



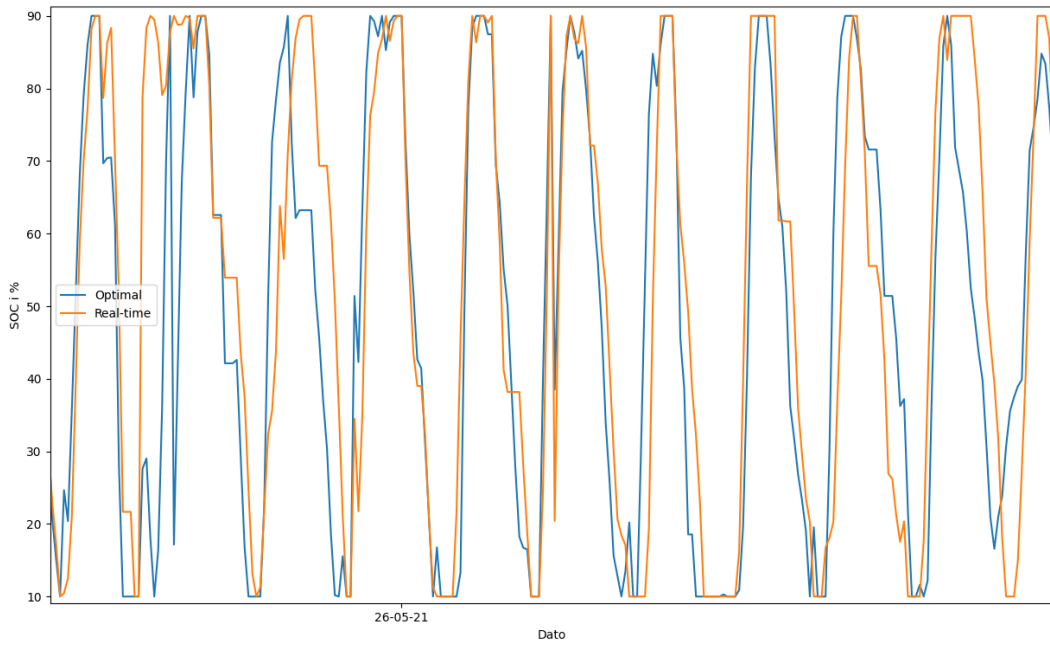
Figur 56: Akkumulert besparelse over et halvt år



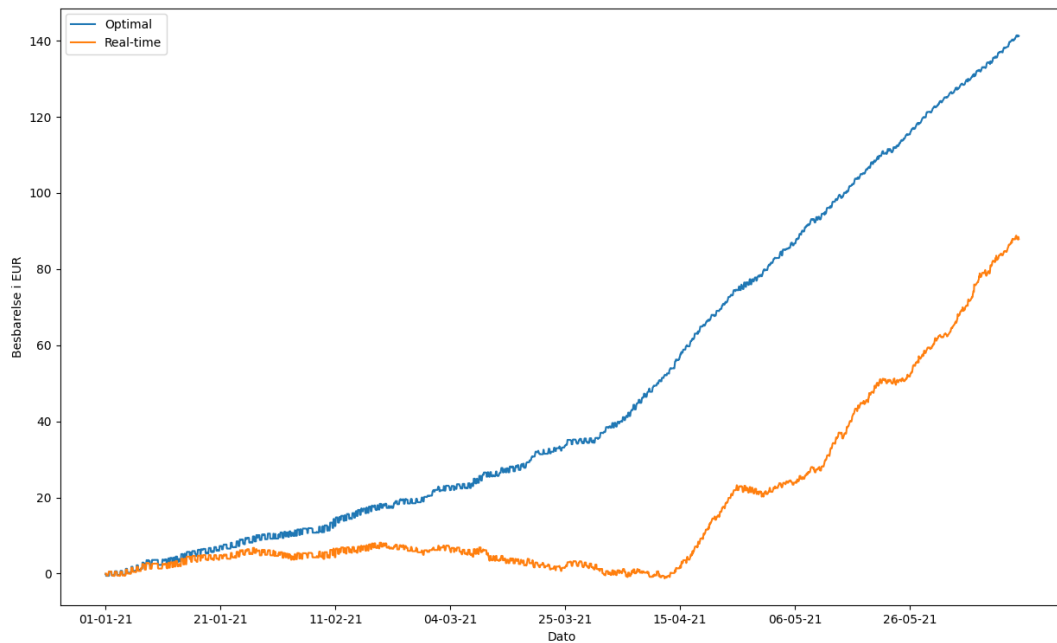
Figur 57: *State of Charge* optimal mot sanntids kjøring

Situasjon	sanntid uten solceller	Optimal uten solceller
Reduksjon strømpris	2,32%	2,37%
Total besparelse	37,6 EUR	38,5 EUR
Besparelse nettleie	12,3 EUR	12,36 EUR
Oppladning på batteri fra nettet	2103 kWh	2098 kWh
Effektivitet	98,1%	98,1%

Testen av sanntidskjøring uten solceller montert 56 viser at besparelsen er tilnærmet lik som for et ideelt **EBS**. Dette skyldes at strømforbruket er syklisk med 24 timers intervall og strømpris er kjent mellom 12 og 36 timer frem i tid. Dette betyr at det er liten usikkerhet i tallene som brukes til å predikere fremtiden. Figur 57 viser at *State of Charge* beveger seg tilnærmet likt som for ideelt **EBS** med noen få variasjoner. Modellen antar at forbruk idag blir identisk med forbruket igår. Variasjonen skyldes derfor at forbruket variere fra dag til dag, for eksempel at middagen blir laget noen timer tidligere eller at mange elektriske maskiner brukes på samme tid i motsetning til forbruket dagen før.



Figur 58: State of charge optimal mot realtids kjøring



Figur 59: Akkumulert besparelse over et halvt år

Situasjon	sanntid med solceller	Optimal med solceller
Reduksjon strømpris	7.02%	11.19%
Total besparelse	88.7 EUR	141 EUR
Besparelse nettleie	36.9 EUR	42.0 EUR
Oppladning på batteri fra nettet	1060 kWh	1109 kWh
Effektivitet	97.7%	97.7%

Med solceller montert 59 blir det større ulikhet mellom ideell og sanntid på 52 EUR. Dette er fordi en ny variabel, egenprodusert energi, blir introdusert. Denne er vanskeligere å predikere, fordi den avhenger av værdata. Selv om solcelleproduksjon til en viss grad er syklisk fordi solen står opp og går ned omtrent på samme tid variere effekten av skydekket. I modellen brukes snitt av 3 forrige dager for å bedre ytelsen, men det er ikke tilstrekkelig på grunn av værrets uforutsigbarhet. For diskusjon rundt sanntidskjøring se underkapittel 8.2

6 Økonomisk oversikt

De to hovedkomponentene i systemet er batteripakken og inverteren. Solcellene blir ikke tatt med i regnskapet da det er tenkt som en del etablert mikronett og ikke selve **EBS**. Nevnt tidligere blir det tatt utgangspunkt i de to ulike batteripakkene *Tesla Powerwall* på 13.5 kWh og *Evyon* sin batteripakke på 36 kWh som er laget av brukte elbil batterier. Hos *Evyon* kreves det en ekstern inverter mens *Tesla Powerwall* har en innebygd inverter. Inverteren som blir brukt sammen med *Evyon* sin batteripakke er *Growatt SPA4000 10000TL3 BH*. I tillegg til *Tesla Powerwall* og *Evyon* ses det på hva kostnad blir dersom en tar utgangspunkt i markedspris for litium-ion batteripakker målt i EUR/kWh. Prisene er vist i tabeller under.

	Batteripakke	Inverter	Sum
13.5 kWh <i>Tesla Powerwall</i>	10300 EUR	inkludert i batteripakke	10300 EUR
36 kWh <i>Evyon</i> (Brukte elbilbatterier)	10000 EUR	2650 EUR	12650 EUR
13.5 kWh Markedspris(124 EUR/ kWh)	1674 EUR	2650 EUR	4324 EUR
36 kWh Markedspris(124 EUR/ kWh)	4464 EUR	2650 EUR	7114 EUR

For å kunne seie om batteripakker er lønnsomt beregnes intern rente(**IRR**) og netto nåverdi(**NNV**) av prosjektet der ulike avkastningskrav og kontantstrømmer testes. Avkastningskravene som blir brukt er 2,5% fordi det er forventet at styringsrenten skal opp mot 2,5% allerede i 2024 ifølge rentebanen til Norges bank [5]. Derfor antas det også at sparerenten hos bankene kommer opp på dette nivået og det settes derfor som et minimums avkastningskrav. I tillegg brukes 7%, tar utgangspunkt i gjennomsnittlig avkastning på akjsemarkedet trukket fra inflasjon [32]. Disse to verdiene anses derfor som representative iforhold til andre investeringer enkelt tilgjengelig for alminnelige husholdninger. Det tas utgangspunkt i besparelsen fra 2021, dette kan potensielt være optimistiske tall men grunnet den anspente energisituasjonen og overgangen til fornybar energi kan det forsvares at energiprisene skal opp dog var gjennomsnittsprisen for strømmen i 2021 75.1 EUR mens statnett anslår at den skal stabilisere seg på 45 EUR [44]. Rapporten tokk ikke hensyn til krigen i Ukraina og det kan derfor være en større sannsynlighet for at prisen i nærmeste fremtiden ligger noe høyere enn 45 EUR. I tillegg er den økte volatiliteten på strømprisen som tydelig kom til syne i 2021 forventet å vedvare og øke etterhvert som ikke styrbar produksjon øker. Det testes i tillegg med vekst i besparelse tilsvarende inflasjonen. Det er ikke forventet at strømprisen skal stige tilsvarende årlig inflasjonen men for å se hvordan en økt besparelse påvirker avkastningen på investering undersøkes det i tillegg. Kontantstrømmen som blir brukt er 360 EUR årlig i 15år for uten inflasjonsvekst og 360 EUR i 15 år med en årlig vekst på 2% for med inflasjonvekst

Uten inflasjonvekst	IRR	NNV 2,5%	NNV 7%
13.5 kWh <i>Tesla Powerwall</i>	-7,16%	-5843 EUR	-7021 EUR
36 kWh <i>Evyon</i> (Brukte elbilbatterier)	-5,65%	-6298 EUR	-7978 EUR
13.5 kWh Markedspris(124 EUR/ kWh)	2,92%	133 EUR	-1045 EUR
36 kWh Markedspris(124 EUR/ kWh)	1,00%	-763 EUR	-2442 EUR

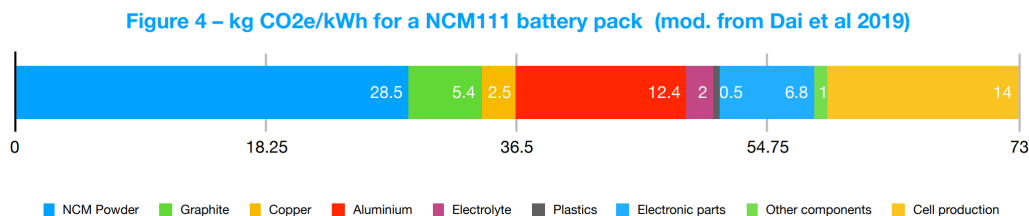
Med inflasjonvekst	IRR	NNV 2,5%	NNV 7%
13.5 kWh <i>Tesla Powerwall</i>	-5,51%	-5208 EUR	-6612 EUR
36 kWh <i>Evyon</i> (Brukte elbilbatterier)	-3,98%	-5394 EUR	-7395 EUR
13.5 kWh Markedspris(124 EUR/ kWh)	4,70%	768 EUR	-636 EUR
36 kWh Markedspris(124 EUR/ kWh)	2,76%	142 EUR	-1859 EUR

Bastert på 2021 data blir resultatet at ingen av de ferdig løsningene *Tesla Powerwall* og *Evyon* har en positiv internrente eller Netto nåverdi. Det betyr at for *Tesla Powerwall* forsvinner 7,16% av investeringen på 10300 EUR årlig gjennom hele perioden på 15år. Noe som betyr at netto nåverdi med 2,5% avkastningskrav blir -5843 EUR. Det er noe bedre for *Evyon* sine brukte elbilbatterier men fortsatt langt fra å være lønnsomt. Det betyr at med forutsetningen og begrensingene brukt så lønner det seg ikke økonomiske å kjøpe **BMS**. Strømprisen og svingengene på denne kan ikke styres av enkeltpersoner noe som betyr at den viktigste faktoren for å få **EBS** til å bli lønnsomt er lave investeringskostnader. Når markedspris for litium-ion batteripakker brukes får en netto nåverdi med 2,5% avkastningskrav på 133 EUR. Det er ikke en god investering men dersom alternativet er å beholde pengene på en sparekonto kan de samfunnsmessige fordelene med **EBS** gjøre at noen ønsker å installere slike systemer siden det hverken blir en god investering men heller ikke en økonomisk belastning. Det er også verdt å merke at dersom strømprise faller fra nivåene i 2021 som er sannsynlig at den gjør på sikt blir resultatet større negative netto nåverdier og internrenter.

7 Miljøregnskap

Oppgaven modellerer et teoretisk **EBS**, men vil i praksis bestå av flere elektriske komponenter nødvendige for installasjon, som invertere, solcellepanel og *Maximum Power Point Tracking* (**MPPT**). Batteripakken ansees som høyst miljøpåvirkende og vil derfor vurderes i størst grad. Batteriproduksjon og resirkulering av batterier påvirker begge miljøet. Utvinning av råvarer, transport og gjenvinning av batterier har et betydelig klimaavtrykk.

Et **EBS** kan bestå av både brukte og nye battericeller. Bruken av nye battericeller krever utvinning av nye råvare eventuelt at råvarene delvis kommer fra resirkulerte batterier. Dette krever både store mengder energi under produksjon og CO2 utslipp i forbindelse med utvinningen av råvare.



Figur 60: Fordeling av CO2 utslipp basert på råvaretype [11]

Totalt slippes det ut 73 kg CO₂e/kWh, *NCM POWDER* står for størst andel av utslippet dette er utvinning og raffinering av Nikkel, Mangan og Kobolt som står for 28,5 kg CO₂e/kWh. Dernest er aluminium på 12.4 kg CO₂e/kWh. Tall fra Greenhouse Gases, Regulated Emissions, and Energy Use in Transportation (GREET) modellen[11]. I tillegg til de 73 kg CO₂e/kWh kommer det utslipp i forbindelse med transport av råvarer til og fra raffineriet, produksjonslokalt, og gruvene som typisk gjøres på lasteskip. Det er viktig å poengtere at data brukt i studiene som refereres til er sentrert rundt og omhandler litium-ion batterier hovedsaklig i el-bil industrien. Utslippet og andelen av utslippet råmaterialene står for begrenses til en spesiell katode-type kalt $LiNi_{1/3}Mn_{1/3}Co_{1/3}O_2$ **NMC111**. Anoden begrenses til grafit.

Bruken av brukte elbilbatterier i **EBS** skaper i teorien ingen negative tilleggsbelastninger på miljøet då istedenfor at battericellene går direkte til gjenvinning blir levetiden forlenget i tiden batteriet blir brukt som energilager, deretter resirkuleres det på samme måte som batterier som ikke blir brukt til energilagring.

Problemet med gjenvinning av brukte batterier er at kostnaden for brukte litium-ion celler ikke kan gå over en viss terskel, da gjenvinning i seg selv må være økonomisk lønnsomt uten subsidier. En forskningartikkel [54] konkluderer med at kostnaden for brukte litium-ion celler ikke kan overskride 2,87 dollar per kg^{-1} .

Råstoffmangel på kobolt, nikkel og litium som skrevet om i kapittel 1.2 og 2.2 er med på å gjøre nyproduserte såvel som brukte batterier vesentlig dyrere. I følge nevnte forskningsartikkel bruker hele gjenvinningsprosessen 149,80 MJ/kg energi, og har drivhusgassutslipp på 10,53 kg/kg. Produksjon av batterier fra råstoff bruker 163,81 MJ/kg energi og med et drivhusgassutslipp på 11,28 kg/kg.

Til sammenlikning slipper forbrenning av bensin ut 0.162 kg CO₂e/km for en bil som bruker 0.58 liter/mil [23]. Det betyr at utslipp fra produksjon av en batteripakke på 13.5 kWh skaper et utslipp på 985.5 kg CO₂e/kWh tilsvarende 608 mil med en bensin-bil.

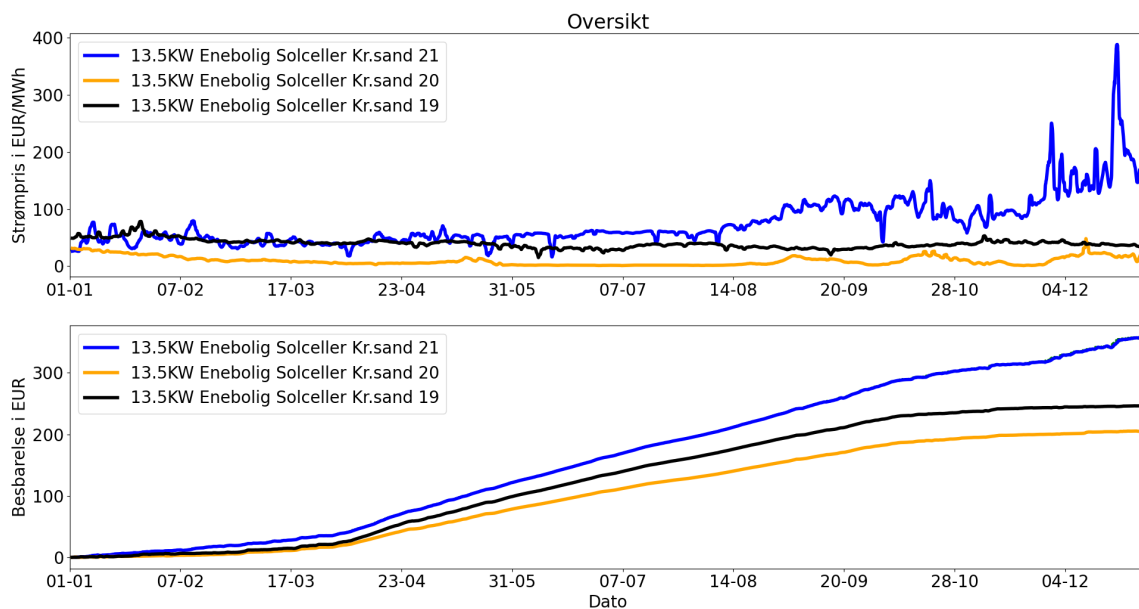
Det finnes miljømessige fordeler med at strømmettet blir bedre utnyttet. Dog i mindre i grad i Norge grunnet den store andelen vannkraft som er styrbare. I andre land som England og Tyskland som bruker gass til å stabilere nettet er miljøgevinsten i følge [8] på 1.98 MtCO₂e dersom batterier kan supplere 29.1% av balanseringkapasiteten til gass i UK. Tilsvarende forbruket av 611 bensinbiler som kjører 2000 mil om året.

8 Diskusjon

Ulike forbruksmønstre, prissoner og batteristørrelser testes og sammenliknes for å undersøke hvilken påvirkning de har på ytelsen til **EBS**. Det betyr større usikkerhet til resultatene da ulike årsforbruk og prissoner blandes på en unaturlig måte. Hvordan systemet ville påvirket strømmettet og fremtidsutsikter diskuteres før tekniske begrensninger og feilkilder tas opp. Forslag til videre arbeid med modellen utviklet drøftes til slutt.

8.1 Ulike forbruksmønstre og prissoner

Strømpriser 2019 til 2021 for Kristiansand(NO2)

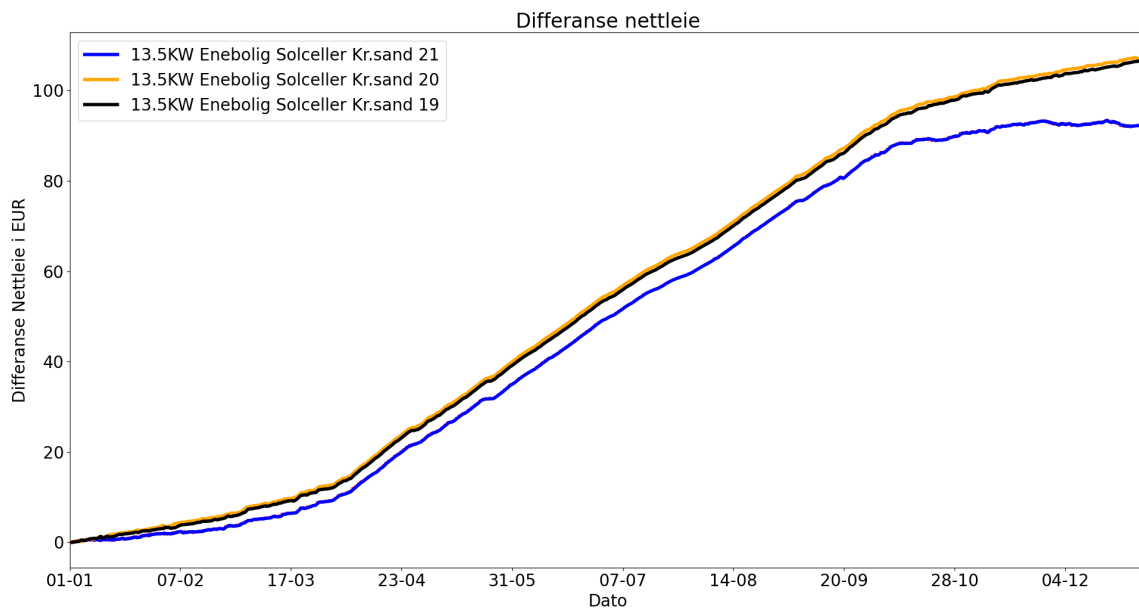


Figur 61: Sammenlikning akkumulert besparelse Enebolig Kristiansand 2019-2021

I 2019 var prissvingningene mindre, med et standardavvik på 8,23 EUR i motsetning til 47,21 EUR i 2021. Gjennomsnittsprisen i 2019 lå på 37,27 EUR mens snittprisen for 2021 lå på 75,1 EUR. Prisnivået i 2019 vurderes som mest representativ for fremtiden av brukte datasett, da de ligger nærmere historisk strømpris i Norge. Historisk har strømprisen vært relativt stabil grunnet den høye andelen vannkraft. Fremover er det forventet at prissvingningene øker, men 2019 kan fortsatt være et godt referanseår for å unngå for optimistiske resultat. 2020 var et år med unormalt lave strømpriser delvis drevet av nedstengingen av samfunnet som følge av covid-krisen. Prisnivået som fulgte av krisen forventes ikke å vedvare i overskuelig fremtid, men prissvingningene som følge av overgang til ikke-fleksible energikilder forventes å øke. Overgangen til fornybar kraftproduksjon har blitt fremskyndet av økonomiske sanksjoner mot Russland.

Med 2019 prisene klarte modellen å redusere strømkostnaden med 11,6% tilsvarende 247 EUR i motsening til 12,1% tilsvarende 360 EUR i 2021. Størst besparelse i prosent var i 2020 på 13,1% tilsvarende 206 EUR. Årsaken til ulikhetene i besparelse og reduksjon i strømpris kommer av flere årsaker. Total strømkostnad uten batteriet varierer med gjennomsnittsprisen på strøm gjennom året. Det fører til at reduksjon i % blir størst i 2020 da strømprisen er lavest, og styrkes av at 108 EUR av de 206 EUR kommer fra besparelse på nettleie.

Det viser at svingingene på strømprisen har mindre å si for besparelse når solceller eller andre energikilder er installert. Besparelsen kommer hovedsaklig fra økt konsum av egenprodusert energi som følge av batterilagring, også kalt *load-shifting*. *Load-shifting* vil si å flytte tidspunktet for strømkonsum, og muliggjøres av å ha tilgang til oppsamlet egenprodusert energi. Hvis **EBS** med installert energikilde kan dekke boligens strømbehov i perioder med høye strømpriser, anses det av algoritmen som en mer besparende strategi enn å handle strøm og utnytte prissvingningene på kort sikt. Algoritmen har mer tilgjengelig energi og kan utnytte prissvingningene på lengre sikt.



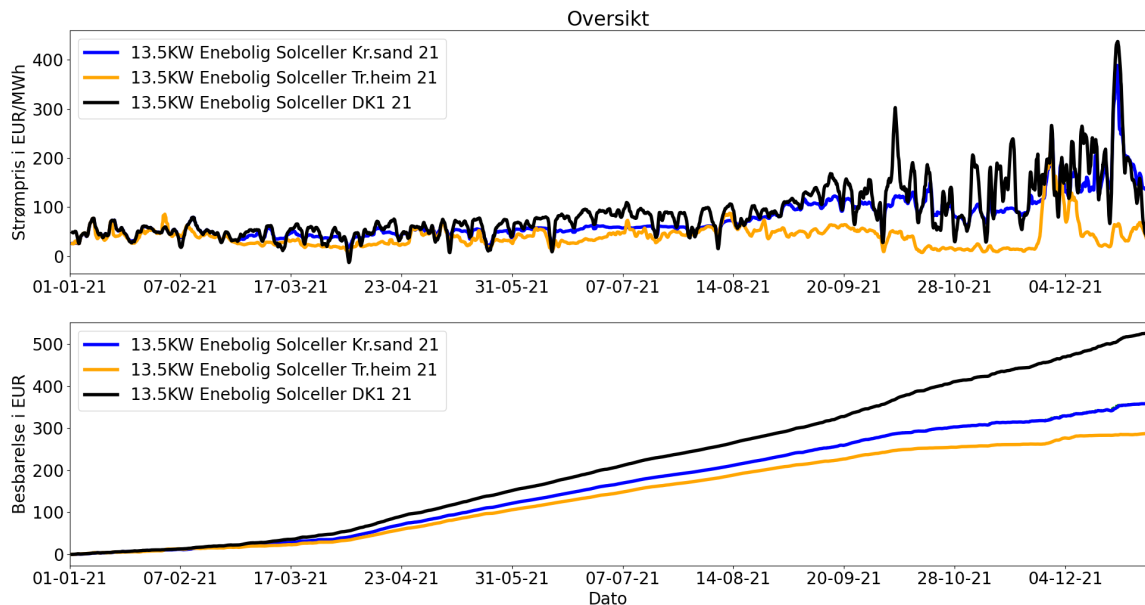
Figur 62: Akkumulert relativ besparelse på nettleie 2019-2021

Figur 62 viser at besparelsen på nettleie er størst for Kristiansand 21. Det skyldes i stor grad at den økte strømprisen høsten 21 fører til at modellen velger bort å optimalisere for redusert nettleie og heller lader batteripakken når strømprisen er lav. Relevant data samlet i tabell under.

Situasjon	2021	2020	2019
Reduksjon strømpris	12,13%	13,07%	11,55%
Total besparelse	360 EUR	206 EUR	247 EUR
Besparelse nettleie	92,5 EUR	108 EUR	107 EUR
Oppladning på batteri fra nettet	2021 kWh	2000 kWh	1958 kWh
Effektivitet	97,6%	97,6%	97,6%

Sammenlikning av prissonene Jylland, Kristiansand og Trondheim

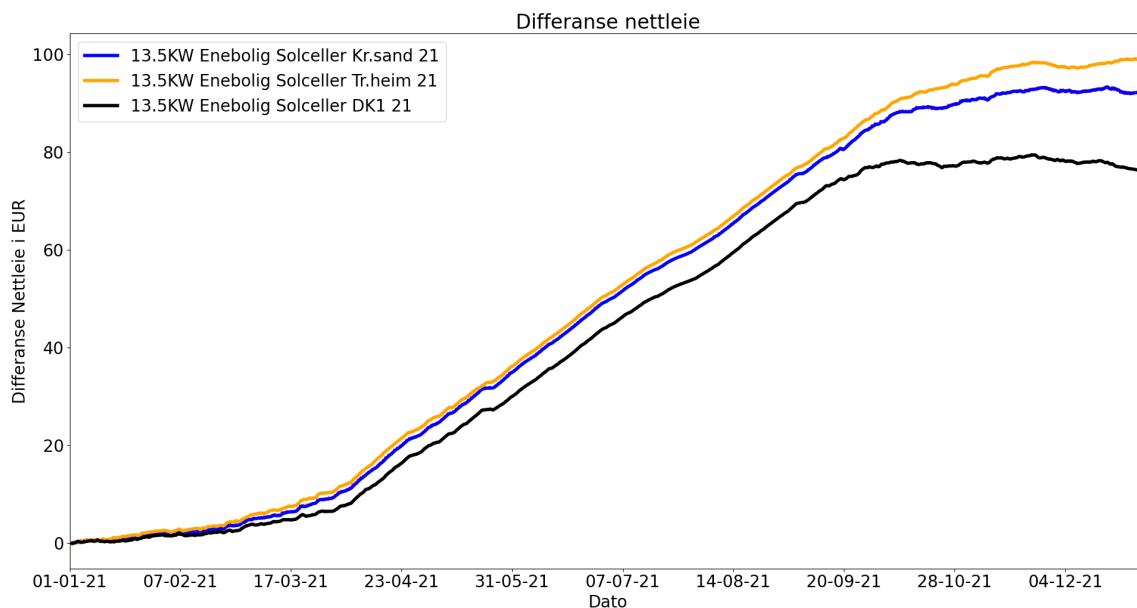
Ytelsen til **EBS** vurderes i forskjellige prissoner året 2021. Sone Kristiansand (NO2) i Sør-Norge har vært en av de dyreste sonene da utenlandskabler til England, Tyskland og Danmark er koblet på og har ført til at utenlandske strømpriser blir importert. Trondheim (NO3) er den sørligste sonen i Norge med relativt lav strømpris. Det skyldes store vindparker i Nord-Sverige og ikke nok overføringskapasitet mot de sørligere sonene til at prisen jevnes ut mot sørligere deler av landet. Jylland (DK1) ligger i Danmark som ikke har vannkraft til å stabilisere strømmettet og som er et netto importerende land. Dette viser hvordan strømmarkedet oppfører seg med høy andel ikke-styrbar produksjon.



Figur 63: Sammenlikning akkumulert besparelse Enebolig Kristiansand, Trondheim, DK1

For sone DK1 2021 er besparelsen på 16,46%, som er mer enn både Kristiansand 2021 og Kristiansand 2019. Fra det tidspunktet solproduksjonen tar seg opp i begynnelsen av april, stiger akkumulert besparelse jevnt opp til en total for året på 527 EUR. Hovedgrunnen for den

store besparelsen er høyere gjennomsnittspris på 88,1 EUR mot 75,1 EUR i Kristiansand (NO2). Det hender også at strømprisen går i negativt hvilket fører til at en får betalt for å lade opp batteriet, men når avgifter legges til blir i praksis strømmen fortsatt en kostnad. Danmark har i tillegg andre avgiftsnivåer som ikke blir tatt hensyn til. Her blir det sett på som at prisene for DK1 blir brukt i Norge.

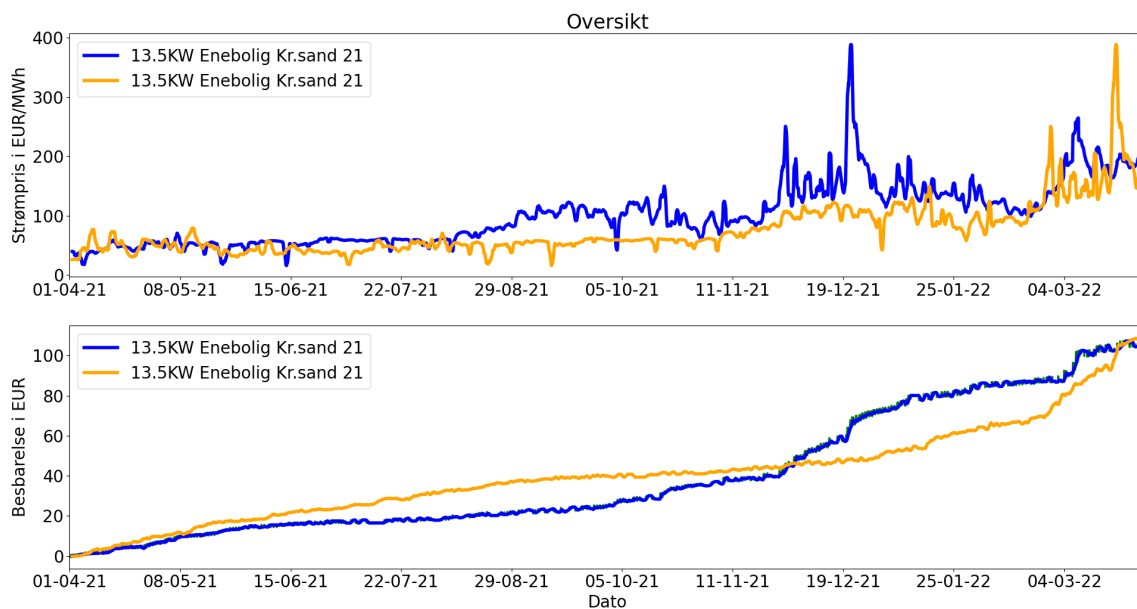


Figur 64: Akkumulert relativ besparelse på nettleie Kristiansand, Trondheim, DK1

Situasjon	Kristiansand 2021	Trondheim 2021	DK1 2021
Reduksjon strømpris	12,13%	13,64%	16,46%
Total besparelse	360 EUR	288 EUR	527 EUR
Besparelse nettleie	92,5 EUR	99,3 EUR	76,3 EUR
Oppladning på batteri fra nettet	2021 kWh	2115 kWh	2908 kWh
Effektivitet	97,6%	97,6%	97,2%

Siste året

Det siste året 2022 har strømprisene vært unormalt høye grunnet den ekstraordinære situasjonen. Dette fører til at resultatet ikke nødvendigvis blir representativ for fremtiden men for å illustrere hvordan et **EBS** handtere høye og volatile strømpriser testes data for 01-03-21 til 01-03-22.

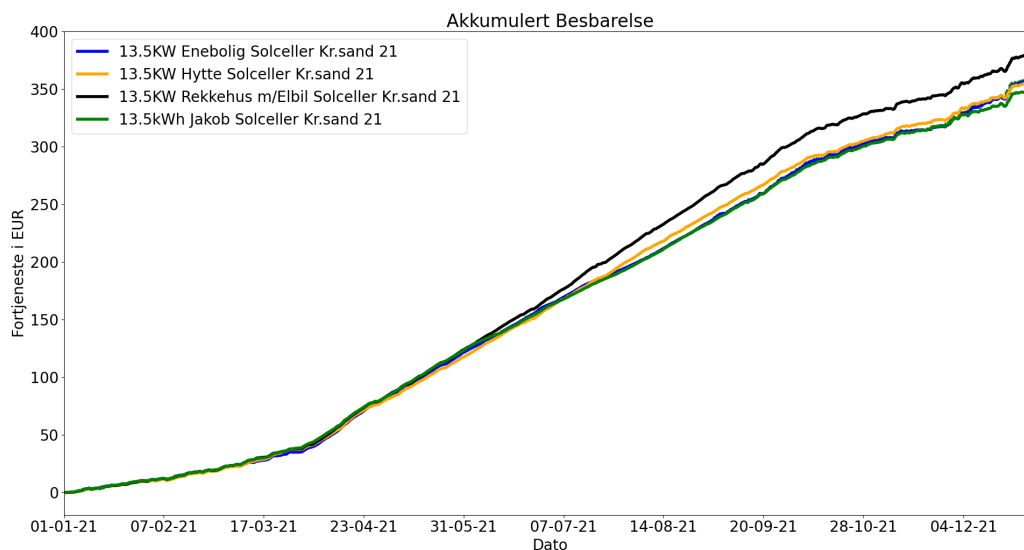


Figur 65: Sammenlikning akkumulert besparelse mellom 01-03-21 til 01-03-22 og 01-01-21 til 31-12-21

Strømpris har falt etter desember 2021 helt frem til starten av mars 22. Før den har steget noe igjen. Figuren 65 viser at endringene i strømmarkedet etter 2021 ikke har noen praktisk betydning for ytelsen til **EBS**.

Ulike forbruksmønstre

For å undersøke hvordan ulike forbruksmønstre påvirker ytelsen til EBS teste strømforbruket til fire ulike boliger. De fire er lokalisert i Stavanger området noe som betyr at dataen ikke er gjeldende for alle soner. Summen av forbruk gjennom hele året er også relativt likt med intervallet 15,3MWh-22,8MWh.



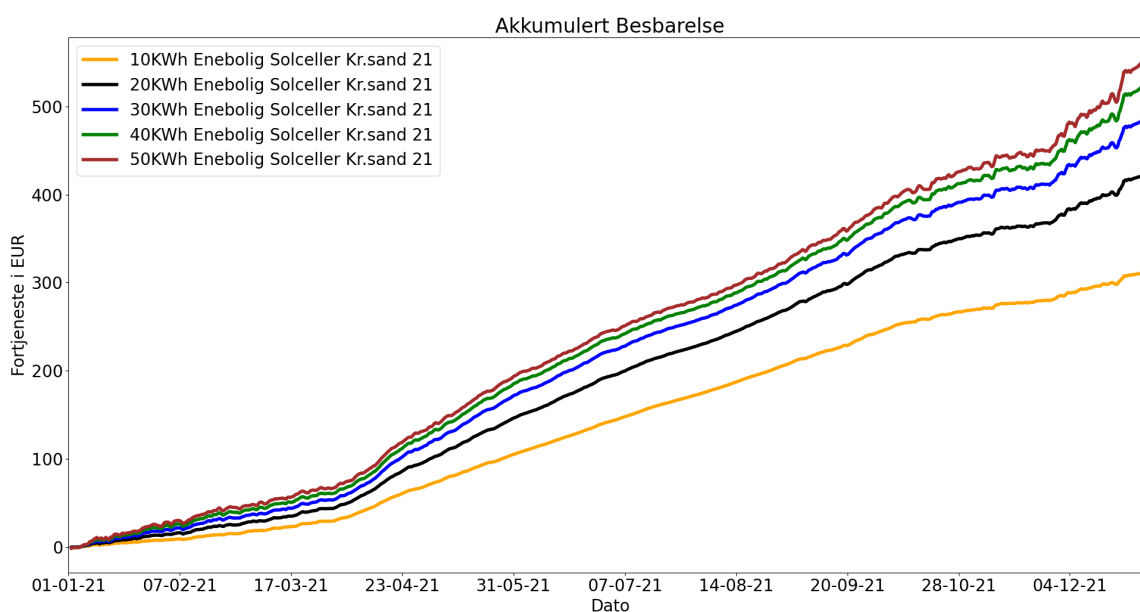
Figur 66: Sammenlikning akkumulert besparelse for ulike forbruksmønstre

Situasjon	Enebolig	Rekkehus m/elbil	Leilighet	Hytte
Strømforbruk	22,8 MWh	21,5 MWh	15,3 MWh	18,5 MWh
Standardavvik strømforbruk	1,7	1,4	1,17	1,29
Reduksjon strømpris	12,13%	14,94%	21,75%	17,63%
Total besparelse	360 EUR	383 EUR	351 EUR	358 EUR
Besparelse nettleie	92,5 EUR	101 EUR	102 EUR	99,3 EUR
Oppladning på batteri fra nettet	2021 kWh	1881 kWh	1594 kWh	1660 kWh
Effektivitet	97,6%	97,3%	97,0%	97,2%

Resultatet av testen viser at ytelsen til EBS blir i liten grad påvirket av ulike forbruksmønstre. Selv om reduksjon i strømpris i prosent endret seg betydelig fra 12,13%-21,75% avhenger det i større grad av hvor stort forbruket er og derav hvor høy strømprisen uten EBS er. Den totale besparelsen i EUR gjennom hele perioden er nesten lik for alle husstandene mellom 351,2-382,5 EUR. Untaket er rekkehuset med elbil der det kan tenkes at det uregelmessig høye strømtrekkene fra elbilladingen kan skape ekstra muligheter for å redusere strømkostnaden. Oppladning på batterier varierer med forbruk siden større forbruk betyr at en kan handle på lavere marginer og fortsatt spare mer vist ved at reduksjon i strømpris i % følger total oppladning på batteriet.

Ulike batteristørrelser

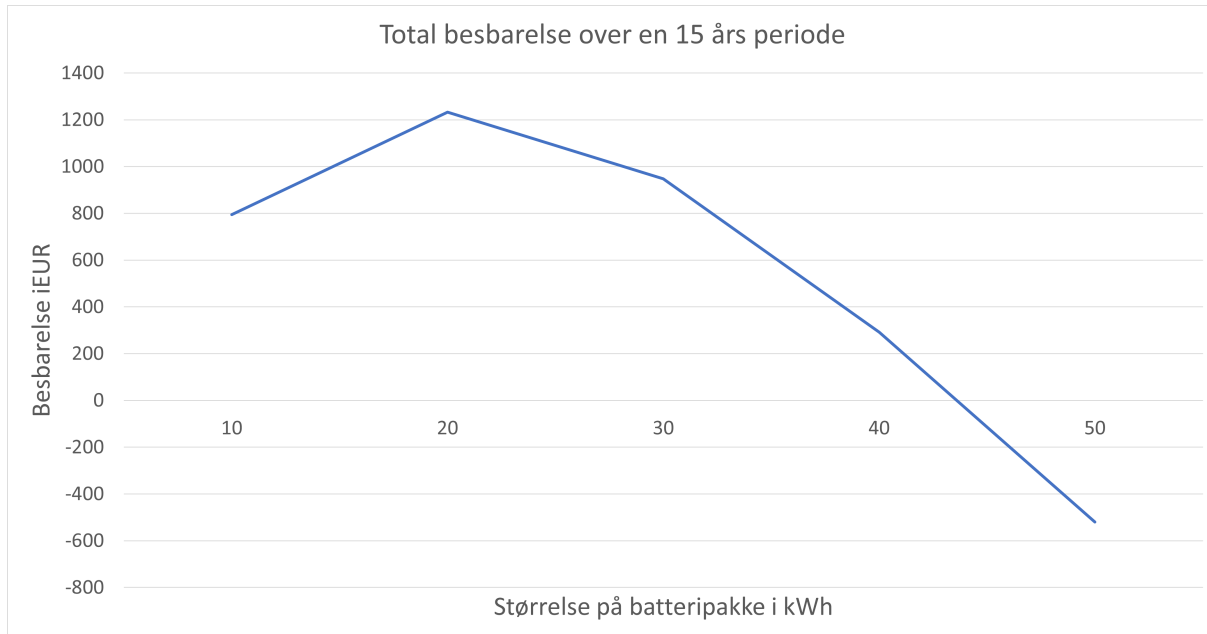
Ulike batteristørrelser teste for å undersøke hvordan ytelsen til systemet blir påvirket. Økt batteristørrelse betyr økte kostnader, som betyr at besparelsen over hele livsløpet må øke mer en kostnaden for batteripakken for at det skal lønnes seg å øke størrelsen på batteripakken. Andre fordeler med større batteripakke er at balansering kapasiteten for strømmettet samtidig stiger. Dog kan ikke private kunder idag ta del i frekvensreguleringsmarkedet ved å selge ledig kapasitet til netteier å få betalt for denne tjensten slik en kan i det kommersielle markedet. En potensiell løsning for at privatpersoner skal få tilgang til markedet er dersom kan melde inn ledig kapasitet til en kommersile aktøre som spesialeres seg på feltet og kan handle i markedet for alle kundene som har meldt inn ledig kapasitet og at deretter profitten blir delt mellom medlemmen i selskapet basert på hvor mye kapasitet som har blitt stilt til disposisjon.



Figur 67: Sammenlikning akkumulert besparelse 10-50 kWh batteripakke

Situasjon	10 kWh	20 kWh	30 kWh	40 kWh	50 kWh
Reduksjon strømpris	10,54%	14,32%	16,46%	17,78%	18,74%
Total besparelse	312 EUR	424 EUR	488 EUR	527 EUR	555 EUR
Besparelse nettleie	75,0 EUR	115 EUR	131 EUR	134 EUR	132 EUR
Oppladning på batteri fra nettet	1543 kWh	2804 kWh	3857 kWh	4715 kWh	5427 kWh
Effektivitet	97,9%	97,1%	96,6%	96,1%	95,8%

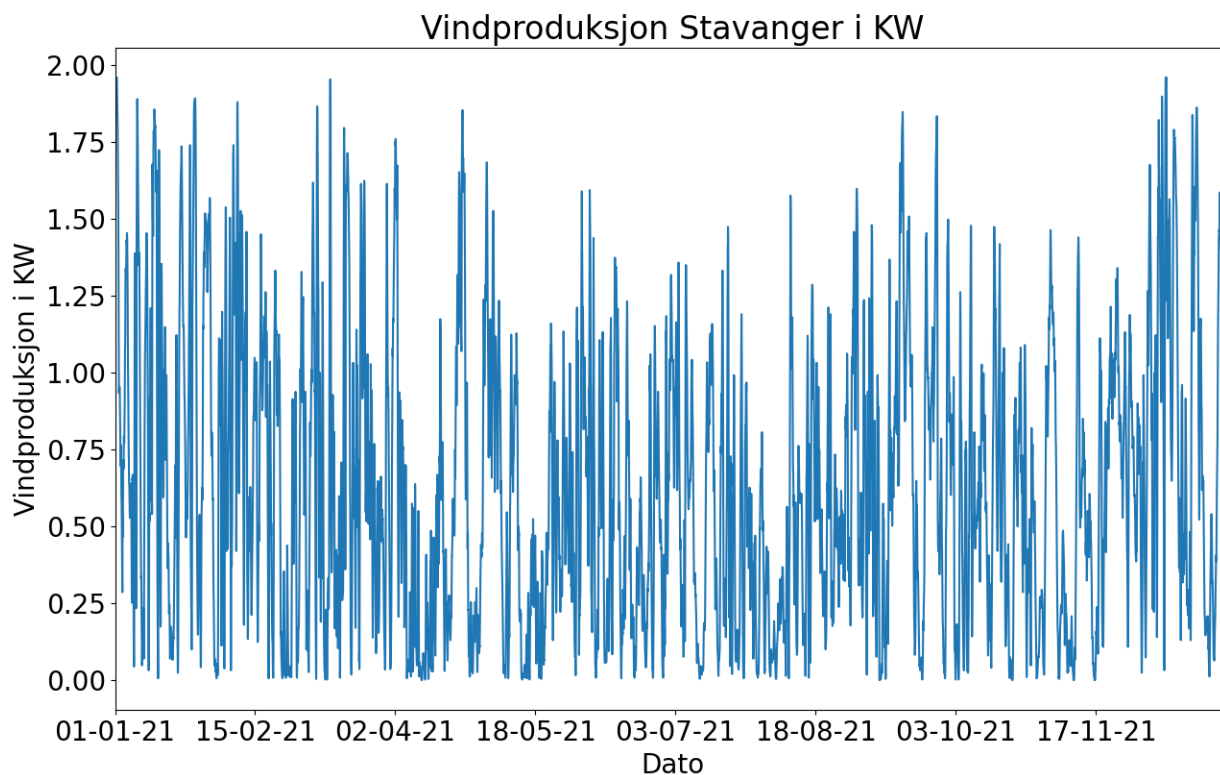
Økt kapasitet på batteripakken gir økt besparelse men forholdet er ikke proporsjonalt. Det betyr at dersom en tar hensyn til den økte kostnaden og måler over en 15års periode lønner det seg ikke å øke kapasiteten over 20 kWh vist i figur 68. Kostnaden som brukes er 124 EUR/kWh og 2650 EUR for inverter.



Figur 68: Sammenlikning mellom ulike batteristørrelser over 15 år

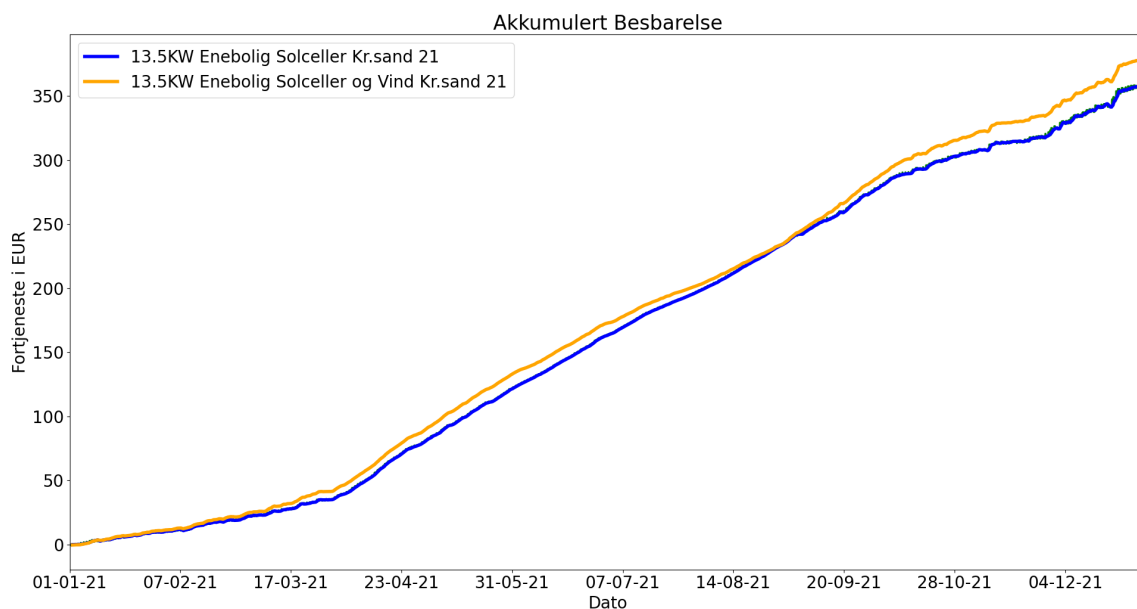
Med vind produksjon

Siden reduksjonen i strømpris er størst i perioden det er mye sol, kan det være aktuelt å installere **EBS** i forbindelse med andre energikilder som feks vindmøller. Det er typisk mer vindproduksjon om høsten og vinteren i Norge hvilket betyr at vind sammen med solceller kan dekke store deler av behovet for energi gjennom hele året når det er koblet opp mot et **EBS**.



Figur 69: Vindproduksjon i Stavanger 2019

Vind-data fra Stavanger i 2019 er hentet fra Renewables Ninja [34]. Det tas utgangspunkt i ei vindmølle på 2,5 kW, som er en størrelse som kan være aktuelt for private å installere på egen eiendom. Renewables Ninja har ikke så små vindturbiner som standard. Vindturbinen *Vestas V47 660* med en høyde på 30 meter ble valgt mens maks produksjon ble satt til 2kW noe som kan føre til feil i forbindelse med at ulike vindmølleblad er satt opp til å produsere mest ved bestemte vindhastigheter. Dog er det ikke et problem for bruken av dataen i denne oppgaven.



Figur 70: Akkumulert besparelse for kun solceller mot solceller og vind produksjon

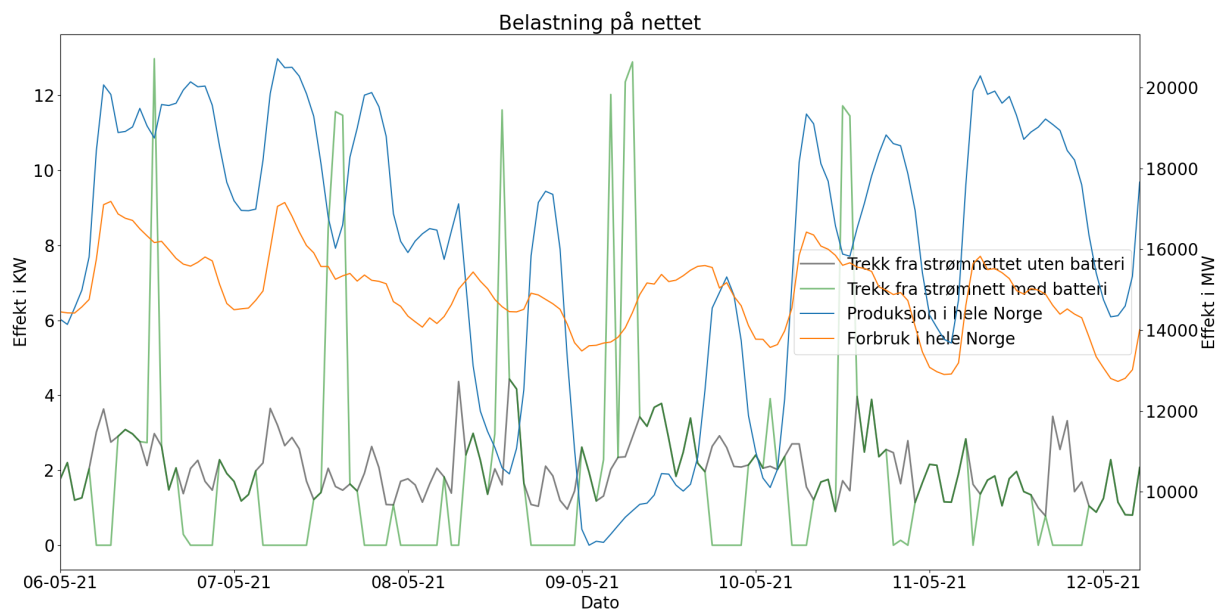
Situasjon	Solceller	Solceller og vind
Reduksjon strømpris	12,13%	17,29%
Kostnad uten batteri	2964 EUR	2202 EUR
Total besparelse	360 EUR	381 EUR
Besparelse nettleie	92,5 EUR	76,8 EUR
Oppladning på batteri fra nettet	2021 kWh	895 kWh
Effektivitet	97,6%	97,3%

Resultatet av å introdusere vindproduksjon sammen med solcelleproduksjon førte til en minimal økning i besparelse på 21,2 EUR. Dog økte reduksjon i strømpris betydelig som et resultat av at vindproduksjonen reduserte kostnaden av strømmen uten batteri. Det forklares ved at nevneren i brøken blir mindre og reduksjonen derav større. Det kommer av at vindproduksjon har en maks effekt på 2kW, som sjeldent er større enn forbruket i boligen. Det fører til direkte reduksjon i strømbruk fra strømmettet uten **EBS**. Oppladning fra nettet til batterier har blitt mer enn halvert som følge av at vindproduksjonen er mye mer stabil gjennom hele året og ikke har den samme døgnvariasjonen som solceller. Derav reduseres behovet for nettstrøm. Som en konklusjon er effekten av batteripakke sammen med vindproduksjon mye mindre enn effekten av solceller sammen med batteripakke, da vindproduksjon ikke har samme døgnvariasjon som solceller.

8.2 Sanntidskjøring

For å få bedre resultat med sanntidskjøring må værmeldingen brukes til å forutsi produksjonen fra solcellene. Værmeldingen alene vil ikke bidra til et perfekt resultat, men en kommer nærmere sanntid enn om man antar at neste dags produksjon blir lik snittet av de 3 forrige dagene. Da antar enten systemet at det skal komme mer produksjon fra solcellen enn det som er planlagt. En får da ikke plass til alt på batteriet på en optimal måte og ender opp med å selge strøm på ugunstige tidspunkt. Dersom produksjonen blir mindre enn forventet så har en ikke ladet opp batteriet tilstrekkelig for at det skal klare holde boligen selvforsynt med egenprodusert strøm gjennom hele døgnet i tidsrommet. Verdata kan også brukes til å kalibrer forventet strømforbruk ved at en hensyntar høyere strømforbruk til oppvarming når temperatur synker.

8.3 Systemets innvirkning på strømmettet



Figur 71: Strømtrekk fra nett sammenliknet med nasjonalt forbruk og produksjon

På figuren 71 er strømtrekk fra nettet med og uten batteri sammenliknet med nasjonalt forbruk og produksjon av strøm i Norge for 06-05-2021 til 12-05-21. Trekk fra strømmettet uten batteri følger ikke svingningen til det nasjonale forbruket. Ulikheter forårsaket av temperaturforskjeller innad i landet kan ha ført til at forbruksmønstret for boligen testet ikke følger flertallet av boliger i Norge. Samtidig er også forbruk fra industrien med som ikke følger temperatur eller har en typisk døgnsyklus som boliger.

Trekk fra strømmettet med batteri beveger seg motsyklisk med at batteriet lades ut når det

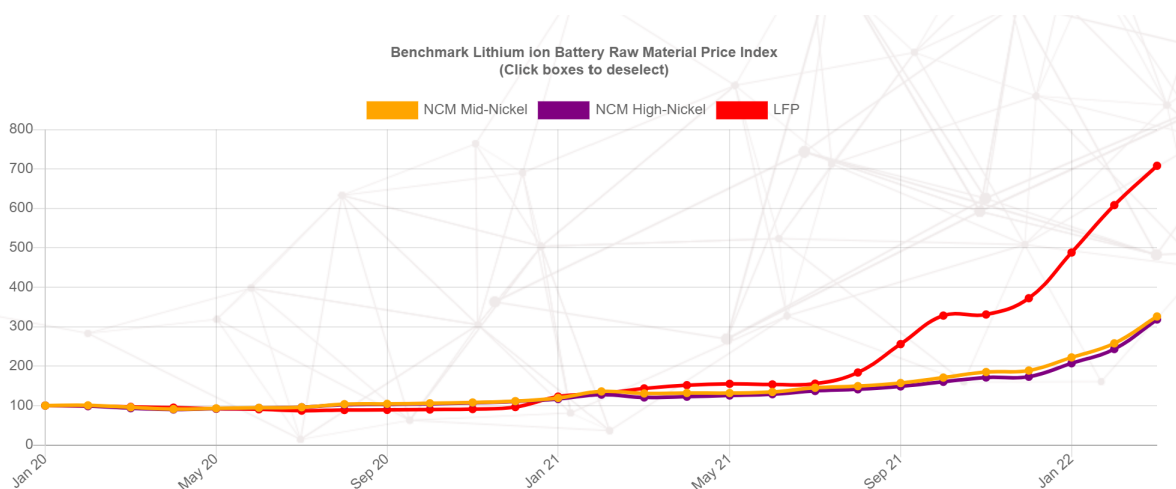
er en produksjontopp mens batteriet lades opp når det er en produksjonbunn. For eksempel mellom 06-05-21 og 07-05-21 lades batteri opp akkurat i det det er en dypp i produksjonen. Fallet i produksjon for den timen kan være forårsaket av ulike hendelser. Det er tydelig at det ble mindre behov for energi akkurat i det punktet, hvilket også medfører en prisubalanse dersom produksjonen ikke blir redusert tilstrekkelig, for eksempel fordi mange kraftverk ikke stenger ned, selv om de får litt mindre betalt i en time, da kostanden av å stenge ned anlegget er større enn tapet av å produsere. Rett etter oppladning så går produksjonen og til en topp før den begynner å falle. I denne perioden der produksjonen er høyest brukes strøm fra batteriet til å forsyne boligen, og strømtrekket fra nettet blir 0KW. Det at produksjonen er veldig høy over en kort periode betyr normalt høye strømpriser fordi når *Nord Pool* fastsetter prisen så får hele tiden de som er villige til å produsere strømmen billigst lov til å produsere. Når de billige produsentene ikke har mer kapasitet igjen må produsenter som skal ha mer betalt for strømmen starte produksjonen hvilket fører til høyere priser for alle.

Det er tydelig sammenheng mellom produksjon i Norge og trekk fra strømnnett med batteri, fordi strømbørsen *Nord Pool* som fastsetter når strøm skal produsere og til hvilken pris løser optimaliseringsproblemet med lineær programmering der målet er at alle som ønsker strøm får det til lavest mulig pris.

Det er ikke like stor direkte sammenheng mellom forbruk i Norge og trekk fra strømnnett med batteri, fordi det i denne perioden er eksport av strøm til utlandet. Produksjonsgrafene ligger store deler av tiden over forbruksgrafene, hvilket betyr at den norske forbruksgrafene ikke gir det fulle bilde på hva som setter strømprisen i denne perioden. Derav anses produksjon i Norge som den beste faktoren for å bestemme om **EBS** vil redusere belastningen på strømnettet. Produksjonen har som mål å til en hver tid akkurat dekke forbruk siden strøm er ferskvare og brukes idet den produseres.

8.4 Framtidsutsikter

Ønsket om et sammensatt, internasjonalt strømmnett gjennom avtaler som ACER o.l kan bety en fremtid hvor land er avhengige av hverandre i større grad enn før, og kraftproduksjon utenlands kan påvirke nasjonale priser. Innføringen av ikke-fleksible kraftkilder gjør nemlig forsyningen og dermed prisen lite forutsigbar. Uforutsigbarhet kan tenkes å øke i takt med nyinnføringer og endrede kraftavtaler. Dermed kan behovet for **EBS** som utviklet i denne oppgaven drastisk øke med årene, og før det blir en nødvendighet for privatpersoner, kan det tenkes at behovet for korttidslagring av energi i forbindelse med vind- og solkraft vil bli nødvendig gitt utbygging.



Figur 72: Graf som viser prisøkning på litium-celler, hentet fra [33]

Et global råvaremangel som startet i 2021 har ført til at prisen på råvarene brukt i batterier har steget betydelig. Prisene på råvarene brukt i nikkell, mangan og kobolt **NCM** celler har steget med over 300% siden starten av 2020, og slår direkte ut på at lønnsomhet til **EBS** blir redusert. Hvilket er negativt i og med at lønnsomheten i **EBS** var under stort press før prisøkningen på batteripakker. Prisøkningen fører til at færre prosjekter kan gjennomføres og gjør det vanskelig for et lønnsomt privat marked innenfor batterilagring med litium teknologi. Det kan på sikt dersom prisene stabilisere seg høyere føre til at andre batterityper vokser frem og kan ta over for litium innenfor batterilagring.

8.5 Potensielle feilkilder

For å kunne lage en modell for å simulere et **EBS** har det blitt gjort forenklinger som kan føre til feil resultat. Modellen har ikke blitt testet på et fysisk system, som betyr at modellen ikke er blitt verifisert direkte. Dog er det blitt brukt data fra andres forskning som grunnlag for modellen. I tillegg er faktisk data for strømforbruk, strømpris og solcellerproduksjon hentet inn.

Degradering av batteri blir ikke tatt korrekt hensyn til siden det kun simuleres over ett år om gangen. Modellene reduserer kapasiteten på batteriet avhengig av antall ladesykluser. Når det kun kjøres over et år kommer ikke effekten av degraderingen tydelig til syne, og degraderingen blir nullstilt hver gang modellen kjøres på ny som betyr at degraderingen i praksis ikke har blitt tatt hensyn til i modellen.

Batteriets levetid er essensiell i utregningen av lønnsomhet og ikke minst med tanke på klimaavtrykket. Felles for mye av tidligere forskning på levetid og antall mulige ladesykluser er at de er basert på relativt få datasett. Siden modellen ikke har blitt testet fysisk over flere år er det knyttet usikkerhet til levetiden i praksis.

Modellene tillater at 20 kW strøm som ikke blir lagret eller brukt kan selges tilbake til nettet vist i figur 46. Dette er en svakhet ettersom i noen tilfeller kan trekket fra batteriet i modellen overstige maksimal effekt begrenset av inverteren på 10 kW. Salg tilbake til nettet kan i tillegg begrenses av netteier med at nettet hele tiden må kunne ta imot maksimal effekt fra den enkelte bolig uten at det overbelaster strømmettet.

Under diskusjonsdelen har ulike forbruk, strømpriser og energiproduksjoner blitt sammenliknet på tvers av år og geografiske soner, hvilket kan skape problemer fordi de ulike parametrene avhenger av og påvirker hverandre.

Inverterkarakteristikken har blitt forenklet til en trinnviss karakterstikk forklart i 3.4. Ulempene er at modellen blir mindre dynamisk og gi feil energitap gjennom oppladningsinverteren i enkelte tilfeller. På grunn av at modellen tilpasser oppladningen for å være på det mest effektive trinnet.

Tap av ladning på batteriet blir ikke tatt hensyn til. Effektiviteten reduseres av batteriets egne energitap, dog i liten grad siden strømmen lagres på batterier over kort tid.

Endring i nettleietakster vil gjøre at modellen gir feil resultat siden besparelsen på nettleien bidrar betydelig til reduksjonen i strømpris. Fastleddet har ikke blitt tatt hensyn til siden per 2021 er ikke fastleddet avhengig av et makstrekk som det er blitt foreslått. Det har dog vært diskusjon rundt hvordan den nye nettleiemodellen blir men den skal ha som mål at det skal lønne seg å jevne ut strømtrekket slik at belastningen på strømmettet blir mindre. Det betyr at mest sannsynlig kan **EBS** redusere nettleie mer med nye nettleiemodeller enn nåværende modell.

Som nevnt i kapittel 2.1 er ytelsen til MIP-løseren brukt i dataprogrammet avgjørende. Denne ytelsen avhenger til en viss grad av hva og hvordan løseren mates med objektet og begrensningene den tildeles, men også av avgjørelsene den tar under sin lokale kjøring. Her går den innom programfiler avhengig av problemene som oppstår i utregningen, hvorav noen er enkle heuristiske snarveier. Da MIP-løserne i synkende grad er avhengig av kvaliteten på *input*, ligger et stort forbedringspotensial - og med andre ord en teknisk begrensning - innenfor ytelsen til løserne tilgjengelig på markedet idag. De er ikke spesifikt utviklet for å minimere total kostnad av **EBS** eller liknende installasjoner.

8.6 Videre arbeid

- Videre utvikle for bedre sanntids resultat (ta med værmelding, AI)
- Ta hensyn til ny nettleiemodell som kommer iløpet av 2022
- Teste modellen på et fysisk system
- Bruke mer presis inverter karakterstikk
- Teste med andre batteri typer
- Modellere større kommersielle **EBS** med lagringskapasitet på over 1MW

9 Konklusjon:

Resultatet av undersøkelsen viser at det er mulig å redusere strømkostnaden med 2,9%-22% avhengig av forbruksmønster, strømprisene, batteripakkens størrelse og om **EBS** er koblet til egenprodusert energi. Den årlige besparelsen i EUR varierte fra 111 EUR - 527 EUR. Det er et tydelig skille i besparelse avhengig av om **EBS** er tilkoblet energikilde. Selve energikilden tenkes installert i forkant av **EBS**, og inngår ikke i utregningene for besparelse.

Som følge av at reduksjonen i strømpris er betydelig bedre med solceller, har **EBS** tilkoblet solceller fått hovedfokus. Den økonomiske oversikten 6 viser at total besparelse inkludert avgifter over en 15 års periode ble negativ for begge løsningene som er tilgjengelig i markedet idag med en Netto nåverdi (**NNV**) på henholdsvis -5843 EUR for *Tesla Powerwall* med 13,5 kWh kapasitet og -6298 EUR for *Evyon* sitt brukt elbilbatteri med kapasitet på 36 kWh. I tillegg ble markedspriser for batteripakker brukt som ga en **NNV** på 133 EUR for batteripakke på 13,55 kWh og -763 EUR for batteripakke på 36 kWh. Dette viser at kostnaden av batteriene er den viktigste faktoren for lønnsomheten til **EBS**

Resultatene tyder på at Jylland (DK1, med solceller), er prissonen som fører til størst reduksjon i strømpris på 16,5% tilsvarende 527 EUR. Minst reduksjon i strømpris er i prissonen Stavanger (NO2, med solceller) der reduksjonene var 12.1 % tilsvarende 360 EUR. Det fremkommer at prissonen med mest varierende prissingninger og høyest generelt prisnivå fører til størst prosentvis reduksjon i strømpris og besparelse.

Ulike forbruksmønster hadde lite påvirkning på resultat når det ble testet med 4 ulike forbruk med solceller i prissone Kristiansand(NO2) for 2021. Det skyldes delvis at solcelleproduksjon settes likt for de 4 ulike forbruksmønstrene. Større selvforbruk av strøm produsert av solcellepanelet bidrar mest til reduksjon i strømkostnad. Både det variable nettleieleddet reduseres og kjøp av strøm fra nettet reduseres.

Forbruksmønsteret til en studentleilighet førte til størst prosentvis reduksjon i strømpris. Høyest besparelse kom av forbruksmønsteret til rekkehus med elbil. Prosentvis reduksjon i strømpris kommer av batteripakkens størrelse i forhold til forbruket, og studentleiligheten slår ut på dette målet da forbruket er betydelig lavere enn andre forbrukssett. Det fremkommer at forbruksmønsteret med høyest variasjon i strømtrekk fører til størst besparelse.

Total besparelse til systemet øker ved økning av batterikapasitet. En batteripakke med kapasitet på 20 kWh vil være ideelt for en privat bolig med et tilnærmet gjennomsnittlig årsforbruk av strøm, og produksjon fra eget solcellepanel. Av resultatene fremkommer det at en overgang fra 10 kWh batterikapasitet til 20 kWh medfører en økning i besparelse på 112 EUR, sammenstilt mot en økning i besparelse på 25 EUR ved en overgang fra batterikapasitet 40 kWh til 50 kWh.

Systemets effektivitet avtar ved økning av batterikapasitet. En lav batterikapasitet fører til at

EBS initierer færre ladesykluser. Algoritmen velger tidspunkt for opp- og utlading, og med begrenset batterikapasitet prioriteres kvalitative ladesykluser ovenfor kvantitative.

Sanntidskjøring uten solceller ble nesten identisk som ideell **EBS**, med en differanse i besparelse på -0,05%. Kjøring med solceller resulterte i betydelig svekket ytelse i perioden mellom lav produksjon fra solcellene og sommerperioden der produksjonen er jevnt høy. Dette resulterte i en differanse på -4,2% mot ideell **EBS**. Selv om ytelsen er svakere enn ideell **EBS** vil balanseringseffekten på nettet være tilnærmet lik optimal fordi *day-ahead* prisen er fastsett på forhånd. Som betyr at under Sanntidskjøring treffes normalt de samme oppladning og utladningstidspunktetene.

References

- [1] Edd Grent (2022). *The Electric Revolution: EV Sales Doubled in 2021 and Are Surging This Year Too*. URL: <https://singularityhub.com/2022/03/15/electric-car-sales-doubled-last-year-and-will-just-keep-going-up/>. (accessed: 14.04.2022).
- [2] IEA (2022). *Electric cars fend off supply challenges to more than double global sales*. URL: <https://www.iea.org/commentaries/electric-cars-fend-off-supply-challenges-to-more-than-double-global-sales>. (accessed: 14.04.2022).
- [3] Ahmad Arabkoohsar. “Chapter One - Classification of energy storage systems”. In: *Mechanical Energy Storage Technologies*. Ed. by Ahmad Arabkoohsar. Academic Press, 2021, pp. 1–12. ISBN: 978-0-12-820023-0. DOI: <https://doi.org/10.1016/B978-0-12-820023-0.00001-8>. URL: <https://www.sciencedirect.com/science/article/pii/B9780128200230000018>.
- [4] Aurecon. “Hornsdale Power Reserve Year 2 Technical and Market Impact Case Study”. In: (2020). URL: <https://www.aurecongroup.com/markets/energy/hornsdale-power-reserve-impact-study/>.
- [5] Norges Bank. *PENGEPOLITISK RAPPORT MED VURDERING AV FINANSIELL STABILITET 1/2022*. URL: <https://www.norges-bank.no/aktuelt/nyheter-og-hendelser/Publikasjoner/Pengepolitisk-rapport-med-vurdering-av-finansiell-stabilitet/2022/ppr-12022/innhold/>. (accessed: 08.05.2022).
- [6] BloombergNEF. *Battery Pack Prices Fall to an Average of \$132/kWh, But Rising Commodity Prices Start to Bite*. URL: <https://about.bnef.com/blog/battery-pack-prices-fall-to-an-average-of-132-kwh-but-rising-commodity-prices-start-to-bite/>. (accessed: 25.02.2022).
- [7] Energy Charts. *Net installed electricity generation capacity in Germany*. URL: https://www.energy-charts.info/charts/installed_power/chart.htm?l=en&c=DE&year=-1&stacking=sorted. (accessed: 24.01.2022).
- [8] Jahedul Islam Chowdhury et al. “Techno-environmental analysis of battery storage for grid level energy services”. In: *Renewable and Sustainable Energy Reviews* 131 (2020), p. 110018.
- [9] University of Washington Clean Energy Institute. *Flow batteries*. URL: <https://www.cei.washington.edu/education/science-of-solar/flow-battery/>. (accessed: 25.01.2022).
- [10] C.M. Costa et al. “Recycling and environmental issues of lithium-ion batteries: Advances, challenges and opportunities”. In: *Energy Storage Materials* 37 (2021), pp. 433–465. ISSN: 2405-8297. DOI: <https://doi.org/10.1016/j.ensm.2021.02.032>. URL: <https://www.sciencedirect.com/science/article/pii/S2405829721000829>.
- [11] Qiang Dai et al. “Life Cycle Analysis of Lithium-Ion Batteries for Automotive Applications”. In: *Batteries* 5.2 (2019). ISSN: 2313-0105. DOI: 10.3390/batteries5020048. URL: <https://www.mdpi.com/2313-0105/5/2/48>.
- [12] SEAI EirGrid. *Electricity*. URL: <https://www.seai.ie/data-and-insights/seai-statistics/monthly-energy-data/electricity/>. (accessed: 11.04.2022).
- [13] Jan Emblemsvåg. “Wind energy is not sustainable when balanced by fossil energy”. In: *Applied Energy* 305 (2022), p. 117748.

- [14] Energizer. *Nickel Metal Hydride (NiMH) Handbook and Application Manual*. URL: https://data.energizer.com/pdfs/nickelmetalhydride_appman.pdf. (accessed: 26.02.2022).
- [15] Chair of Energy Storage Technology TUM School of Engineering and Design Technical University of Munich. *SimSES - Software for techno-economic simulation of stationary energy storage systems*. URL: <https://www.epe.ed.tum.de/en/ees/simses/>. (accessed: 06.04.2022).
- [16] ENS. *Annual and monthly statistics*. URL: <https://ens.dk/en/our-services/statistics-data-key-figures-and-energy-maps/annual-and-monthly-statistics>. (accessed: 28.03.2022).
- [17] Jørgen Sørgård Erdal. “Stochastic Optimisation of Battery System Operation Strategy under different Utility Tariff Structures”. In: *IEEE Transactions on Power Electronics* 1 (2017), p. 111. DOI: <http://hdl.handle.net/11250/2456367>.
- [18] Evyon. *Evyon*. URL: <https://www.evyon.com/concepts-and-prototypes/>. (accessed: 06.04.2022).
- [19] Richard Folkson. *Alternative fuels and advanced vehicle technologies for improved environmental performance: towards zero carbon transportation*. Elsevier, 2014.
- [20] Hallgeir Vågenes Frank HaugsbøGøran Bohlin. *På innsiden av kraftbørsen*. URL: <https://www.vg.no/nyheter/innenriks/i/reJOPm/paa-innsiden-av-kraftboersen>. (accessed: 18.04.2022).
- [21] Deqiang Gan et al. *Electricity markets and power system economics*. CRC Press, 2014.
- [22] Jarand Hole og Hallgeir Horne. *Batterier vil bli en del av kraftsystemet*. URL: http://publikasjoner.nve.no/faktaark/2019/faktaark2019_14.pdf. (accessed: 11.04.2022).
- [23] Fremtiden i våre hender. *Faktaark om klimagassutslipp fra forbruk*. URL: <https://www.framtiden.no/dokarkiv/faktaark/147-klimagassutslipp-fra-forbruk/file.html>. (accessed: 08.05.2022).
- [24] Lars Olav; Rosvold Hofstad Knut; Askheim. *Knut A.: kraftutveksling med utlandet i Store norske leksikon på snl.no*. URL: https://snl.no/kraftutveksling_med_utlandet. (accessed: 26.01.2022).
- [25] IFBF. *What is a flow battery?* URL: <https://flowbatteryforum.com/what-is-a-flow-battery/>. (accessed: 25.01.2022).
- [26] EMD International. *energyPRO*. URL: <https://www.emd-international.com/energypro/>. (accessed: 06.04.2022).
- [27] EMD International. *The MILP Solver optimization method in energyPRO*. URL: https://www.emd-international.com/files/energypro/HowToGuides/HowToGuide_MILP%5C%20solver.pdf. (accessed: 06.04.2022).
- [28] California ISO. *What the duck curve tells us about managing a green grid*. URL: https://www.caiso.com/documents/flexibleresourceshelprenewables_fastfacts.pdf. (accessed: 01.04.2022).
- [29] Andrea Lodi. “The Heuristic (Dark) Side of MIP Solvers”. In: *Hybrid Metaheuristics*. Ed. by El-Ghazali Talbi. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 273–284. ISBN: 978-3-642-30671-6. DOI: 10.1007/978-3-642-30671-6_10. URL: https://doi.org/10.1007/978-3-642-30671-6_10.
- [30] Adriana C. Luna et al. “Mixed-Integer-Linear-Programming-Based Energy Management System for Hybrid PV-Wind-Battery Microgrids: Modeling, Design, and Experimental

- Verification”. In: *IEEE Transactions on Power Electronics* 32.4 (2017), pp. 2769–2783. DOI: 10.1109/TPEL.2016.2581021.
- [31] Arumugam Manthiram. “An outlook on lithium ion battery technology”. In: *ACS central science* 3.10 (2017), pp. 1063–1069.
- [32] J.B. MAVERICK. *What Is the Average Annual Return for the SP 500?* URL: <https://www.investopedia.com/ask/answers/042415/what-average-annual-return-sp-500.asp>. (accessed: 08.05.2022).
- [33] Benchmark Minerals. *LITHIUM ION BATTERY RAW MATERIAL PRICE INDEX*. URL: <https://www.benchmarkminerals.com/lithium-ion-battery-raw-material-index/>. (accessed: 18.04.2022).
- [34] Renewables ninja. *Renewables ninja*. URL: <https://www.renewables.ninja/>. (accessed: 28.03.2022).
- [35] NRK. *Enige om forslag til ny nettleie*. URL: <https://www.nrk.no/norge/enige-om-forslag-til-ny-nettleie-1.15893096>. (accessed: 30.03.2022).
- [36] Linda Ørstavik Öberg. *Slik blir din nye nettleie i 2022*. URL: https://www.huseierne.no/nyheter/slik-blir-den-nye-nettleien/?gclid=Cj0KCQjw_4SBhCgARIsAAlegrUHjVh16XX6VphBh4S821wcB. (accessed: 30.03.2022).
- [37] Nord Pool. *Historical market data*. URL: <https://www.nordpoolgroup.com/historical-market-data/>. (accessed: 28.03.2022).
- [38] Nord Pool. *Single hourly order*. URL: <https://www.nordpoolgroup.com/trading/Day-ahead-trading/Order-types/Hourly-bid/>. (accessed: 25.01.2022).
- [39] American Clean Power. *Vanadium Redox (VRB) Flow Batteries*. URL: <https://energystorage.org/why-energy-storage/technologies/solid-electrode-batteries/>. (accessed: 25.01.2022).
- [40] QuantumScape. *Solid-State Battery Landscape*. URL: <https://www.quantumscape.com/blog/solid-state-battery-landscape/>. (accessed: 12.01.2022).
- [41] Regjeringen. *Klimaendringer og norsk klimapolitikk*. URL: <https://www.regjeringen.no/no/tema/klima-og-miljo/innsiktsartikler-klima-miljo/klimaendringer-og-norsk-klimapolitikk/id2636812/>. (accessed: 11.04.2022).
- [42] Repack. *RePack Home Pro - Energy to fuel all your needs*. URL: <https://www.repack.no/home-pro/>. (accessed: 26.03.2022).
- [43] Shripad T. Revankar. “Chapter Six - Chemical Energy Storage”. In: *Storage and Hybridization of Nuclear Energy*. Ed. by Hitesh Bindra and Shripad Revankar. Academic Press, 2019, pp. 177–227. ISBN: 978-0-12-813975-2. DOI: <https://doi.org/10.1016/B978-0-12-813975-2.00006-5>. URL: <https://www.sciencedirect.com/science/article/pii/B9780128139752000065>.
- [44] Statnett. “Statnetts Kortsiktige Markedsanalyse 2021-2026”. In: (2021). URL: <https://www.statnett.no/for-aktorer-i-kraftbransjen/planer-og-analyser/kortsiktig-markedsanalyse/>.
- [45] Statnett. *Tall og data fra kraftsystemet*. URL: <https://www.statnett.no/for-aktorer-i-kraftbransjen/tall-og-data-fra-kraftsystemet/#produksjon-og-forbruk>. (accessed: 24.01.2022).
- [46] European Association for Storage of Energy. *EASE_TD_{NiMH}*. URL: https://ease-storage.eu/wp-content/uploads/2016/03/EASE_TD_NiMH.pdf. (accessed: 24.01.2022).

- [47] Tesla. *Powerwall*. URL: <https://www.tesla.com/powerwall>. (accessed: 26.03.2022).
- [48] Tibber. *Et uvanlig strømselskap*. URL: <https://tibber.com/no>. (accessed: 30.03.2022).
- [49] Tibber. *Solproduksjon med Tibber*. URL: <https://support.tibber.com/nb/articles/4669873-solproduksjon-med-tibber>. (accessed: 30.03.2022).
- [50] Cong Nam Truong et al. “Economics of residential photovoltaic battery systems in Germany: The case of Tesla’s Powerwall”. In: *Batteries* 2.2 (2016), p. 14.
- [51] Stathis Tselepis. “12 years operation of the Gaidouromantra Microgrid in Kythnos island, COI-3869”. In: Dec. 2012.
- [52] Battery University. *BU-409: Charging Lithium-ion*. URL: <https://batteryuniversity.com/article/bu-409-charging-lithium-ion>. (accessed: 26.02.2022).
- [53] Battery University. *BU-808: How to Prolong Lithium-based Batteries*. URL: <https://batteryuniversity.com/article/bu-808-how-to-prolong-lithium-based-batteries>. (accessed: 26.02.2022).
- [54] Siqin Xiong, Junping Ji, and Xiaoming Ma. “Environmental and economic evaluation of remanufacturing lithium-ion batteries from electric vehicles”. In: *Waste Management* 102 (2020), pp. 579–586. ISSN: 0956-053X. DOI: <https://doi.org/10.1016/j.wasman.2019.11.013>. URL: <https://www.sciencedirect.com/science/article/pii/S0956053X19307081>.
- [55] Chengjian Xu et al. *Future material demand for automotive lithium-based batteries - communications materials*. Dec. 2020. URL: <https://doi.org/10.1038/s43246-020-00095-x>.
- [56] T.-K. Ying et al. “Studies on rechargeable NiMH batteries”. In: *International Journal of Hydrogen Energy* 31.4 (2006), pp. 525–530. ISSN: 0360-3199. DOI: <https://doi.org/10.1016/j.ijhydene.2005.04.018>. URL: <https://www.sciencedirect.com/science/article/pii/S0360319905001205>.
- [57] Akira Yoshino. “The birth of the lithium-ion battery”. In: *Angewandte Chemie International Edition* 51.24 (2012), pp. 5798–5800.
- [58] Anqi Zeng et al. *Battery Technology and recycling alone will not save the electric mobility transition from future cobalt shortages*. Mar. 2022. URL: <https://www.nature.com/articles/s41467-022-29022-z>.

10 Vedlegg

10.1 Opprinnelig oppgavebeskrivelse

Forslag oppgavetittel:

Kortidslagring av strøm på batteri

Kraftregulering til batteri for stabilisering av strømnnett og strømpriser.

Hva det gjelder:

Strømnettet avhenger i økende grad av fornybar energi. Strømmerket globaliseres, og Norge eksporterer egenprodusert strøm utover Europa. Uforutsigbar kraftproduksjon og eksportavtaler legger press på landets kraftforsyning, og stiller nye krav til regulering og stabilisering av ekstremer på markedet så vel som på strømnettet. Behovet for lagring av energi, over kort og lang tidshorison blir således styrket, batterier kan fungere som et regulerende ledd, både økonomisk og teknisk.

Formålet er å undersøke hvorvidt det er fordelaktig med private energilagringssystemer for hver enkelt husholdning og i hvilken grad det kan være fordelaktig for strømnettet.

Problemstilling:

Verden skal bort fra fossilenergi og over på fornybar energi. Denne overgangen skaper både muligheter og potensielle problemer. Kraften som produseres fra fornybare kilder avhenger av været, i motsetning til kull, gass og olje hvor produksjonen kan justeres i forhold til behovet til en hver tid. Statnetts prognoser anslår at kraftbehovet i Norge kommer til å vokse mest av alle landene i Norden med en anslått vekst på 19TWh innen utgangen av 2026. Største bidragsyttere til veksten er elektrifisering av petroleumsindustrien, ny industri (bla. batterifabriker) og datasentre. Norges kraftoverskudd vil nærme seg null ved en forbruksvekst på 19TWh over en 5 års periode, og flere kraftverk må bygges ut.

Variabel produksjon i Norge forventes å øke fra 4GW til 16GW. Styrbar produksjon er uendret da de fleste store ufredete elver allerede er bygget ut. (Småkraftverk kan hjelpe?). Balansen mellom tilgjengelig, styrbar produksjon, og forbruk som ikke kobler ut på pris har blitt svekket i de siste 5-6 årene. Økende forbruk dekkes i økende grad av vindkraftproduksjon, og kan tyde på at balansen svekkes ytterligere. "Fleksibelt industriforbruk, batterier og smart ladning blir sentralt for å kunne møte

utfordringene", ifølge Statnett. Fornybar kraftproduksjon kan ikke styres, men avhenger av vær og vind. I områder og land med store lokale klimavariasjoner vil effektiviteten variere følgelig. Variasjonene i effekt bidrar til å øke strømprisenes volatilitet. Eksempelvis kan en sol- og vindfull sommerdag sende prisen ned i det negative, mot at prisen femdobles mot årlig gjennomsnitt en høytrykksperiode om vinteren.

Batteri som virkemiddel for å stabilisere strømnettet er kjernen av denne oppgaven. Kraftflyt skal reguleres inn og ut av et batteri. Hvordan kraftflyten reguleres bestemmes av en algoritme, som skal basere seg på batteriets indre egenskaper så vel som viktige eksterne parametre. Strømpris, vær, egenproduksjon av kraft og mitigering av frekvensforskjeller er noen av parametrene som tenkes regulert. Hvor stor påvirkning hver av faktorene har for stabilisering av strømnettet avhenger i stor grad av størrelsen på kraftreguleringssystemet. Funksjonen algoritmen skal utøve er å bruke mest mulig av egenprodusert kraft, og selge minst mulig tilbake. På den måten skal det forsøkes å oppnå en mer stabil og forutsigbar kraftflyt, samt mindre volatile priser. Andre fordeler kan være sikkerhet mot strømbrydd, da store batterianlegg kan gjenkjenne når systemet det er koblet til isoleres fra resten av nettet. Deretter kan anlegget handle for at nettet lokalt skal være stabilt i påvente av at systemet kobles på igjen. Frekvensregulering stabiliserer strømnettet på 50Hz ved hjelp av kraftregulering til batteri.

Økonomisk lønner det seg også, data fra det som var verdens største batterianlegg "Horndale Power Reserve" som har en utladningskapasitet på 100MW og en lagringskapasitet på 129MWh. Anlegget er lokalisert sør i Australia og er koblet til 275kV nettverket gjennom samme punkt som Hornsdale vindpark på 30MW. Evalueringsrapporten utredet av Aurecon hevder at iløpet av 2019 reduserte

batterilagringsanlegget kostanden på det nasjonale nettet med 119 millioner australske dollar. Av hvilket kommer 14 millioner dollar fra en 5 timers separasjon der batteriet var med på å stabilisere den sør australske delen som ble frakoblet resten av landet. Batteriet klarte å stabilisere nettet på 50.15 Hz etter en topp på 50.85 Hz etter separeringen iløpet av en periode på 16 minutter. Hornsdale prosjektet beviser at det er behov for batteri koblet til nettet både for å redusere kostnader i forbindelse med frekvens kontroll men også for å sikre at strømmettet er operativt selv om deler av nettet kobles fra hverandre i perioder.

Målet med oppgaven er å undersøke fordelene og lønnsomheten av å installere batteripakker i private husholdninger. Batteripakker kan utnyttes som enkeltstående system, eller med tilknyttet egenprodusert energi tilsvarende solcellepanel, vindmøller, geotermisk energi såvel som bioenergi.

Et enkeltstående system, og et system med én ekstern energikilde skal undersøkes. I begge tilfeller skal den mest hensiktsmessige batteristørrelsen vurderes ut ifra kravene:

- Reduskjon i strømpris(\%)
- Responstid
- Nedbetalinstid
- Utslipp

Metoden for å lage energibehandlingssystem samt undersøke kravene til batteripakken er å utvikle:

- Analyseverktøy
- Optimeringsalgoritme
- Reguleringsmetode

En foreslått algoritme med tilhørende programvare skal utvikles basert på tidligere forskning. Inngangsvariablene blir

- Kraftproduksjon
- Forbruk
- Tid på døgnet
- Tilstand på batteriet
- Variablene skal utnyttes med formål om å:
 - Stabilisere strømtrekk
 - Lagre overflødig energi
 - Spare økonomisk
 - Minske klimaavtrykk

Dette har en positiv samfunnseffekt ved at strømprisen blir jevnere for alle forbrukere selv om det ikke har batteri selv. Dersom det blir installert tilstrekkelig batterikapasitet blir prisen flattere fordi balansen mellom forbruk og produksjon korrigeres raskere, det gjør det også mulig for kraftverkene å produsere mer strøm når forbruket er lavt, fordi strømmen som da produseres kan lagres blant annet ved hjelp av batterier. Det er med å sikre at det er mindre behov for kraftverk som starter kun i perioder der forbruket er ekstremt høyt. Hvilket er positiv for miljøet fordi det ikke blir bygget unødvendig mange kraftverk for å dekke et maks strømtrekk som varer i så lite som en time.

10.2 Notater

Batteri

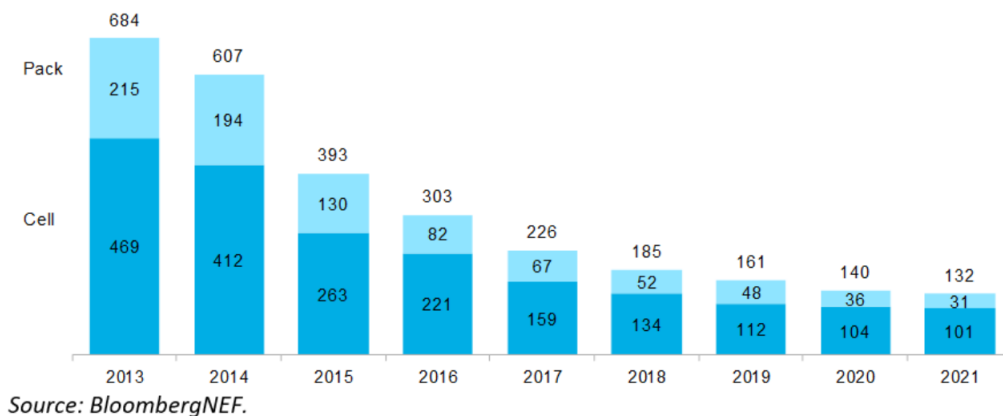
Batteriet er den viktigste komponenten i et **EBS**. Utviklingen skjer fort og det blir stadig utviklet nye batterityper. Det betyr at det er en risiko for at batteriteknologien blir utdatert før livsløpet til et batteri i bruk er ferdig. Valg av korrekt teknologi blir essensielt, og må basere seg på informasjon tilgjengelig i nåtid samt prognoser for fremtiden. Spesifikt i denne oppgaven er levetid, energitetthet, kontinuerlig kraft og kostnad/kWh viktige parametre.

Kilde til generell beskrivelse av de forskjellige batteriteknologiene.[3]

4 poenggivende egenskaper diskuteres HVorfor brukes hvorfor ikke

Litium-ion

Litium-ion er den meste brukte batteritypen, fordi kostnaden er relativt lav på en snittpris på 132 dollar per kWh for 2021. Diagram 73 viser at kostnaden for litium-ion celler stadig reduseres, mye på grunn av press fra bilprodusenter [6].

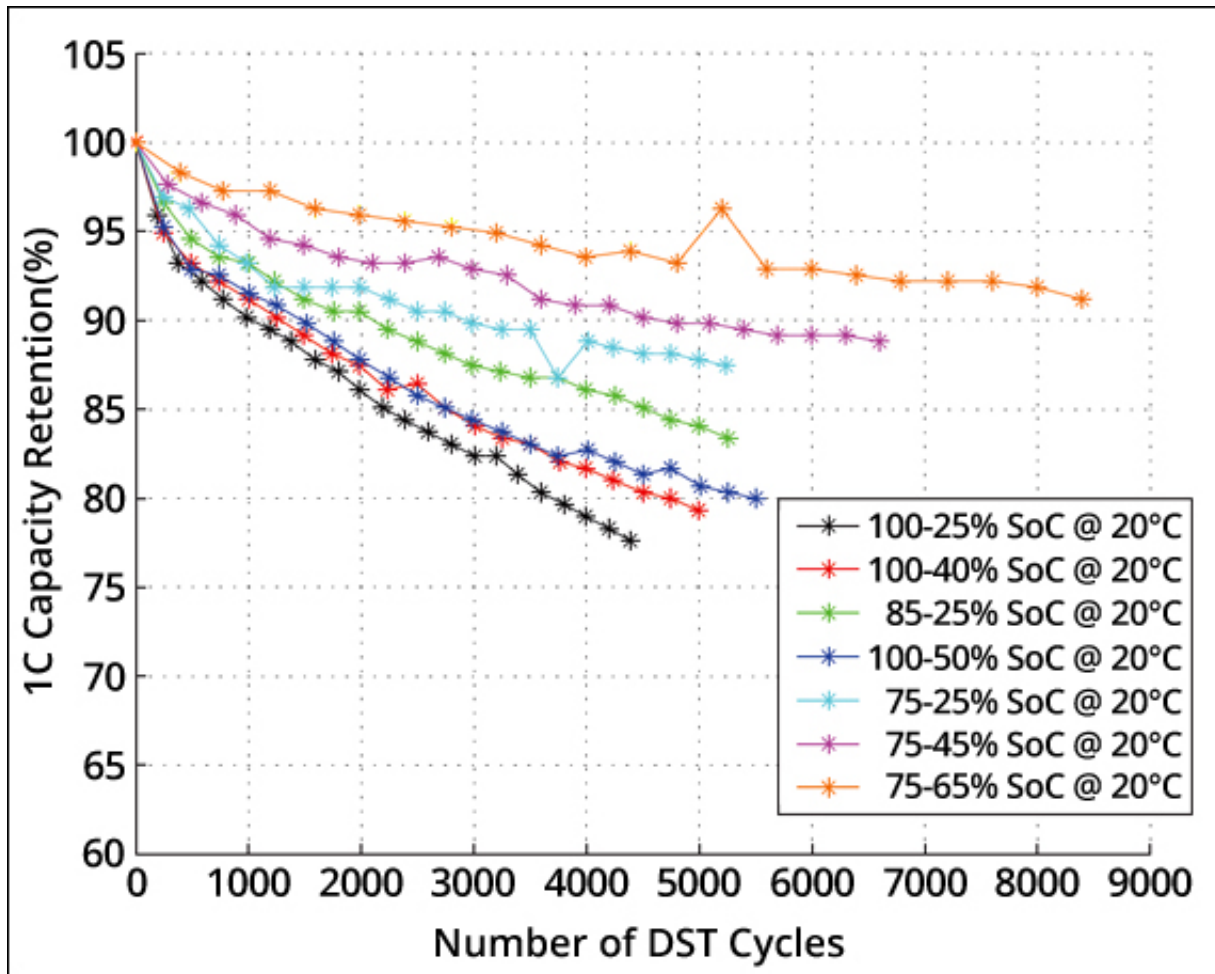


Figur 73: Diagram som viser utviklingen i pris for litium-ion celler

Litium-ion har idag en energi tetthet på mindre enn 250 Wh/kg og mindre enn < 650 Wh/l [31]. Energitettheten når det kommer til batterilagring koblet til strømmettet er ikke avhengig av at energitettheten økes med mindre det er begrenset areal der energilagringssystemet skal installeres. Derimot er automobil industrien avhengige av energitettheten fordi rekkevidden økes. Det samme gjelder små elektronikk som datamaskiner, mobiler, høyretelefoner osv, der brukstiden kan økes. For alle bruksområder er dog fordelene med øke energitetthet lavere forbruk av råvarer fordi mer energi er lagret på mindre fysisk plass.

Litium-ion celler har lang levetid dersom BMS systemet tar hensyn til at fulle ladesykluser

reduseres kapasiteten og levetiden på cellene.



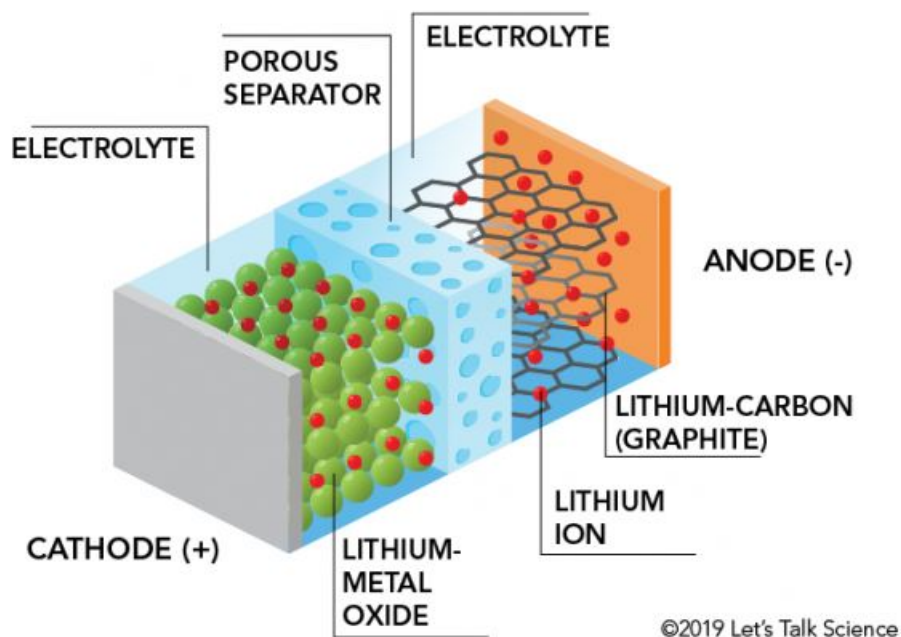
Figur 74: Graf som viser hvordan levetiden og kapasiteten reduseres avhengig av ulike lade og utladning områder, hentet fra [53]

For batterilagringssystemet er det ønskelig med langlevetid men også utnytte kapasiteten som er tilgjengelig på batteri for å kunne forskyve forbruk mot pris mest mulig. Tar derfor utgangspunkt i 85%-25% og 75%-25% grafene i figuren og setter levetiden til 5000 for å ha en margin som sikrer mot for lang estimert levetid ettersom de har stor påvirkning på lønnsomheten. Dog er grafene tilnærmet lineære, så for bruksområdet lagring av strøm i private hjem er ikke det at den opprinnelige kapasiteten er redusert til 80% noe problem. Som tyder på at batteri kan ha en levetid på over 10000 sykluser. Litium batterier har en anbefalt laderate på 1C men klarer opp mot 3C [52].

Litium-ion cellene består av en negativ elektrode som er laget av litium metal oksid, denner

delen blir typisk lagt opp et lag med kobber som vist i figur 75. Den positive elektoden som består av grafitt blir typisk lagt oppå aluminium. Elektrolyt materialet som er mest brukt er LiPF6 som består av litium, fosfat og fluor. Den forhindrer elektronene å flytte seg mellom negativ og positive katode [57].

PARTS OF A LITHIUM-ION BATTERY

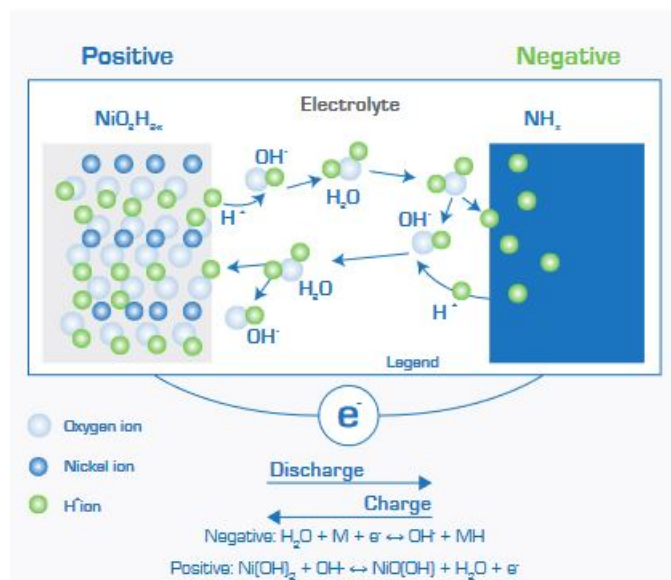


Figur 75: Detaljert grafisk fremvisning av en litium-ion celle

NiMH

Nikkel-metallhydrid (NiMH) celler er generelt brukt som oppladbar erstatning for tradisjonelle Nikkel-Kadmium (NiCd) celler. Med en litt lavere nominell spenning er også NiMH celler mindre utsatt for sprenging og lekkasje enn tradisjonelle Nikkel-Kadmium celler. Begge celletypene benytter seg av samme kjemiske reaksjon på den positive elektroden, som bruker Nikkel-Oksid-Hydroksid (NiOOH). Den negative elektroden i NiMH-celler bruker istedenfor Cadmium en hydrogenabsorberende legering.

Illustration: Charging principal of NiMH

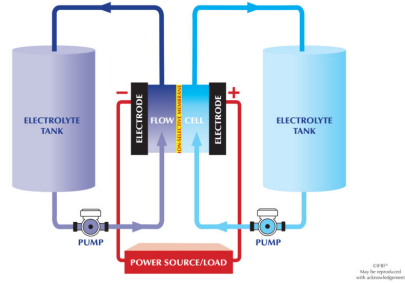


Figur 76: Grafisk fremstilling av kjemisk prosess i NiMH-celler. Hentet fra EASE[46]

NiMH-celler har en energitetthet på rundt 170-420Wh/l [43]. Ved å endre på den kjemiske sammensetningen i elektrodene, har man på typiske AA-celler nådd en spesifikk energitetthet på opp mot 75Wh/kg[56]. NiMH-celler har mindre "minneeffekt" og er mer miljøvennlige enn konvensjonelle NiCd-batteri. Sammenliknet med litium-ion tåler NiMH flere ladesykluser og har dermed lengre levetid, og under ideelle forhold kan NiMH batteier klare opp mot 1000 ladesykluser. NiMH har flat utladningskarakteristikk og en anbefalt oppladningsrate på C/3 til 1C [14]. NiMH cellene har en pris på 250-450 dollar per kWh [19]

Redox Flow-Batteri

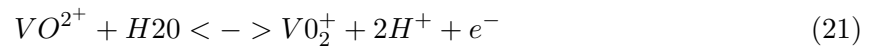
Et redox-flow-batteri (RFB) er et oppladbart batteri som vanligvis består av to tanker med elektrolytter. Løsningene flyter gjennom en eller flere elektrokjemiske celler med en ion-selektiv skillemembran. Ved oppladning skilles elektroner fra den uladete positive elektrolytten, og drar gjennom elektroden ut av den positive halvdelene av celledatabelen. Deretter når elektronene den negative elektroden, hvor elektronene hopper på den uladete negative elektrolytten[9]. Den kjemiske sammensetningen til begge elektrolyttløsningene endres, og batteriet ansees oppladet. Ved utladning reverseres de kjemiske reaksjonene, og batteriet går tilbake til opprinnelig tilstand.



Figur 77: Grafisk fremstilling av flow-batteri. Hentet fra IFBF[25].

Det finnes flere typer RFB, som skiller av hvilket elektroaktive redoks-par som utnyttes. De mest kjente er Sink/Brom, Jern/Krom og Vanadium/Vanadium. Vanadium Redox Flow-Batteri (VRB) utnytter Vanadiums fire ulike valenstilstander, og lagrer energi ved bruk av redoksparet V^{2+}/V^{3+} i den negative halvcellen og V^{4+}/V^{5+} i den positive halvcellen[39]. I følge **University of Washington** er VRB en lovende teknologi for energilagring på strømnetskala[9].

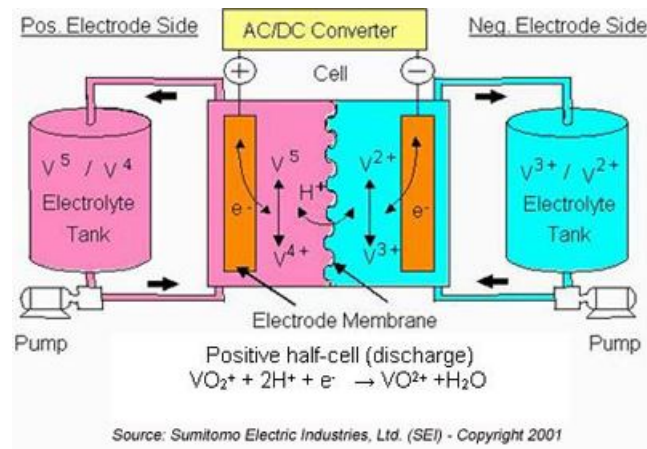
Den kjemiske reaksjonen for tar plass på katoden er:



Og den kjemiske reaksjonen på anoden:



Funksjonen av et VRB kan fremstilles grafisk:

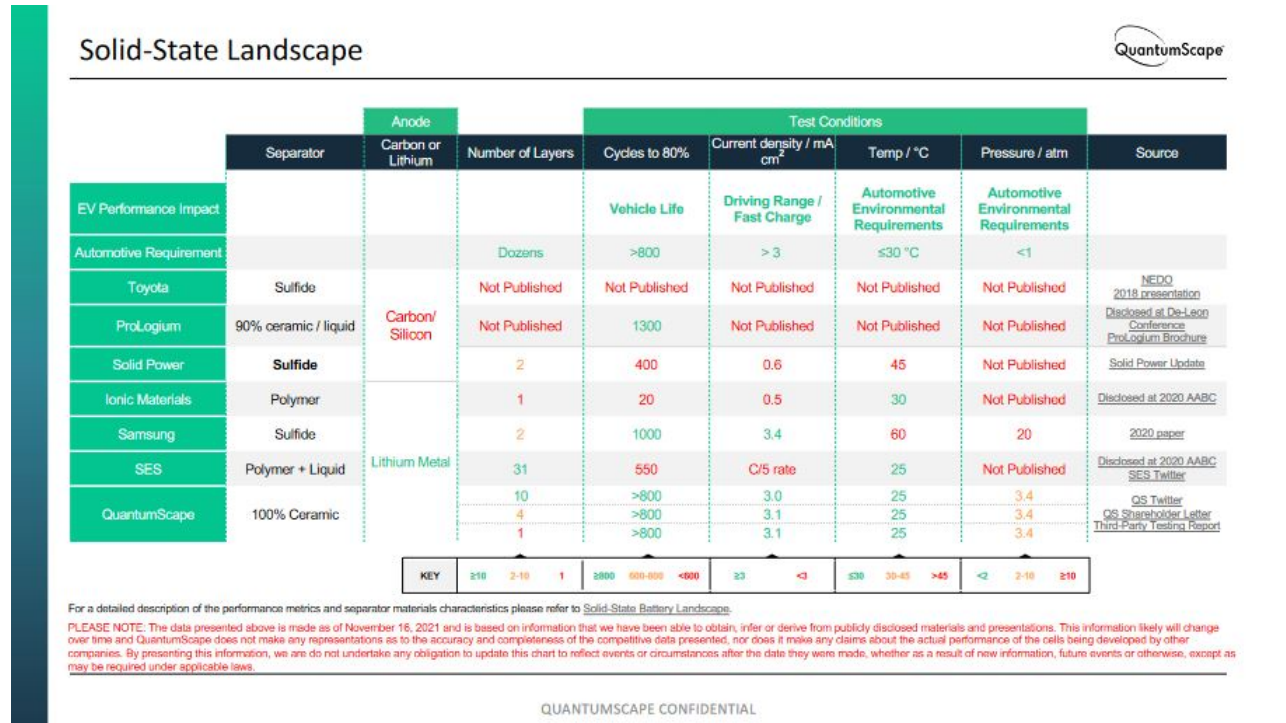


Figur 78: Grafisk fremstilling av VRB. Hentet fra ACP[39].

Universitetet i Washingtons *Clean Energy Institute (CEI)* har sammen med *UniEnergy Technology (UET)* utviklet et VRB på størrelse med en shipping-container, som har en effekt på 600kW og kapasitet på 2.2MWh[9].

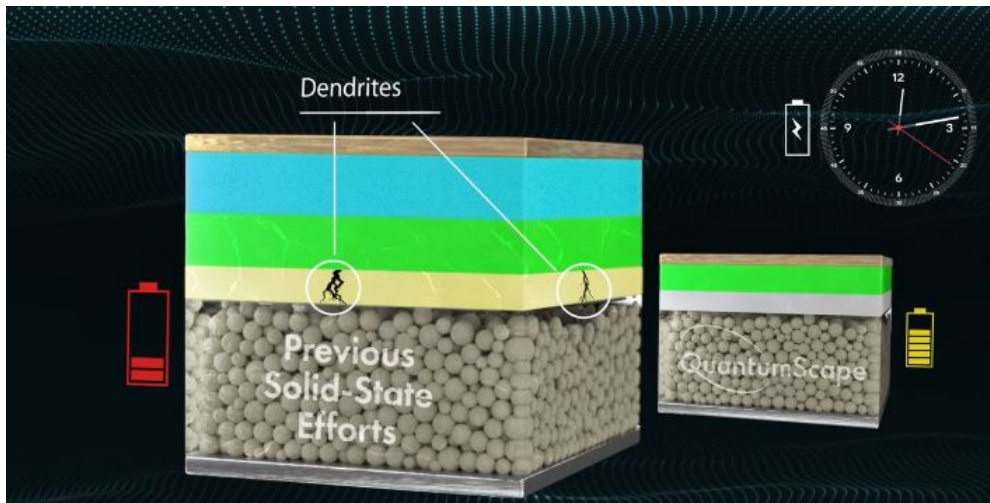
Solid state

Solid-State Batterier (SSB) er bygget opp av celler som består av en elektrisk kontakt på en katode, et separerende lag og en elektrisk kontakt direkte på separatorene. De forskjellige solid-state teknologiene i utvikling skilles blant annet på hva som brukes som separator.[40]



Figur 79: Sammenlikning av solid-state teknologier i elbil-landskapet per 16.11.2021, QuantumScape[40]

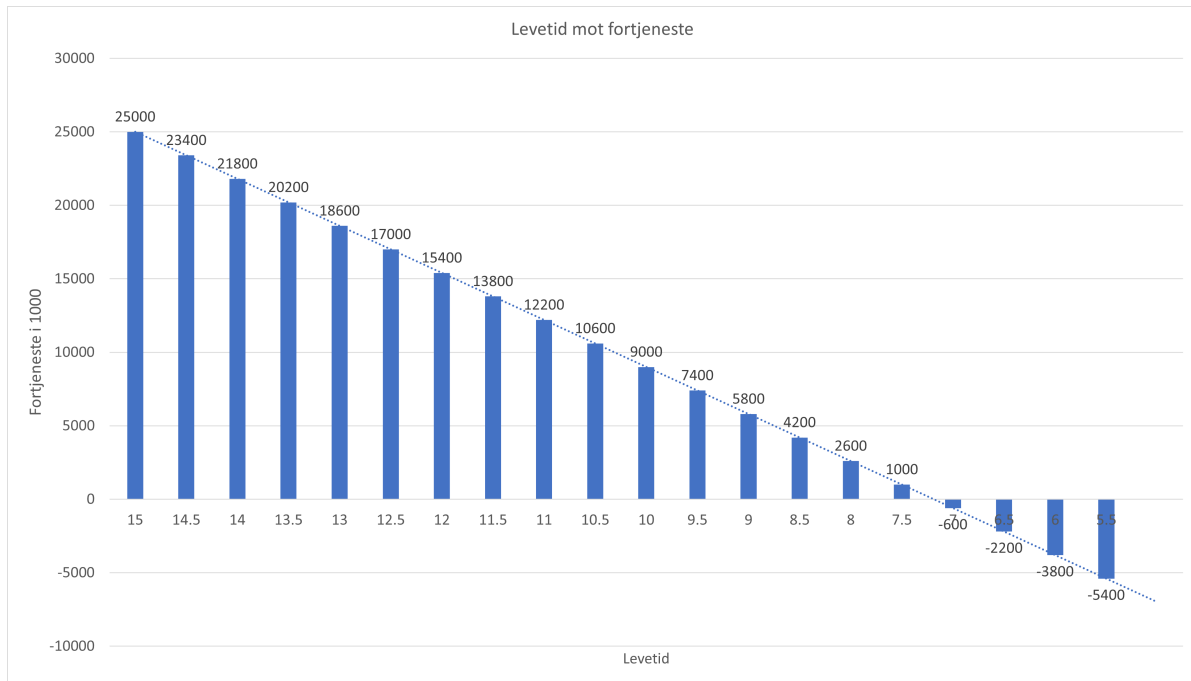
I "QuantumScape" sin teknologi dannes anoden av fast lithiummetall som bygges opp mellom den negative elektriske kontakten og separatorene. Dette blir da en en-cellet SSB. Andre to-cellede SSB teknologier baserer seg på en negativ katode av carbon og silisium. En utfordring for SSB er å ha en separator som kan tåle høy strømføring ved hurtiglading av f.eks. elbiler. Ved høy strømføring kan det dannede lithiumlaget skyte ut dendritter som ødelegger separatorene og får batteriet til å sprenges.



Figur 80: Tidligere SSB teknologier med en katode og svak separator kan implodere ved høy strømføring, en-cellet SSB med ren keramisk separator er QuantumScapes foreslåtte løsning per 12.01.2022. Hentet fra

Levetid

Den totale avkastningen på et energibehandlingssystem avhenger av levetiden til batteriet, og øker hvis innkjøpsprisen til batteriet (og systemet) kan deles over et større antall år. I eksempelet under settes årlig inntekt til 3200 NOK, total batterikapasitet settes til 10kWh og kostnad/kWh settes til 2300 NOK og levetiden måles mellom 15 og 5 år med intervall på 0.5 år.

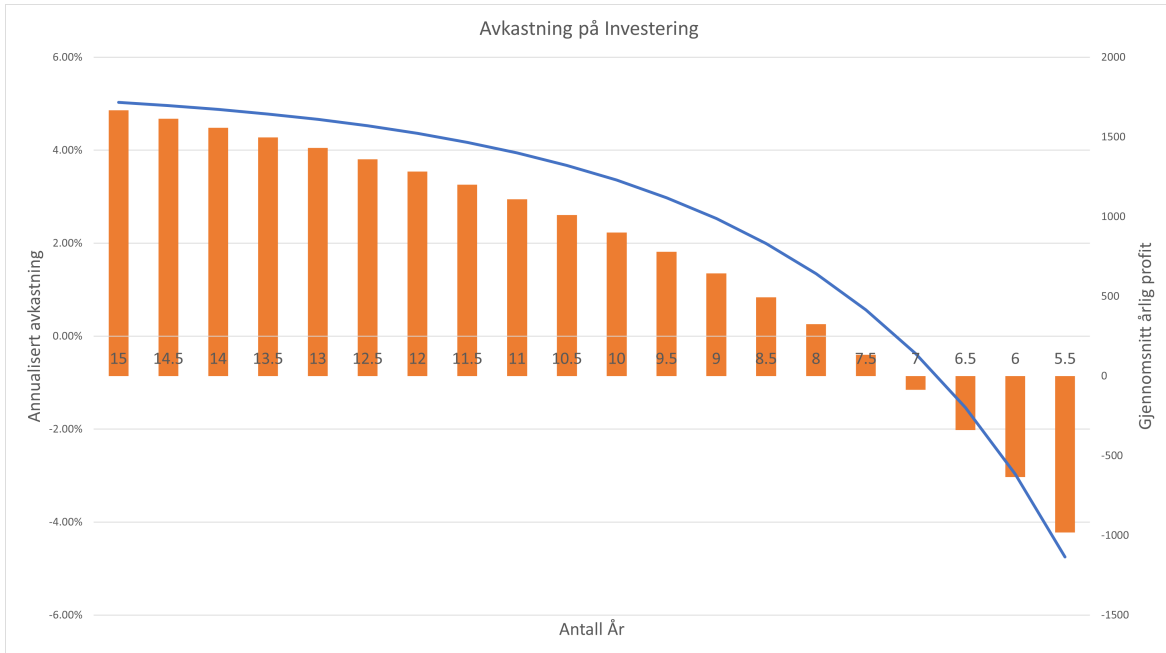


Figur 81: Diagram som viser fortjenesten med hensyn på levetiden

Fortjenesten øker lineært med levetiden. Likningen kan beskrives som:

$$Fortjeneste = \text{Årliginntekt} \cdot levetid - (\text{Kostnad}/kWh \cdot \text{TotalBatterikapasitet}) \quad (23)$$

For hvert ekstra år systemet opererer, øker fortjenesten med 3200 NOK. Figuren 86 viser hvordan endring i kostnad påvirker nullpunktet og muligheten for fortjeneste. Annualisert avkastning beveger seg dog ikke lineært (forklar hvorfor).



Figur 82: Figur som viser annualisert avkastning avhengig av levetid

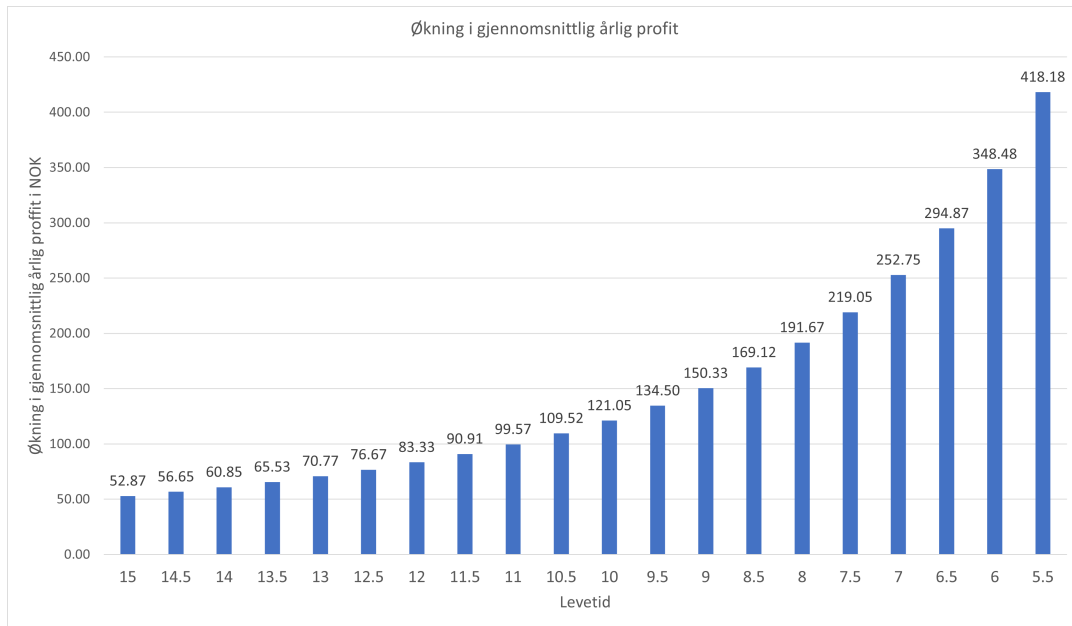
Figuren 82 viser hvordan gjennomsnittlig profitt og annualisert avkastning endrer seg mot levetiden. Gjennomsnittsprifitt beskrives med likningen

$$Gjennomsnittsprifitt = \frac{\text{Årliginntekt} \cdot \text{levetid} - (\text{Kostnad}/kWh \cdot \text{TotalBatterikapasitet})}{\text{Levetid}} \quad (24)$$

Den annualiserte avkastningen beskrives med likningen

$$Annualisertavkastning = \left(\frac{\text{Fortjeneste}}{\text{Kostnad}/kWh \cdot \text{TotalBatterikapasitet}} + 1 \right)^{\left(\frac{1}{\text{Levetid}} \right)} - 1 \quad (25)$$

Foretatte modelleringer viser en betydelig profittøkning ved forlenget levetid de første årene. Deretter flater kurven i 82 ut jo nærmere den kommer uendelig levetid. Ut i fra 82 vil en forlengelse av levetid fra 7,5 år til 8 år, resulterer i en økning i gjennomsnittsprifitt fra respektivt 133kr til 325kr. Det er en økning av fortjeneste på 244% gitt et halvt år forlenget levetid, som utgjør 1602 NOK over hele levetiden. Det faktum at inntekten ikke er konstant, men avhenger av strømprisen, egenprodusert energi og ytelsen til algoritmen, påvirker avkastningen på investeringen i stor grad, vist i resultatkapittelet.

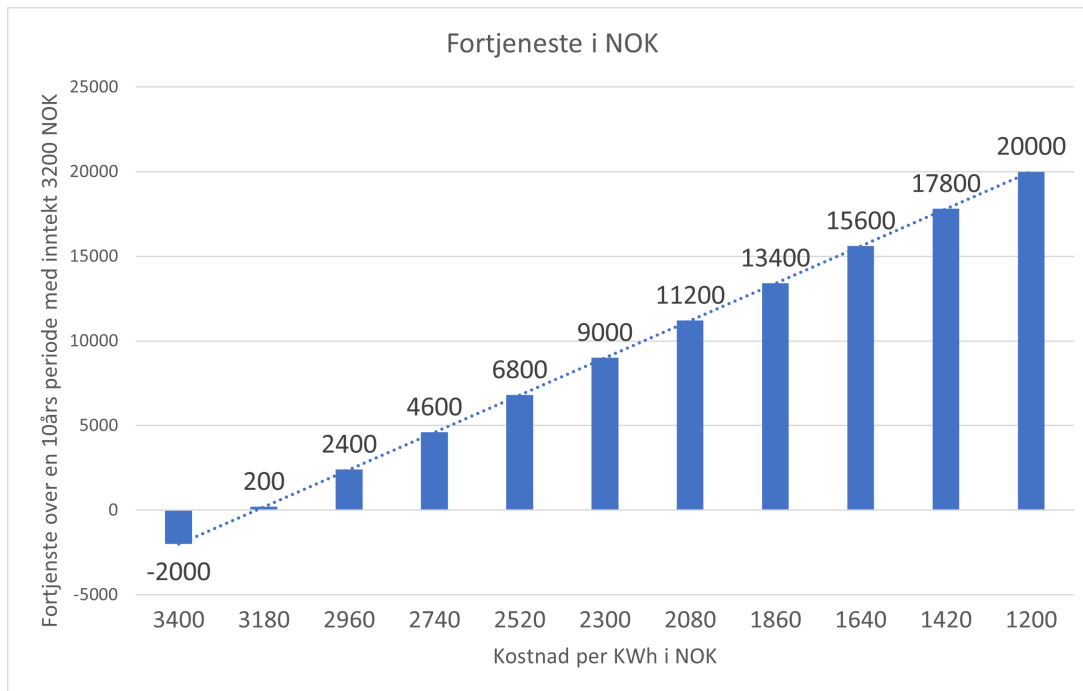


Figur 83: Diagram som viser hvor mye gjennomsnittlig årlig profitt øker avhengig av levetid

Figuren 83 viser økning av gjennomsnittlig årlig profitt for hvert år av forlenget levetid. Hvis levetiden til batteriet går fra 5.5 år til 6 år, øker gjennomsnittlig profitt med 348.48 NOK.

Kostnad

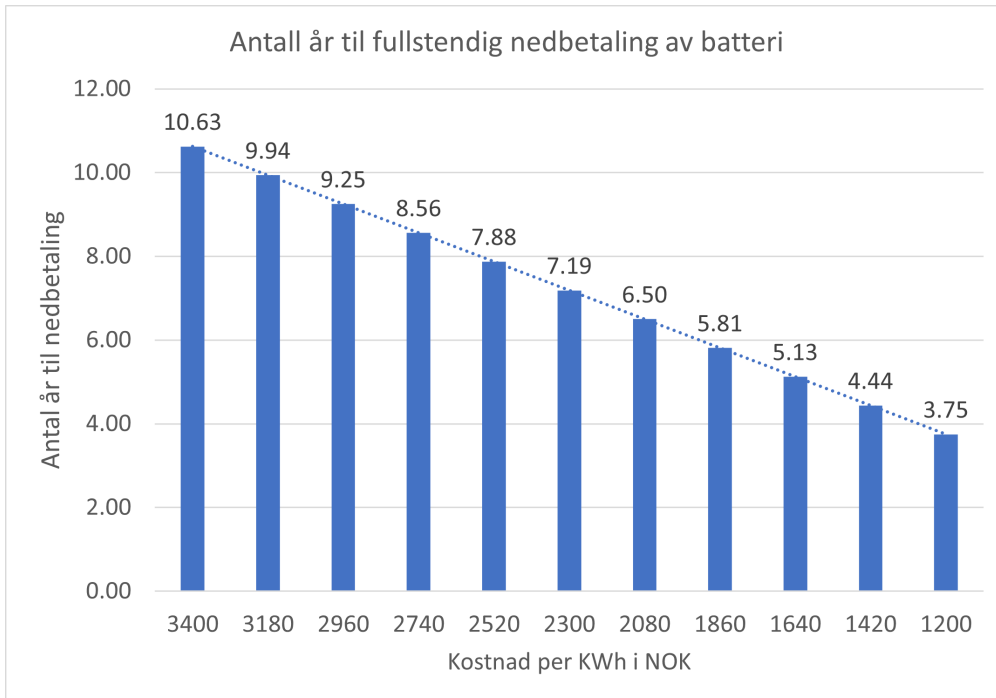
Kostnad per kWh er viktig å forholde seg til, og for at systemet skal lønne seg å installere, må inntekten over levetiden være større enn kostnaden. Det betyr at for å øke inntjeningen kan enten kostnaden gå ned eller levetiden gå opp.



Figur 84: Diagram som viser fortjensten over en 10års periode avhengig av celle kostnad/kWh

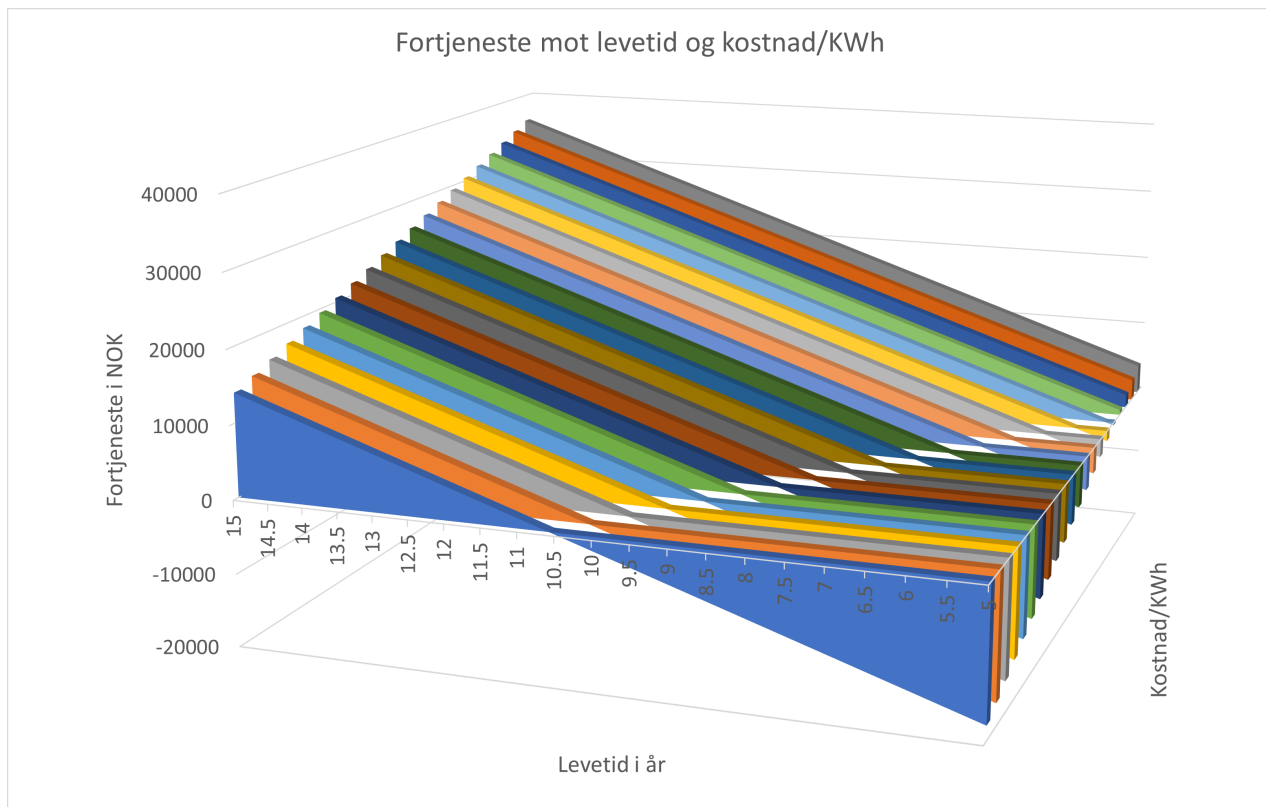
Når kostnaden av battericellen går ned stiger den potensielle fortjenesten. For å illustrere poenget viser figur 84 hvordan fortjenesten påvirkes over en 10års periode når kostnaden stiger. Grafen viser 11 punkter fra 3400NOK/kWh til 1200NOK/kWh med intervallet 220NOK/kWh. Inntekten årlig settes konstant til 3200NOK. Forholdet er lineært, og viser at for hver NOK kostnaden av batteriet reduseres med tjener 1 NOK per året som gir 10 NOK over hele 10års perioden og kan beskrives som $KostnadNOK = FortjenesteOver10ar - 10NOK \cdot kostnad/kWh$. Det lineære forholdet gjelder kun når investeringen er betalt kontant og ikke lånefinansiert.

Risikoen reduseres samtidig betydelig når kostnaden reduseres, fordi det betyr at nedbetalingstiden reduseres. Energimarkedet er i konstant endring, det forventes økt volatilitet og at strømprisen skal stabiliseres på høyere nivå 55EUR/MWh i Sør Norge og 25EUR/MWh i Nord Norge, sammenliknet med 100EUR/MWh i snitt fra Sept til Des 2021 [44]. Men det er prognoser og ingen garanti for at strømprisen holdes på nivå som kreves for at batterilagring skal være lønnsomt. Med kortere nedbetalingstid så øker sannsynligheten for at prognosene stemmer siden prognosens usikkerhet stiger i takt med hvor langt frem prognosen strekker seg.



Figur 85: Diagram som viser nedbetalingstid for et system på 10kWh og årlig inntekt på 3200 NOK

Antall år til fullstendig nedbetaling med hensyn på Kostnad per kWh er lineær og kan beskrives som $Antall_{\text{år}} = 0.00313 \cdot \text{Kostnad}/\text{kWh}$ vist i figur 85. Samme betingelser som i figur 84, 3200 NOK i årlig inntekt og ikke lånefinansiert. Figur 84 som viser fortjenesten og figur 85 må ses i sammenheng, når nedbetalingstiden faller øker inntjeningen og risiko reduseres og motsatt når nedbetalingstiden øker faller inntjeningen og risiko økes. Som vist i figur 86.



Figur 86: 3D figur som viser hvordan levetiden og kostnaden påvirker fortjeneste

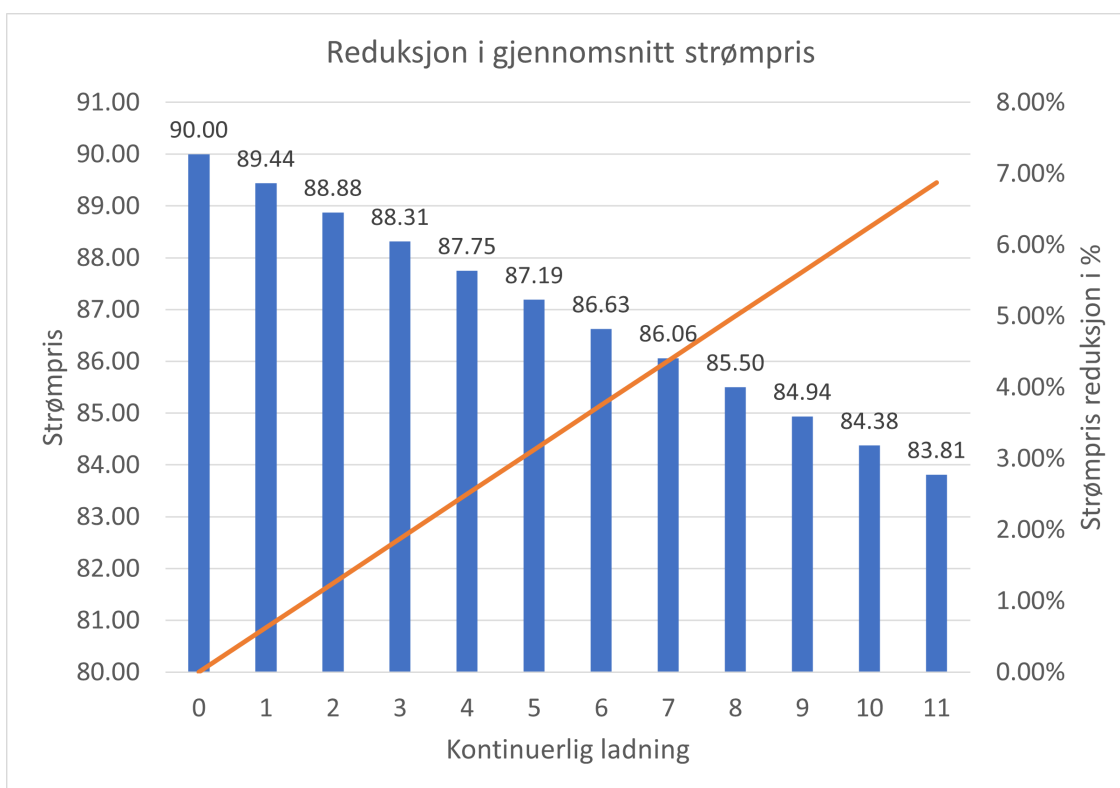
Figuren 86 viser forholdet mellom fortjeneste, levetid og kostnad når inntjening settes konstant til 3200 NOK og kapasitet på batteripakke til 10kWh. Grafen nærmest viser for en kostnad/kWh på 3400 NOK og den inderste viser kostnad/kWh på 1200 NOK samme som resten av figurene. Hovedpoenget er at maksimal lønnsomhet oppnås når levetiden er lengst og kostnad per kWh er høyest, og at det er betydelige variasjoner mellom beste og verste tilfelle.

Med en levetid på 5år og kostnad/kWh på 3400Nok gir det -18000 i fortjeneste mens med en levetid på 15år og kostnad/kWh på 1200 NOK gir den en fortjeneste på 36000 NOK, hvilket gir en differanse på 54000. Risikoen som systemet har mot levetiden kommer tydelig frem og må tas hensyn til når BMS systemet lages slik at batteriet skånes mest mulig mot unødvendige og potensielt ikke lønnsomt drift. Skjæringspunktet der systemet blir lønnsomt bev

Energitettheten er viktig dersom boligen som batteriet skal installeres i har begrenset plass, noe vi antar at flestparten av de potensielle brukerne av systemet er opptatt av. Men energitettheten har ikke en direkte positiv økonomisk effekt, som gir lavere prioritering enn de andre parametrene levetid, kontinuerlig kraft og kostnad/kWh.

Kontinuerlig kraft

Kontinuerlig kraft er viktig fordi det er en parameter som begrenser inntjeningsmuligheten. Store svingninger på strømprisen er positivt for batterilagringsystem, men en begrensende faktor er kontinuerlig lade kraft fordi det bestemmer hvor mye svingningene kan bli utnyttet. Det påvirker også måten algoritmen operer på fordi en må hensyn ta hvor mye strøm som kan lagres på en bestemt mengde tid. For noen batteriteknologier som litium-ion er ikke ladekurven lineær men avhengig av hvor mye ladning som allerede er på batteriet vist i ???. For å illustrere påvirkningen av kontinuerlig kraft viser figur 87 hvor mye snittprisen for et døgn kan reduseres på en time med ulike ladehastigheter. Med en kraft på 11KW over en time og konstant strømpris på 90EUR/MWh reduseres gjennomsnittkostnaden for strømmen gjennom et døgn(24 timer) med 6.88% til 83.81 EUR/MWh. En kraft på 5KW gir en reduksjon på 3.13% til 87.19 EUR/MWh



Figur 87: Diagram som viser reduksjon i gjennomsnitts strømpris avhengig av kontinuerlig kraft

Det som påvirker valget av batteriteknologi mest er

Graf som viser hvilken av de 4 parametrene som påvirker mest

Målet er å finne et batteri som passer best med de 4 parametrene beskrevet over, levetid, energitetthet, kontinuerlig kraft og kostnad/kWh. ulike aktuelle batterityper er beskrevet under.

For å sammenlikne de ulike teknologien brukes en matrise for å sammenlikne levetid, energitetthet, responstid, kostnad/kWh og innfører et poengsystem basert på hvilke faktorer som betyr mest for profitt muligheten.

Kilde til generell beskrivelse av de forskjellige batteriteknologiene.[3]

10.3 Tekniske begrensninger

Ytelsen til EBS begrenses av naturlige faktorer som lagringskapastiet mulig for litium-ion batteri, effektgraden av inverterne og termisk energitap. Overføringskapasitet tilgjengelig på strømkablene, frekvensen EBS tåler og dataprossesseringskraften nødvendig for optimaliseringsalgoritmen er alle tekniske begrensninger. Hvor godt EBS yter begrenses av nevnte faktorer. Hvor mye hver faktor påvirker ytelsen kartlegges ikke.

Likvel vil det være logisk å annta at fremtidig teknologisk innovasjon vil kunne by på mindre kostnad per kWh batterlagring, høyere strømflyt gjennom kabler, mindre tap gjennom invertere og større produksjon fra solceller.

EBS er modulært og består av flere avanserte teknologiske element. Derfor er ytelsen til EBS svært skjør, i form av at hvert ledd kan oppleve svikt, energitap eller andre uforutsette feil. I praksis vil derfor EBS være svært teknisk begrenset.

Det teoretiske systemet utviklet i denne oppgaven har sine tekniske begrensninger også. Datasettene brukt er i tilfeller historiske, i andre kvalifiserte gjetninger. Bestemt oppførsel er ikke opp til algoritmen i seg selv i alle ledd, men også til dels avhengig av begrensninger satt av utvikleren av programmet. Derfor kan det være et stort forbedringspotensial hva gjelder optimalisering av systemets programkode.

Prossesseringshastigheten til dataprogrammet kan virke begrensende, men vil også føre til et betydelig strømforbruk. Satt inn i regnskapet om besparelse vil det i praksis være et strømtap fra selve kjøringen av EBS. Dette strømtapet må kartlegges og finnes ubetydelig i forhold til besparelse for at EBS skal være lønnsomt og attraktivt å installere i en privat husholdning.

Som nevnt i kapitlet om tidligere forskning er ytelsen til MIP-løseren brukt i dataprogrammet svært avgjørende. Denne ytelsen avhenger til en viss grad av hva og hvordan den mates med objektet og begrensningene den tildeles, men også av avgjørelsene den tar under sin lokale kjøring. Her går den innom programfiler avhengig av problemene som oppstår i utregningen, hvorav noen er enkle heuretiske snarveier. Da MIP-løserne i synkende grad er avhengig av kvaliteten på *input*, ligger et stort forbedringspotensial - og med andre ord en teknisk begrensning - innenfor ytelsen til løserne tilgjengelig på markedet idag. De er ikke spesifikt utviklet for å minimere total kostnad av **EBS** eller liknende installasjoner.

Feltet som har blitt forsket på er ikke et nytt felt. Former for mikronett med solcelleforsyning og batterilager har blitt utprøvd hyppig etter årtusenskifte. Det finnes også mye forskning på

energilagings- og kontrollsystemer liknende det modellert i denne oppgaven. Det har også blitt gjort betydelig forskning på optimaliseringsproblem av typen MIP som brukes i denne oppgaven. Nyttigheten av dette papiret kommer av sammensetningen av tidligere forskning, og anvendelsen i form av et helt nytt ”produkt”, nemlig et **EBS** som opererer med flere nyanserte betingelser. Liknende systemer har blitt konkludert som ”ulønnsomme” i andre forskningspapir, da batteriets levetid mot kostnad og tap gjennom inverter kombinert med stabile strømpriser har resultert i større belastning enn vinning for potensielle kunder.

Hvordan kunne ESS blitt modellert bedre? Hva har vi oversett?

Kan private ESS eksistere i symbiose med offentlige ESS? Isåfall med skatt?

Selve batteriet, vil det i fremtiden være hovedsaklig litium eller gått over til flow? Råvareprisene stiger osv...

mer vind og fornybar fører til mer bruk av gass, kan batterier være med på å forhindre det?

strømpris fremover begrensinger i modell på hvordan degraderingen av batterier påvirkes med ulike max og min syklususer. Er det fordel at alle batteri systemene feks styres av et selskap eller staten slik at de ulike systemene vett hva de andre gjør eller at hver enkelt privat har sitt eget system og tar risikoen selv. potensielle feil i modellering av ladekurve og tapgjennom inverter mot hvordan egenskapene er i virkelige system.

Batteriets miljøpåvirkning avhenger i størst grad av gjenvinningsprosessen. Avhengig av kjemisk sammensetning brukes ulike prosesser med ulike utslipp for å gjenvinne batteriet. Det finnes ingen universell gjenvinningsprosess, men regnskap gjort i [54] og en artikkel kalt *Recycling and environmental issues of lithium-ion batteries: Advances, challenges and opportunities* fra 2021 [10], tyder på at energi og utslipp av ulike gjenvinningsprosesser står for mesteparten av klimaavtrykket til litium-ion batteri.

10.4 strømmarkedet

På slutten av 1990-tallet kom dot-com krasjet som satte en midlertid stopper på utviklingen, men i 2001 videreutviklet Storbritannia strømmarkedet til å inkludere bilateral kontraktbasert frihandel [21].

I et pool markedet melder produsentene inn hvor mye de forventer å produsere den kommende dagen, og konsumentene kan legge inn hvor mye det ønsker å kjøpe. Dette i norden gjøres hos strømbørsen Nord Pool, de har to strømmarkeder *day-ahead* og *intraday*. *Day-ahead* markedet er der hvor de store volumene handles, markedet lukker kl 12.00 hver dag, då må alle parter ha lagt inn ordrene. Det finnes 4 ordre typer på *day-ahead* markedet hos Nord Pool.

- Single hourly orders

- Block orderes
- Exclusive groups
- Flexi orderes

Single hourly orderes er det meste brukte, her angir kjøper eller selger volum og pris de ønsker å handle for hver time. Det er mulig å velge mellom pris avhengig og pris uavhengig ordre. Et eksempel på en pris avhengig ordre er dersom en aktør ønsker å kjøpe 70MW alle timene mellom 01-24 til en minimumspris på -500€ og en maksimumspris på 3000€. Øktereren vil motta et leveranseskjema tilsvarende det volum som er spesifisert for hver enkelt time uavhengig av prisnivå. Prisavhengige ordre gjør det mulig å definere flere prissteg feks at kl 01 ønsker aktøren å kjøpe 40MW til en pris mellom -500€ og 50€, selge -10MW dersom prisen er mellom 60€ og 70€, og selge -40MW dersom prisen er mellom 130€ og 3000€. Aktøren godtar da at Nord Pool gjennomfører en lineær interpolasjon justere handlet volum for å passe med andre ordre vist i figur 88 [38].

Price	-500	50	50.1	55	55.1	60	60.1	3 000
Hour								
01	50	50	0	0	-10	-10	-30	-30
02	100	70	0	0	-50	-50	-30	-30
03	150	50	0	0	-10	-10	-30	-30
:	:	:	:	:	:	:	:	:
24	170	100	0	0	-5	-5	-30	-30

Figur 88: Tabell som viser eksempel på prisavhengig ordre hentet fra [38]

Block ordre brukes når en kunde ønsker å handle for et bestemt antall sammenhengende timer gjennom dagen. Dette kan være for å unngå dyre kostnader ved oppstart og stenging av anlegg for potensielt kun 1 time drift. Exclusive group brukes dersom en aktør har flere blokker som ikke kan gå gjennom parallelt, ordretypen sikrer at kun en av ordrene i en gruppering kan gå gjennom om gangen. Flexi ordre brukes av aktører som har et stort forbruk slik at de kan selge tilbake strøm som de ikke bruker til spot pris ved å stenge ned noe av den industrielle prosessen i en bestemt tidsperiode, flexi ordre kan ikke vare lenger enn 23 timer. Navnet Flexi kommer fra at det er fleksibelt når ordren starter men at det blir bestemt av en algoritme.

10.5 Kode optimalisering

```
1 import matplotlib.pyplot as plt
2 import numpy as np
3 import pandas as pd
4 from mip import Model, xsum, CONTINUOUS, BINARY
5 import math
6 import random
7 from statsmodels.graphics.gofplots import qqplot
8 from canvas_display import sol
9
10 #Setter slik at alle rader og kolloner viser
11 pd.set_option('display.max_columns', None)
12 pd.set_option('display.max_rows', None)
13
14
15
16 kapasitet_batteri = 13.5 # KWh
17 alt_data = pd.DataFrame()
18 def lastdata():
19
20     #Laste inn strømpris
21     strom = pd.read_excel(r"C:\Users\gonna\pythonProject\betingelser\elspot -
22 prices_2021_hourly_eur.xlsx", index_col=None, header=2)
23     alt_data['Strompris'] = strom['Kr.sand']
24     alt_data['Strompris'] = alt_data['Strompris'].str.replace('[A-Za-z]', '',
25 regex=True).str.replace(',', '.').astype(float)
26     #laste inn dato
27     for t in range(len(alt_data.index)):
28         alt_data.at[t, 'Dato'] = strom.at[t, 'Dato'].strftime('%d-%m-%y')
29
30     #Laste inn solceller
31     pv = pd.read_csv(r"C:\Users\gonna\pythonProject\betingelser\pvforstvg.csv",
32 index_col=None, header=3)
33     pv['electricity'] = pv['electricity']/10
34     alt_data['Produksjon'] = pv['electricity']
35     #Laste inn vind
36     wind = pd.read_csv(r"C:\Users\gonna\pythonProject\betingelser\ninja_wind.csv
37 ", index_col=None, header=3)
38     wind['electricity'] = wind['electricity']*2
39     #alt_data['Produksjon'] = alt_data['Produksjon'] + wind['electricity']
40
41     return alt_data
42
43 lastdata()
44
45
46
47 def glidendegjennomsnitt(alt_data):
48     #Kalkulere 5 og 9 timers glidende gjennomsnitt
49     alt_data["MA5"] = alt_data["Strompris"].rolling(window=6).mean()
50     alt_data["MA9"] = alt_data["Strompris"].rolling(window=9).mean()
51     alt_data["Signal"] = alt_data["MA5"]-alt_data["MA9"]
52
53 glidendegjennomsnitt(alt_data)
```

```

50
51 #Finner strømpris for KWh
52 alt_data['StromprisKWh'] = alt_data["Strompris"]/1000
53 alt_data['StromprisKWhMVA'] = alt_data['StromprisKWh']*1.25
54 alt_data['StromprisKWhMVA'] = alt_data['StromprisKWhMVA']+0.02051
55
56
57 #Laste inn forbruk
58 li = []
59 jan = pd.read_csv(r"C:\Users\gonna\pythonProject\betingelser\meteringvalues-
    jan21.csv", index_col=None, header=0)
60 feb = pd.read_csv(r"C:\Users\gonna\pythonProject\betingelser\meteringvalues-
    feb21.csv", index_col=None, header=0)
61 mar = pd.read_csv(r"C:\Users\gonna\pythonProject\betingelser\meteringvalues-
    mar21.csv", index_col=None, header=0)
62 apr = pd.read_csv(r"C:\Users\gonna\pythonProject\betingelser\meteringvalues-
    april21.csv", index_col=None, header=0)
63 mai = pd.read_csv(r"C:\Users\gonna\pythonProject\betingelser\meteringvalues-
    mai21.csv", index_col=None, header=0)
64 jun = pd.read_csv(r"C:\Users\gonna\pythonProject\betingelser\meteringvalues-
    jun21.csv", index_col=None, header=0)
65 jul = pd.read_csv(r"C:\Users\gonna\pythonProject\betingelser\meteringvalues-
    jul21.csv", index_col=None, header=0)
66 aug = pd.read_csv(r"C:\Users\gonna\pythonProject\betingelser\meteringvalues-
    aug21.csv", index_col=None, header=0)
67 sept = pd.read_csv(r"C:\Users\gonna\pythonProject\betingelser\meteringvalues-
    sept21.csv", index_col=None, header=0)
68 okt = pd.read_csv(r"C:\Users\gonna\pythonProject\betingelser\meteringvalues-
    okt21.csv", index_col=None, header=0)
69 nov = pd.read_csv(r"C:\Users\gonna\pythonProject\betingelser\meteringvalues-
    nov21.csv", index_col=None, header=0)
70 des = pd.read_csv(r"C:\Users\gonna\pythonProject\betingelser\meteringvaluse-
    des21.csv", index_col=None, header=0)
71 li.append(jan)
72 li.append(feb)
73 li.append(mar)
74 li.append(apr)
75 li.append(mai)
76 li.append(jun)
77 li.append(jul)
78 li.append(aug)
79 li.append(sept)
80 li.append(okt)
81 li.append(nov)
82 li.append(des)
83 frame = pd.concat(li, axis=0, ignore_index=True)
84 alt_data["Forbruk"] = frame['KWH 60 Forbruk'].str.replace('[A-Za-z]', '', regex=
    True).str.replace(',','').astype(float)
85
86
87
88
89
90

```

```

91 def milp(pris, kapasitet_batteri, forbruk, produksjon, sol):
92
93     T = range(len(forbruk))
94
95     m = Model()
96
97
98     batteri = [m.add_var(var_type=CONTINUOUS, lb=kapasitet_batteri*0.1, ub=
kapasitet_batteri*0.9) for t in T]
99     flyt = [m.add_var(var_type=CONTINUOUS, lb=0, ub=10) for t in T]
100     flytforbruk = [m.add_var(var_type=CONTINUOUS, lb=0, ub=11) for t in T]
101     nettleie = [m.add_var(var_type=CONTINUOUS) for t in T]
102     tapinverter = [m.add_var(var_type=CONTINUOUS, lb=0, ub=0.9) for t in T]
103     tapinverterforbruk = [m.add_var(var_type=CONTINUOUS, lb=0) for t in T]
104     salg = [m.add_var(var_type=CONTINUOUS, lb=0, ub=20) for t in T]
105
106
107     m.objective = (xsum(pris[t] * (flyt[t]-salg[t]/1.25) for t in T) + xsum(
nettleie[t] for t in T) + (xsum(pris[t] * flytforbruk[t] for t in T)))
108
109
110     # Setter at flyten inn skal være samme størrelse som forbruk
111     m += xsum(flyt[t] + flytforbruk[t] + (produksjon[t]*sol) for t in T) == xsum
((forbruk[t] + tapinverter[t]) + tapinverterforbruk[t] + salg[t] for t in T)
112
113     #Legger til flyt og trekker fra forbruk fra forrige tilstand
114     for t in T:
115         if t == 0:
116             m += batteri[t] == (kapasitet_batteri/2) - forbruk[t] + flyt[t] + (
produksjon[t]*sol) + flytforbruk[t] - tapinverter[t] - tapinverterforbruk[t]
- salg[t]
117         else:
118             m += batteri[t] == batteri[t-1] - forbruk[t] + flyt[t] + (produksjon
[t]*sol) + flytforbruk[t] - tapinverter[t] - tapinverterforbruk[t] - salg[t]
119
120     #Tar hensyn til ladekurve
121     for t in T:
122         m += flyt[t] <= 0.0000670163*math.pow((batteri[t]/kapasitet_batteri)
*100, 3) - 0.0131177*math.pow((batteri[t]/kapasitet_batteri)*100, 2) +
0.63076923*((batteri[t]/kapasitet_batteri)*100) + 2.30769231
123
124     #Ta hensyn til tapt kapasitet overtid
125     for t in T:
126         m += batteri[t] <= kapasitet_batteri - ((0.0000325 * xsum(flyt[t] +
produksjon[t]*sol for t in T)/kapasitet_batteri) * kapasitet_batteri)# -
kapasitet_batteri*0.15
127
128
129     #flytforbruk skal ikke være større en det som går til forbruk
130     for t in T:
131         m += flytforbruk[t] <= forbruk[t]
132
133     #Ta hensyn til tap gjennom Inverter til batteri
134     for t in T:

```

```

135     #m += tapinverter[t] >= flyt[t]*0.14
136     if flyt[t] >= 6:
137         m += tapinverter[t] >= flyt[t]*0.07
138     elif 3 <= flyt[t] <= 5.999999999999999:
139         m += tapinverter[t] >= flyt[t]*0.2
140     elif flyt[t] <= 2.999999999999999:
141         m += tapinverter[t] >= flyt[t] * 0.47
142
143     #Ta hensyn til tap inverter forbruk
144     for t in T:
145         m += tapinverterforbruk[t] >= (forbruk[t] - flytforbruk[t])*0.05
146
147     #Ta hensyn til nettleie
148     teller = 0
149     for t in T:
150         if 0 <= teller <= 5 or 22 <= teller <= 23:
151             m += nettleie[t] == flyt[t] * (0.0292*1.25) + flytforbruk[t] *
(0.0292*1.25) #kveld
152             if 6 <= teller <= 21:
153                 m += nettleie[t] == flyt[t] * (0.0417*1.25) + flytforbruk[t] *
(0.0417*1.25) #dag
154             if teller == 23:
155                 teller = 0
156             teller += 1
157
158
159     m.optimize()
160
161     teller = 0
162     for t in T:
163         alt_data.at[t, 'TapInverterbatteri'] = tapinverter[t].x
164         alt_data.at[t, 'TapInverterforbruk'] = tapinverterforbruk[t].x
165         alt_data.at[t, 'Flyt'] = flyt[t].x
166         alt_data.at[t, 'Nettleie'] = nettleie[t].x
167         alt_data.at[t, 'Batteri'] = ((batteri[t].x)/kapasitet_batteri)*100
168         alt_data.at[t, 'Flytforbruk'] = flytforbruk[t].x
169         alt_data.at[t, 'Salg'] = salg[t].x
170
171         if 0 <= teller <= 5 or 22 <= teller <= 23:
172             if (forbruk[t] - produksjon[t]*sol) > 0:
173                 alt_data.at[t, 'Nettleieuten'] = (forbruk[t]-(produksjon[t]*sol)
) * (0.0292*1.25) #kveld
174             else:
175                 alt_data.at[t, 'Nettleieuten'] = 0
176         if 6 <= teller <= 21:
177             if (forbruk[t] - produksjon[t]*sol) > 0:
178                 alt_data.at[t, 'Nettleieuten'] = (forbruk[t]-(produksjon[t]*sol)
) * (0.0417*1.25) #dag
179             else:
180                 alt_data.at[t, 'Nettleieuten'] = 0
181         if teller == 23:
182             teller = 0
183         teller += 1
184

```

```

185     if (forbruk[t] - produksjon[t]*sol) > 0:
186         alt_data.at[t, 'PrisutenBatteri'] = (alt_data.at[t, "Forbruk"]*
alt_data.at[t, 'StromprisKWhMVA'])+alt_data.at[t, 'Nettleieuten'] - ((
alt_data.at[t, 'Produksjon']*alt_data.at[t, 'StromprisKWhMVA'])*sol*0.95)
187     else:
188         alt_data.at[t, 'PrisutenBatteri'] = (alt_data.at[t, "Forbruk"]*
alt_data.at[t, 'StromprisKWhMVA'])+alt_data.at[t, 'Nettleieuten'] - (
alt_data.at[t, 'Produksjon']*alt_data.at[t, 'StromprisKWh']*sol*0.95)
189
190     alt_data['PrismedBatteri'] = (alt_data['StromprisKWhMVA']*alt_data['Flyt
'])+alt_data['Nettleie'] + (alt_data['StromprisKWhMVA']*alt_data['
Flytforbruk']) - (alt_data['Salg']*alt_data['StromprisKWh'])
191     #legger til ladning som allerede er på batteri
192     alt_data.at[0, 'PrismedBatteri'] = (kapasitet_batteri/2) * alt_data.at[0, '
StromprisKWhMVA']
193     #trekker fra det som er på batteri når oppgaven slutter
194     alt_data.at[len(alt_data)-1, 'PrismedBatteri'] = alt_data.at[len(alt_data)
-1, 'PrismedBatteri'] - (alt_data.at[len(alt_data)-1, 'Batteri']/100) *
kapasitet_batteri * alt_data.at[len(alt_data)-1, 'StromprisKWhMVA']
195     for t in range(len(alt_data.index)):
196         if t == 0:
197             alt_data.at[t, 'AkkfortjenesteBatteri'] = sum(alt_data['
PrisutenBatteri'].head(0)) - sum(alt_data['PrismedBatteri'].head(0))
198             alt_data.at[t, 'AkkfortjenesteBatteri'] = sum(alt_data['
PrisutenBatteri'].head(1)) - sum(alt_data['PrismedBatteri'].head(1))
199         else:
200             alt_data.at[t, 'AkkfortjenesteBatteri'] = sum(alt_data['
PrisutenBatteri'].head(t+1)) - sum(alt_data['PrismedBatteri'].head(t+1))
201         alt_data['AkkfortBatteriDiff'] = alt_data['AkkfortjenesteBatteri'].diff()
202
203     return alt_data
204
205
206 milp(alt_data['StromprisKWhMVA'], kapasitet_batteri, alt_data["Forbruk"],
alt_data['Produksjon'], sol)
207 #alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\betingelser\ferdig\
Krsand21\13KW_EB_uS_Krsand21.xlsx")
208
209
210
211
212 def printdata(alt_data):
213     print("Totalt forbruk {} KWh".format(round(sum(alt_data['Forbruk']), 1)))
214     print("Kostnad uten batteri {} EUR".format(round(sum(alt_data['
PrisutenBatteri']), 2)))
215     print("Kostnad med batteri {} EUR".format(round(sum(alt_data["PrismedBatteri
"]), 2)))
216     print("Differanse {} EUR i besparelse, hvilket betyr en reduksjon på {}%".
format(round(sum(alt_data['PrisutenBatteri']) - sum(alt_data['PrismedBatteri
']), 2),
217
218         round(((sum(alt_data['PrisutenBatteri']) - sum(
alt_data['PrismedBatteri']))) / sum(alt_data['PrisutenBatteri']))) *

```

```

100,
219
    2)))
220
221 printdata(alt_data)
222
223
224
225 #print(alt_data)
226
227
228 def align_yaxis_np(ax1, ax2):
229     axes = np.array([ax1, ax2])
230     extrema = np.array([ax.get_ylim() for ax in axes])
231     tops = extrema[:,1] / (extrema[:,1] - extrema[:,0])
232
233     if tops[0] > tops[1]:
234         axes, extrema, tops = [a[::-1] for a in (axes, extrema, tops)]
235
236
237     tot_span = tops[1] + 1 - tops[0]
238
239     extrema[0,1] = extrema[0,0] + tot_span * (extrema[0,1] - extrema[0,0])
240     extrema[1,0] = extrema[1,1] + tot_span * (extrema[1,0] - extrema[1,1])
241     [axes[i].set_ylim(*extrema[i]) for i in range(2)]
242
243
244
245 alt_data.to_excel("13KW_JA_S_Kr21.xlsx")

```

10.6 Kode realtid-kjøring

```
1
2 import matplotlib.pyplot as plt
3 import numpy as np
4 import pandas as pd
5 from mip import Model, xsum, CONTINUOUS, BINARY
6 import math
7
8 #Setter slik at alle rader og kolloner viser
9 pd.set_option('display.max_columns', None)
10 pd.set_option('display.max_rows', None)
11
12
13 sol = 1
14 kapasitet_batteri = 13.5 # KWh
15 alt_data = pd.DataFrame()
16
17 #Laster inn strømprisen
18 strom = pd.read_excel(r"C:\Users\gonna\pythonProject\betingelser\elspot-
    prices_2021_hourly_eur.xlsx", index_col=None, header=2)
19 st = strom['Kr.sand'].head(4000)
20 st = st.reset_index()
21 alt_data['Strompris'] = st['Kr.sand']
22 alt_data['Strompris'] = alt_data['Strompris'].str.replace('[A-Za-z]', '', regex=
    True).str.replace(',', '.').astype(float)
23 for t in range(len(alt_data.index)):
24     alt_data.at[t, 'Dato'] = strom.at[t, 'Dato'].strftime('%d-%m-%y')
25
26 #Laster inn solceller
27 pv = pd.read_csv(r"C:\Users\gonna\pythonProject\betingelser\pvforstvg.csv",
    index_col=None, header=3)
28 pv['electricity'] = pv['electricity']/10
29 alt_data['Produksjon'] = pv['electricity']
30
31 #Finner strømpris for KWh
32 alt_data['StromprisKWh'] = alt_data["Strompris"]/1000
33 alt_data['StromprisKWhMVA'] = alt_data['StromprisKWh']*1.25
34 alt_data['StromprisKWhMVA'] = alt_data['StromprisKWhMVA']+0.02051
35
36 #laster inn forbruk
37 li = []
38
39 jan = pd.read_csv(r"C:\Users\gonna\pythonProject\betingelser\meteringvalues-
    jan21.csv", index_col=None, header=0)
40 feb = pd.read_csv(r"C:\Users\gonna\pythonProject\betingelser\meteringvalues-
    feb21.csv", index_col=None, header=0)
41 mar = pd.read_csv(r"C:\Users\gonna\pythonProject\betingelser\meteringvalues-
    mar21.csv", index_col=None, header=0)
42 apr = pd.read_csv(r"C:\Users\gonna\pythonProject\betingelser\meteringvalues-
    april21.csv", index_col=None, header=0)
43 mai = pd.read_csv(r"C:\Users\gonna\pythonProject\betingelser\meteringvalues-
    mai21.csv", index_col=None, header=0)
44 jun = pd.read_csv(r"C:\Users\gonna\pythonProject\betingelser\meteringvalues-
    jun21.csv", index_col=None, header=0)
```



```

45 jul = pd.read_csv(r"C:\Users\gonna\pythonProject\betingelser\meteringvalues -
    jul21.csv", index_col=None, header=0)
46 aug = pd.read_csv(r"C:\Users\gonna\pythonProject\betingelser\meteringvalues -
    aug21.csv", index_col=None, header=0)
47 sept = pd.read_csv(r"C:\Users\gonna\pythonProject\betingelser\meteringvalues -
    sept21.csv", index_col=None, header=0)
48 okt = pd.read_csv(r"C:\Users\gonna\pythonProject\betingelser\meteringvalues -
    okt21.csv", index_col=None, header=0)
49 nov = pd.read_csv(r"C:\Users\gonna\pythonProject\betingelser\meteringvalues -
    nov21.csv", index_col=None, header=0)
50 des = pd.read_csv(r"C:\Users\gonna\pythonProject\betingelser\meteringvaluse -
    des21.csv", index_col=None, header=0)
51 li.append(jan)
52 li.append(feb)
53 li.append(mar)
54 li.append(apr)
55 li.append(mai)
56 li.append(jun)
57 li.append(jul)
58 li.append(aug)
59 li.append(sept)
60 li.append(okt)
61 li.append(nov)
62 li.append(des)
63 frame = pd.concat(li, axis=0, ignore_index=True)
64 alt_data["Forbruk"] = frame['KWH 60 Forbruk'].str.replace('[A-Za-z]', '', regex=
    True).str.replace(',', '.').astype(float)
65
66
67
68 def milp(pris, kapasitet_batteri, forbruk, produksjon, sol, forrigetilstnad, tid
    , tel):
69
70     T = range(len(forbruk))
71
72     m = Model()
73
74
75     batteri = [m.add_var(var_type=CONTINUOUS, lb=kapasitet_batteri*0.1, ub=
    kapasitet_batteri*0.9) for t in T]
76     flyt = [m.add_var(var_type=CONTINUOUS, lb=0, ub=10) for t in T]
77     flytforbruk = [m.add_var(var_type=CONTINUOUS, lb=0, ub=11) for t in T]
78     nettleie = [m.add_var(var_type=CONTINUOUS) for t in T]
79     tapinverter = [m.add_var(var_type=CONTINUOUS, lb=0, ub=0.9) for t in T]
80     tapinverterforbruk = [m.add_var(var_type=CONTINUOUS, lb=0) for t in T]
81     salg = [m.add_var(var_type=CONTINUOUS, lb=0, ub=20) for t in T]
82
83
84     m.objective = (xsum(pris[t] * (flyt[t]-salg[t]/1.25) for t in T) + xsum(
    nettleie[t] for t in T) + (xsum(pris[t] * flytforbruk[t] for t in T)))
85
86
87     # Setter at flyten inn skal være samme størrelse som forbruk
88     m += xsum(flyt[t] + flytforbruk[t] + (produksjon[t]*sol) for t in T) == xsum

```

```

89 ((forbruk[t] + tapinverter[t]) + tapinverterforbruk[t] + salg[t] for t in T)
90 #Legger til flyt og trekker fra forbruk fra forrige tilstand
91 for t in T:
92     if t == 0:
93         m += batteri[t] == forrigetilstand - forbruk[t] + flyt[t] + (
produksjon[t]*sol) + flytforbruk[t] - tapinverter[t] - tapinverterforbruk[t]
- salg[t]
94     else:
95         m += batteri[t] == batteri[t-1] - forbruk[t] + flyt[t] + (produksjon
[t]*sol) + flytforbruk[t] - tapinverter[t] - tapinverterforbruk[t] - salg[t]
96
97 #Tar hensyn til ladekurve
98 for t in T:
99     m += flyt[t] <= 0.0000670163*math.pow((batteri[t]/kapasitet_batteri)
*100, 3) - 0.0131177*math.pow((batteri[t]/kapasitet_batteri)*100, 2) +
0.63076923*((batteri[t]/kapasitet_batteri)*100) + 2.30769231
100
101 #Ta hensyn til tapt kapasitet overtid
102 for t in T:
103     m += batteri[t] <= kapasitet_batteri - ((0.0000325 * xsum(flyt[t] +
produksjon[t]*sol for t in T)/kapasitet_batteri) * kapasitet_batteri)
104
105
106 #flytforbruk skal ikke være større en det som går til forbruk
107 for t in T:
108     m += flytforbruk[t] <= forbruk[t]
109
110 #Ta hensyn til tap gjennom Inverter til batteri
111 for t in T:
112     #m += tapinverter[t] >= flyt[t]*0.14
113     if flyt[t] >= 6:
114         m += tapinverter[t] >= flyt[t]*0.07
115     elif 3 <= flyt[t] <= 5.999999999999:
116         m += tapinverter[t] >= flyt[t]*0.2
117     elif flyt[t] <= 2.9999999999:
118         m += tapinverter[t] >= flyt[t] * 0.47
119
120 #Ta hensyn til tap inverter forbruk
121 for t in T:
122     m += tapinverterforbruk[t] >= (forbruk[t] - flytforbruk[t])*0.05
123
124 #Ta hensyn til nettleie
125 teller = tel
126 for t in T:
127     if 0 <= teller <= 5 or 22 <= teller <= 23:
128         m += nettleie[t] == flyt[t] * (0.0292*1.25) + flytforbruk[t] *
(0.0292*1.25) #kveld
129     if 6 <= teller <= 21:
130         m += nettleie[t] == flyt[t] * (0.0417*1.25) + flytforbruk[t] *
(0.0417*1.25) #dag
131     if teller == 23:
132         teller = 0
133     teller += 1

```

```

134
135
136 m.optimize()
137
138 teller = tel
139 for t in T:
140     alt_data.at[t, 'TapInverterbatteri'] = tapinverter[t].x
141     alt_data.at[t, 'TapInverterforbruk'] = tapinverterforbruk[t].x
142     alt_data.at[t, 'Flyt'] = flyt[t].x
143     alt_data.at[t, 'Batteri'] = ((batteri[t].x)/kapasitet_batteri)*100
144     alt_data.at[t, 'Salg'] = 0
145     if batteri[t].x - forrigetilstnad + salg[t].x < 0:
146         #under utladning forbruk - det som kommer fra batteriet - eventuelt
147         som kommer fra solceller
148         if (batteri[t].x - forrigetilstnad)*0.95*-1 > alt_data.at[tid, '
149         Forbruk']:
150             alt_data.at[t, 'Salg'] = (alt_data.at[tid, 'Forbruk'] + (batteri
151             [t].x - forrigetilstnad)*0.95)*-1*0
152             alt_data.at[t, 'Flytforbruk'] = alt_data.at[tid, 'Forbruk'] + (
153             batteri[t].x - forrigetilstnad + alt_data.at[t, 'Salg'])*0.95
154         else:
155             #under oppladning forbruk - eventuelt som kommer fra solceller
156             alt_data.at[t, 'Flytforbruk'] = alt_data.at[tid, 'Forbruk'] - ((
157             alt_data.at[tid, 'Produksjon']*0.95 - (batteri[t].x - forrigetilstnad - flyt
158             [t].x + tapinverter[t].x))*sol
159
160
161     if 0 <= teller <= 5 or 22 <= teller <= 23:
162         if (alt_data.at[tid, 'Forbruk'] - alt_data.at[tid, 'Produksjon']*sol
163         ) > 0:
164             alt_data.at[t, 'Nettleieuten'] = (alt_data.at[tid, 'Forbruk'] -
165             alt_data.at[tid, 'Produksjon']*sol) * (0.0292*1.25) #kveld
166         else:
167             alt_data.at[t, 'Nettleieuten'] = 0
168     if 6 <= teller <= 21:
169         if (alt_data.at[tid, 'Forbruk'] - alt_data.at[tid, 'Produksjon']*sol
170         ) > 0:
171             alt_data.at[t, 'Nettleieuten'] = (alt_data.at[tid, 'Forbruk'] -
172             alt_data.at[tid, 'Produksjon']*sol) * (0.0417*1.25) #dag
173         else:
174             alt_data.at[t, 'Nettleieuten'] = 0
175
176     if 0 <= teller <= 5 or 22 <= teller <= 23:
177         alt_data.at[t, 'Nettleie'] = (alt_data.at[t, 'Flytforbruk'] +
178         alt_data.at[t, 'Flyt']) * (0.0292*1.25) #kveld
179     if 6 <= teller <= 21:
180         alt_data.at[t, 'Nettleie'] = (alt_data.at[t, 'Flytforbruk'] +
181         alt_data.at[t, 'Flyt']) * (0.0417*1.25) #dag
182
183
184     if teller == 23:
185         teller = 0
186     teller += 1
187
188     if (alt_data.at[tid, 'Forbruk'] - alt_data.at[tid, 'Produksjon']*sol) >

```

```

0:
176     alt_data.at[t, 'PrisutenBatteri'] = (alt_data.at[tid, "Forbruk"]*
alt_data.at[tid, 'StromprisKWhMVA'])+alt_data.at[t, 'Nettleieuten'] - ((
alt_data.at[tid, 'Produksjon']*alt_data.at[tid, 'StromprisKWhMVA'])*sol
*0.95)
177     else:
178     alt_data.at[t, 'PrisutenBatteri'] = (alt_data.at[tid, "Forbruk"]*
alt_data.at[tid, 'StromprisKWhMVA'])+alt_data.at[t, 'Nettleieuten'] - (
alt_data.at[tid, 'Produksjon']*alt_data.at[tid, 'StromprisKWh']*sol*0.95)
179
180     alt_data.at[t, 'PrismedBatteri'] = (alt_data.at[tid, 'StromprisKWhMVA']*
alt_data.at[t, 'Flyt'])+alt_data.at[t, 'Nettleie'] + (alt_data.at[tid, '
StromprisKWhMVA']*alt_data.at[t, 'Flytforbruk']) - (alt_data.at[t, 'Salg']*
alt_data.at[tid, 'StromprisKWh'])
181
182     if t == 0:
183     alt_data.at[t, 'AkkfortjenesteBatteri'] = sum(alt_data['
PrisutenBatteri'].head(0)) - sum(alt_data['PrismedBatteri'].head(0))
184     alt_data.at[t, 'AkkfortjenesteBatteri'] = sum(alt_data['
PrisutenBatteri'].head(1)) - sum(alt_data['PrismedBatteri'].head(1))
185     else:
186     alt_data.at[t, 'AkkfortjenesteBatteri'] = sum(alt_data['
PrisutenBatteri'].head(t+1)) - sum(alt_data['PrismedBatteri'].head(t+1))
187     alt_data['AkkfortBatteriDiff'] = alt_data['AkkfortjenesteBatteri'].diff()
188
189     return alt_data
190
191
192
193
194 batteri = []
195 datasum = []
196 datasumuten = []
197 datafortjeneste = []
198 dataflyt = []
199 datanettleie = []
200 datanettleieuten = []
201 datasalg = []
202 dataflytforbruk = []
203 tid = 1
204 teller = 1
205
206 kl13 = 0
207 datapris = alt_data['StromprisKWhMVA'].head(24)
208 for t in range(3976):
209     dataforbruk = []
210     dataproduct = []
211     datasol1 = []
212     datasol2 = []
213     datasol3 = []
214     datapris = []
215     if t > 24:
216         for i in range(24):
217             if kl13 <= 12:

```

```

218         datapris.append(alt_data.at[i + tid, 'StromprisKWhMVA'])
219     else:
220         if i <= 36 - kl13:
221             datapris.append(alt_data.at[i + tid, 'StromprisKWhMVA'])
222         else:
223             datapris.append(alt_data.at[tid, 'StromprisKWhMVA'])
224
225     dataforbruk.append(alt_data.at[i+t-25, "Forbruk"])
226
227     if t <= 48:
228         dataproduct.append(alt_data.at[i+t-25, 'Produksjon'])
229     if 48 < t <= 72:
230         datasol1.append(alt_data.at[i+t-25, 'Produksjon'])
231         datasol2.append(alt_data.at[i+t-49, 'Produksjon'])
232         dataproduct.append((datasol1[i] + datasol2[i])/2)
233     if t > 72:
234         datasol1.append(alt_data.at[i+t-25, 'Produksjon'])
235         datasol2.append(alt_data.at[i+t-49, 'Produksjon'])
236         datasol3.append(alt_data.at[i+t-73, 'Produksjon'])
237         dataproduct.append((datasol1[i] + datasol2[i] + datasol3[i])/3)
238
239
240     else:
241         for i in range(t):
242             datapris.append(alt_data.at[i, 'StromprisKWhMVA'])
243             dataforbruk.append(alt_data.at[i, "Forbruk"])
244             dataproduct.append(alt_data.at[i, 'Produksjon'])
245
246     if t == 13:
247         kl = 0
248     if kl13 == 23:
249         kl13 = 0
250     kl13 += 1
251
252     print(datapris)
253     print(dataforbruk)
254     print(dataproduct)
255
256
257     if t == 1:
258         milp(datapris, kapasitet_batteri, dataforbruk, dataproduct, sol,
259             kapasitet_batteri/2, tid, teller)
260         batteri.append(alt_data.at[0, 'Batteri'])
261         datasum.append(alt_data.at[0, 'PrismedBatteri'])
262         datasumuten.append(alt_data.at[0, 'PrisutenBatteri'])
263         datafortjeneste.append(alt_data.at[0, 'PrismedBatteri'] - alt_data.at[0,
264             'PrisutenBatteri'])
265         datanettleie.append(alt_data.at[0, 'Nettleie'])
266         dataflyt.append(alt_data.at[0, 'Flyt'])
267         datasalg.append(alt_data.at[0, 'Salg'])
268         datanettleieuten.append((alt_data.at[0, 'Nettleieuten']))
269         dataforbruk.append(alt_data.at[0, 'Flytforbruk'])
270         tid += 1
271         teller += 1

```

```

270
271     if t > 1:
272         milp(datapris, kapasitet_batteri, dataforbruk, dataprod, sol, (alt_data[
'Batteri'].iloc[0]/100)*kapasitet_batteri, tid, teller)
273
274         batteri.append(alt_data.at[0, 'Batteri'])
275         dataflyt.append(alt_data.at[0, 'Flyt'])
276         datasum.append(alt_data.at[0, 'PrismedBatteri'])
277         datanettleie.append(alt_data.at[0, 'Nettleie'])
278         datasumuten.append(alt_data.at[0, 'PrisutenBatteri'])
279         datafortjeneste.append(alt_data.at[0, 'PrisutenBatteri'] - alt_data.at
[0, 'PrismedBatteri'] + datafortjeneste[t-2])
280         datasalg.append(alt_data.at[0, 'Salg'])
281         datanettleieuten.append((alt_data.at[0, 'Nettleieuten']))
282         dataforbruk.append(alt_data.at[0, 'Flytforbruk'])
283         tid += 1
284         if teller == 23:
285             teller = 0
286             teller += 1
287
288
289
290 datasum[0] = (kapasitet_batteri/2) * alt_data.at[0, 'StromprisKWhMVA']
291 datasum[len(datasum)-1] = datasum[len(datasum)-1] - (batteri[len(batteri)
-1]/100) * kapasitet_batteri * alt_data.at[len(alt_data)-1, 'StromprisKWhMVA
']
292
293
294
295 print(((sum(datasumuten)- sum(datasum))/sum(datasumuten))*100)
296 print(sum(datasumuten) - sum(datasum))
297 print(sum(datanettleieuten) - sum(datanettleie))
298 print(sum(dataflyt))
299 print((1 - ((sum(dataflyt) + sum(dataflytforbruk) - sum(alt_data['Forbruk'])))/
sum(alt_data['Forbruk']))) * 100)
300
301
302 alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\betingelser\ferdig\
Krsand21\13KW_EB_S_Krsand21.xlsx")
303 alt_datas = alt_datas.iloc[1:3976]
304
305 print('fra tidliger')
306 print(round(((sum(alt_datas['PrisutenBatteri']) - sum(alt_datas['PrismedBatteri '
])) / sum(alt_datas['PrisutenBatteri'])) * 100,2))
307 print(round((sum(alt_datas['PrisutenBatteri']) - sum(alt_datas['PrismedBatteri '
], 2))
308 print(round(sum(alt_datas['Nettleieuten']) - sum(alt_datas['Nettleie']),2))
309 print(round(sum(alt_datas['Flyt']),2))
310 print(round((1-((sum(alt_datas['Flyt']) + sum(alt_datas['Flytforbruk']) + (sum(
alt_datas['Produksjon'])*sol) - sum(alt_datas['Forbruk']) - sum(alt_datas['
Salg']))) / sum(alt_datas['Forbruk']))) * 100,1))
311
312 plt.plot(range(len(alt_datas)), alt_datas['Batteri'], label='Optimal')
313 plt.plot(range(3975), batteri, label='Real-time')

```

```

314 plt.xlabel('Dato')
315 plt.legend()
316 plt.ylabel('SOC i %')
317 plt.xticks(np.arange(0, len(alt_datas['Dato']), 500), alt_datas['Dato'].iloc
    [::500], rotation='horizontal')
318 plt.show()
319
320 plt.plot(range(len(alt_datas)), alt_datas['AkkfortjenesteBatteri'], label='
    Optimal')
321 plt.plot(range(3975), datafortjeneste, label='Real-time')
322 plt.xlabel('Dato')
323 plt.ylabel('Besbarelse i EUR')
324 plt.legend()
325 plt.xticks(np.arange(0, len(alt_datas['Dato']), 500), alt_datas['Dato'].iloc
    [::500], rotation='horizontal')
326 plt.show()

```

10.7 Kode GUI

```
1
2 import sys
3 import matplotlib.pyplot as plt
4 from matplotlib.backends.backend_qt5agg import FigureCanvasQTAgg as FigureCanvas
5 from matplotlib.figure import Figure
6 from matplotlib.widgets import Slider
7 from numpy import random
8 import matplotlib.pyplot as plt
9 import numpy as np
10 import pandas as pd
11 from mip import Model, xsum, CONTINUOUS, BINARY
12 import math
13 import time
14 from mainpandas import *
15 import matplotlib.dates as mdates
16 plt.rcParams.update({'font.size': 20})
17 plt.rcParams['axes.xmargin'] = 0
18
19
20 from PyQt5 import QtCore, QtGui, QtWidgets
21
22 sol = 1
23 sammenlikning = 1
24 sammenlikning2 = 1
25 navnaltdata = '13.5KW Enebolig Kr.sand 21'
26
27 class Ui_MainWindow(object):
28     def setupUi(self, MainWindow):
29         MainWindow.setObjectName("MainWindow")
30         MainWindow.resize(1500, 1000)
31         self.centralwidget = QtWidgets.QWidget(MainWindow)
32         self.centralwidget.setObjectName("centralwidget")
33         self.horizontalLayout = QtWidgets.QHBoxLayout(self.centralwidget)
34         self.horizontalLayout.setObjectName("horizontalLayout")
35         self.frame = QtWidgets.QFrame(self.centralwidget)
36         self.frame setFrameShape(QtWidgets.QFrame.StyledPanel)
37         self.frame setFrameShadow(QtWidgets.QFrame.Raised)
38         self.frame.setObjectName("frame")
39         self.verticalLayout = QtWidgets.QVBoxLayout(self.frame)
40         self.verticalLayout.setContentsMargins(0, 0, 0, 0)
41         self.verticalLayout.setSpacing(0)
42         self.verticalLayout.setObjectName("verticalLayout")
43         self.frame_2 = QtWidgets.QFrame(self.frame)
44         self.frame_2.setStyleSheet("QFrame{\n"
45 "    background-color: rgb(0,200,96);\n"
46 "    border: none;\n"
47 "    padding: 5px;\n"
48 "    color: rgb(205,230,255);\n"
49 "    border-radius: 5px;\n"
50 "    font: 75 14pt \"Candara\";\n"
51 "}")
52         self.frame_2 setFrameShape(QtWidgets.QFrame.StyledPanel)
53         self.frame_2 setFrameShadow(QtWidgets.QFrame.Raised)
```



```

54     self.frame_2.setObjectName("frame_2")
55     self.horizontalLayout_2 = QtWidgets.QHBoxLayout(self.frame_2)
56     self.horizontalLayout_2.setObjectName("horizontalLayout_2")
57     self.label = QtWidgets.QLabel(self.frame_2)
58     self.label.setObjectName("label")
59     self.horizontalLayout_2.addWidget(self.label)
60     self.verticalLayout.addWidget(self.frame_2, 0, QtCore.Qt.AlignTop)
61     self.frame_3 = QtWidgets.QFrame(self.frame)
62     sizePolicy = QtWidgets.QSizePolicy(QtWidgets.QSizePolicy.Preferred,
QtWidgets.QSizePolicy.Expanding)
63     sizePolicy.setHorizontalStretch(0)
64     sizePolicy.setVerticalStretch(0)
65     sizePolicy.setHeightForWidth(self.frame_3.sizePolicy().hasHeightForWidth
())
66     self.frame_3.setSizePolicy(sizePolicy)
67     self.frame_3 setFrameShape(QtWidgets.QFrame.StyledPanel)
68     self.frame_3 setFrameShadow(QtWidgets.QFrame.Raised)
69     self.frame_3.setObjectName("frame_3")
70     #create a horizontal layout
71     self.horizontalLayout_4 = QtWidgets.QHBoxLayout(self.frame_3)
72     self.horizontalLayout_4.setObjectName("horizontalLayout_4")
73
74     #label
75     self.frame_3.setStyleSheet("QFrame{\n"
76 "    font: 75 11pt \"Candara\";\n"
77 "}")
78     self.label1 = QtWidgets.QLabel(self.frame_3)
79     self.label1.setObjectName("label1")
80     self.horizontalLayout_4.addWidget(self.label1, 1)
81     #end label
82
83     ##Canvas Here
84     self.figure = plt.figure()
85     self.canvas = FigureCanvas(self.figure)
86     ##end of Canvas
87     ##Add Canvas
88     self.horizontalLayout_4.addWidget(self.canvas, 6)
89     ##end of horizontal layout
90
91     self.verticalLayout.addWidget(self.frame_3)
92     self.frame_4 = QtWidgets.QFrame(self.frame)
93     self.frame_4 setFrameShape(QtWidgets.QFrame.StyledPanel)
94     self.frame_4 setFrameShadow(QtWidgets.QFrame.Raised)
95     self.frame_4.setObjectName("frame_4")
96     self.verticalLayout_2 = QtWidgets.QGridLayout(self.frame_4)
97     self.verticalLayout_2.setContentsMargins(0, 0, 0, 0)
98     self.verticalLayout_2.setSpacing(0)
99     self.verticalLayout_2.setObjectName("verticalLayout_2")
100    #Horizontal layout
101    self.horizontalLayout_5 = QtWidgets.QHBoxLayout(self.frame_4)
102    self.horizontalLayout_5.setContentsMargins(0, 0, 0, 0)
103    self.horizontalLayout_5.setSpacing(0)
104    self.horizontalLayout_5.setObjectName("horizontalLayout_5")
105    #end of horizontal layout

```

```

106     self.pushButton = QtWidgets.QPushButton(self.frame_4, clicked=lambda:
self.plot0noversikt())
107     self.pushButton.setStyleSheet("QPushButton{\n"
108 "    background-color: rgb(0,200,96);\n"
109 "    border: none;\n"
110 "    padding: 5px;\n"
111 "    color: rgb(205,230,255);\n"
112 "    border-radius: 5px;\n"
113 "    font: 75 14pt \"Candara\";\n"
114 "}\n"
115 "\n"
116 "QPushButton: hover{\n"
117 "    background-color: rgb(0,240,115);\n"
118 "}\n"
119 "QPushButton: pressed{\n"
120 "    background-color: rgb(0,200,96);\n"
121 "}\n"
122 """)
123     self.pushButton.setObjectName("pushButton")
124     self.verticalLayout_2.addWidget(self.pushButton, 0, 0)
125     self.pushButton1 = QtWidgets.QPushButton(self.frame_4, clicked=lambda:
self.plot0nforbruk())
126     self.pushButton1.setStyleSheet("QPushButton{\n"
127 "    background-color: rgb(0,200,96);\n"
128 "    border: none;\n"
129 "    padding: 5px;\n"
130 "    color: rgb(205,230,255);\n"
131 "    border-radius: 5px;\n"
132 "    font: 75 14pt \"Candara\";\n"
133 "}\n"
134 "\n"
135 "QPushButton: hover{\n"
136 "    background-color: rgb(0,240,115);\n"
137 "}\n"
138 "QPushButton: pressed{\n"
139 "    background-color: rgb(0,200,96);\n"
140 "}\n"
141 """)
142     self.pushButton1.setObjectName("pushButton1")
143     self.verticalLayout_2.addWidget(self.pushButton1, 2, 0)
144     self.pushButton2 = QtWidgets.QPushButton(self.frame_4, clicked=lambda:
self.plot0nforbruk())
145     self.pushButton2.setStyleSheet("QPushButton{\n"
146 "    background-color: rgb(0,200,96);\n"
147 "    border: none;\n"
148 "    padding: 5px;\n"
149 "    color: rgb(205,230,255);\n"
150 "    border-radius: 5px;\n"
151 "    font: 75 14pt \"Candara\";\n"
152 "}\n"
153 "\n"
154 "QPushButton: hover{\n"
155 "    background-color: rgb(0,240,115);\n"
156 "}\n"

```

```

157 "QPushButton:pressed{\n"
158 "     background-color: rgb(0,200,96);\n"
159 "}\n"
160 "")
161     self.pushButton2.setObjectName("pushButton2")
162     self.verticalLayout_2.addWidget(self.pushButton2, 1, 0)
163     self.pushButton3 = QtWidgets.QPushButton(self.frame_4, clicked=lambda:
self.plotfortjenestesum())
164     self.pushButton3.setStyleSheet("QPushButton{\n"
165 "     background-color: rgb(0,200,96);\n"
166 "     border: none;\n"
167 "     padding: 5px;\n"
168 "     color: rgb(205,230,255);\n"
169 "     border-radius: 5px;\n"
170 "     font: 75 14pt \"Candara\";\n"
171 "}\n"
172 "\n"
173 "QPushButton: hover{\n"
174 "     background-color: rgb(0,240,115);\n"
175 "}\n"
176 "QPushButton:pressed{\n"
177 "     background-color: rgb(0,200,96);\n"
178 "}\n"
179 "")
180     self.pushButton3.setObjectName("pushButton3")
181     self.verticalLayout_2.addWidget(self.pushButton3, 0, 1)
182
183     self.pushButton4 = QtWidgets.QPushButton(self.frame_4, clicked=lambda:
self.plotstd())
184     self.pushButton4.setStyleSheet("QPushButton{\n"
185 "     background-color: rgb(0,200,96);\n"
186 "     border: none;\n"
187 "     padding: 5px;\n"
188 "     color: rgb(205,230,255);\n"
189 "     border-radius: 5px;\n"
190 "     font: 75 14pt \"Candara\";\n"
191 "}\n"
192 "\n"
193 "QPushButton: hover{\n"
194 "     background-color: rgb(0,240,115);\n"
195 "}\n"
196 "QPushButton:pressed{\n"
197 "     background-color: rgb(0,200,96);\n"
198 "}\n"
199 "")
200     self.pushButton4.setObjectName("pushButton4")
201     self.verticalLayout_2.addWidget(self.pushButton4, 1, 1)
202
203     self.pushButton5 = QtWidgets.QPushButton(self.frame_4, clicked=lambda:
self.plotnettleie())
204     self.pushButton5.setStyleSheet("QPushButton{\n"
205 "     background-color: rgb(0,200,96);\n"
206 "     border: none;\n"
207 "     padding: 5px;\n"

```

```

208 "    color: rgb(205,230,255);\n"
209 "    border-radius: 5px;\n"
210 "    font: 75 14pt \"Candara\";\n"
211 "}\n"
212 "\n"
213 "QPushButton: hover{\n"
214 "    background-color: rgb(0,240,115);\n"
215 "}\n"
216 "QPushButton: pressed{\n"
217 "    background-color: rgb(0,200,96);\n"
218 "}\n"
219 "" )
220     self.pushButton5.setObjectName("pushButton5")
221     #self.pushButton5.sizePolicy().setHorizontalStretch(1)
222     self.verticalLayout_2.addWidget(self.pushButton5, 2, 1)
223
224     self.pushButton6 = QtWidgets.QPushButton(self.frame_4, clicked=lambda:
self.plotproduksjon())
225     self.pushButton6.setStyleSheet("QPushButton{\n"
226 "    background-color: rgb(0,200,96);\n"
227 "    border: none;\n"
228 "    padding: 5px;\n"
229 "    color: rgb(205,230,255);\n"
230 "    border-radius: 5px;\n"
231 "    font: 75 14pt \"Candara\";\n"
232 "}\n"
233 "\n"
234 "QPushButton: hover{\n"
235 "    background-color: rgb(0,240,115);\n"
236 "}\n"
237 "QPushButton: pressed{\n"
238 "    background-color: rgb(0,200,96);\n"
239 "}\n"
240 "" )
241     self.pushButton6.setObjectName("pushButton6")
242     self.verticalLayout_2.addWidget(self.pushButton6, 4, 0)
243
244     self.pushButton7 = QtWidgets.QPushButton(self.frame_4, clicked=lambda:
self.plotsoc())
245     self.pushButton7.setStyleSheet("QPushButton{\n"
246 "    background-color: rgb(0,200,96);\n"
247 "    border: none;\n"
248 "    padding: 5px;\n"
249 "    color: rgb(205,230,255);\n"
250 "    border-radius: 5px;\n"
251 "    font: 75 14pt \"Candara\";\n"
252 "}\n"
253 "\n"
254 "QPushButton: hover{\n"
255 "    background-color: rgb(0,240,115);\n"
256 "}\n"
257 "QPushButton: pressed{\n"
258 "    background-color: rgb(0,200,96);\n"
259 "}\n"

```

```

260 ""
261     self.pushButton7.setObjectName("pushButton7")
262     self.verticalLayout_2.addWidget(self.pushButton7, 4, 1)
263
264     self.pushButton8 = QtWidgets.QPushButton(self.frame_4, clicked=lambda:
self.plottap())
265     self.pushButton8.setStyleSheet("QPushButton{\n"
266 "    background-color: rgb(0,200,96);\n"
267 "    border: none;\n"
268 "    padding: 5px;\n"
269 "    color: rgb(205,230,255);\n"
270 "    border-radius: 5px;\n"
271 "    font: 75 14pt \"Candara\";\n"
272 "}\n"
273 "\n"
274 "QPushButton: hover{\n"
275 "    background-color: rgb(0,240,115);\n"
276 "}\n"
277 "QPushButton: pressed{\n"
278 "    background-color: rgb(0,200,96);\n"
279 "}\n"
280 ""
281     self.pushButton8.setObjectName("pushButton8")
282     self.verticalLayout_2.addWidget(self.pushButton8, 3, 0)
283
284     self.pushButton9 = QtWidgets.QPushButton(self.frame_4, clicked=lambda:
self.plotdirekteforbruk())
285     self.pushButton9.setStyleSheet("QPushButton{\n"
286 "    background-color: rgb(0,200,96);\n"
287 "    border: none;\n"
288 "    padding: 5px;\n"
289 "    color: rgb(205,230,255);\n"
290 "    border-radius: 5px;\n"
291 "    font: 75 14pt \"Candara\";\n"
292 "}\n"
293 "\n"
294 "QPushButton: hover{\n"
295 "    background-color: rgb(0,240,115);\n"
296 "}\n"
297 "QPushButton: pressed{\n"
298 "    background-color: rgb(0,200,96);\n"
299 "}\n"
300 ""
301     self.pushButton9.setObjectName("pushButton9")
302     self.verticalLayout_2.addWidget(self.pushButton9, 3, 1)
303
304     self.pushButton10 = QtWidgets.QPushButton(self.frame_4, clicked=lambda:
self.plotoppladning())
305     self.pushButton10.setStyleSheet("QPushButton{\n"
306 "    background-color: rgb(0,200,96);\n"
307 "    border: none;\n"
308 "    padding: 5px;\n"
309 "    color: rgb(205,230,255);\n"
310 "    border-radius: 5px;\n"

```

```

311 "    font: 75 14pt \"Candara\";\n"
312 "}\n"
313 "\n"
314 "QPushButton: hover{\n"
315 "    background-color: rgb(0,240,115);\n"
316 "}\n"
317 "QPushButton: pressed{\n"
318 "    background-color: rgb(0,200,96);\n"
319 "}\n"
320 """)
321     self.pushButton10.setObjectName("pushButton10")
322     self.verticalLayout_2.addWidget(self.pushButton10, 5, 1)
323
324     self.pushButton11 = QtWidgets.QPushButton(self.frame_4, clicked=lambda:
self.plotsalg())
325     self.pushButton11.setStyleSheet("QPushButton{\n"
326 "    background-color: rgb(0,200,96);\n"
327 "    border: none;\n"
328 "    padding: 5px;\n"
329 "    color: rgb(205,230,255);\n"
330 "    border-radius: 5px;\n"
331 "    font: 75 14pt \"Candara\";\n"
332 "}\n"
333 "\n"
334 "QPushButton: hover{\n"
335 "    background-color: rgb(0,240,115);\n"
336 "}\n"
337 "QPushButton: pressed{\n"
338 "    background-color: rgb(0,200,96);\n"
339 "}\n"
340 """)
341     self.pushButton11.setObjectName("pushButton11")
342     self.verticalLayout_2.addWidget(self.pushButton11, 5, 0)
343
344
345     #self.checkbox = QtWidgets.QCheckBox("Solceller")
346     #self.checkbox.setObjectName("Checkbox1")
347     #self.checkbox.stateChanged.connect(lambda: self.solknapp(self.checkbox)
)
348     #self.verticalLayout_2.addWidget(self.checkbox, 6, 0)
349     #self.checkbox.setStyleSheet("QCheckBox::indicator { width:50px; height:
50px; }")
350     #self.checkbox.setStyleSheet("font-family: Candara; font-size: 14pt;")
351
352     self.label2 = QtWidgets.QLabel(self.frame_4)
353     self.label2.setObjectName("label2")
354     #sizePolicy = QtWidgets.QSizePolicy(QtWidgets.QSizePolicy.Maximum,
QtWidgets.QSizePolicy.Preferred)
355     self.label2.setSizePolicy(sizePolicy)
356     #self.label2.sizePolicy().setHorizontalStretch(1)
357     self.label2.setStyleSheet("QFrame{\n"
358 "    font: 75 14pt \"Candara\";\n"
359 "}")
360     self.verticalLayout_2.addWidget(self.label2, 6, 0)

```

```

361     #self.line = QtWidgets.QLineEdit(self.frame_4)
362     #self.line.setObjectName('Line1')
363     #self.line.setStyleSheet("font-size: 14pt;")
364     #self.line.setStyleSheet("font-size: 14pt;")
365     #sizePolicy = QtWidgets.QSizePolicy(QtWidgets.QSizePolicy.Maximum,
QtWidgets.QSizePolicy.Preferred)
366     #self.line.setSizePolicy(sizePolicy)
367     #self.line.sizePolicy().setHorizontalStretch(1)
368     #self.line.setText(f"{kapasitet_batteri}")
369     #self.line.returnPressed.connect(lambda: self.settbatteri())
370     #self.verticalLayout_2.addWidget(self.line, 8, 0)
371
372     self.combo = QtWidgets.QComboBox(self.frame_4)
373     self.combo.setObjectName('Combo1')
374     self.combo.setStyleSheet("font-size: 14pt;")
375     self.combo.addItem("13.5KW Enebolig Kr.sand 21")
376     self.combo.addItem("13.5KW Enebolig Solceller Kr.sand 21")
377     self.combo.addItem("36KW Enebolig Kr.sand 21")
378     self.combo.addItem("36KW Enebolig Solceller Kr.sand 21")
379     self.combo.addItem("13.5KW Rekkehus m/Elbil Kr.sand 21")
380     self.combo.addItem("13.5KW Rekkehus m/Elbil Solceller Kr.sand 21")
381     self.combo.addItem("36KW Rekkehus m/Elbil Kr.sand 21")
382     self.combo.addItem("36KW Rekkehus m/Elbil Solceller Kr.sand 21")
383     self.combo.addItem("13.5KW Jakob Kr.sand 21")
384     self.combo.addItem("13.5KW Jakob Solceller Kr.sand 21")
385     self.combo.addItem("36KW Jakob Kr.sand 21")
386     self.combo.addItem("36KW Jakob Solceller Kr.sand 21")
387     self.combo.addItem("13.5KW Enebolig Kr.sand 19")
388     self.combo.addItem("13.5KW Enebolig Solceller Kr.sand 19")
389     self.combo.addItem("36KW Enebolig Kr.sand 19")
390     self.combo.addItem("36KW Enebolig Solceller Kr.sand 19")
391     self.combo.addItem("13.5KW Rekkehus m/Elbil Kr.sand 19")
392     self.combo.addItem("13.5KW Rekkehus m/Elbil Solceller Kr.sand 19")
393     self.combo.addItem("36KW Rekkehus m/Elbil Kr.sand 19")
394     self.combo.addItem("36KW Rekkehus m/Elbil Solceller Kr.sand 19")
395     self.combo.addItem("13.5KW Enebolig DK1 21")
396     self.combo.addItem("13.5KW Enebolig Solceller DK1 21")
397     self.combo.addItem("13.5KW Enebolig Tr.heim 21")
398     self.combo.addItem("13.5KW Enebolig Solceller Tr.heim 21")
399     self.combo.addItem("13.5KW Hytte Solceller Tr.heim 21")
400     self.combo.addItem("13.5KW Enebolig Solceller Kr.sand 20")
401     self.combo.addItem("13.5KW Enebolig Solceller og Vind Kr.sand 21")
402     self.combo.addItem("13.5KW Hytte Solceller Kr.sand 21")
403
404     self.combo.currentIndexChanged.connect(self.rullegardin)
405     self.verticalLayout_2.addWidget(self.combo, 7, 0)
406
407     self.label3 = QtWidgets.QLabel(self.frame_4)
408     self.label3.setObjectName("label3")
409     self.label3.setSizePolicy(sizePolicy)
410     self.label3.setStyleSheet("QFrame{\n"
411 "    font: 75 14pt \"Candara\";\n"
412 "}")
413     self.verticalLayout_2.addWidget(self.label3, 6, 1)

```

```

414 self.combos = QtWidgets.QComboBox(self.frame_4)
415 self.combos.setObjectName('Combos')
416 self.combos.setStyleSheet("font-size: 14pt;")
417 self.combos.addItem("None")
418 self.combos.addItem("13.5KW Enebolig Kr.sand 21")
419 self.combos.addItem("13.5KW Enebolig Solceller Kr.sand 21")
420 self.combos.addItem("36KW Enebolig Kr.sand 21")
421 self.combos.addItem("36KW Enebolig Solceller Kr.sand 21")
422 self.combos.addItem("13.5KW Rekkehus m/Elbil Kr.sand 21")
423 self.combos.addItem("13.5KW Rekkehus m/Elbil Solceller Kr.sand 21")
424 self.combos.addItem("36KW Rekkehus m/Elbil Kr.sand 21")
425 self.combos.addItem("36KW Rekkehus m/Elbil Solceller Kr.sand 21")
426 self.combos.addItem("13.5KW Jakob Kr.sand 21")
427 self.combos.addItem("13.5KW Jakob Solceller Kr.sand 21")
428 self.combos.addItem("36KW Jakob Kr.sand 21")
429 self.combos.addItem("36KW Jakob Solceller Kr.sand 21")
430 self.combos.addItem("13.5KW Enebolig Kr.sand 19")
431 self.combos.addItem("13.5KW Enebolig Solceller Kr.sand 19")
432 self.combos.addItem("36KW Enebolig Kr.sand 19")
433 self.combos.addItem("36KW Enebolig Solceller Kr.sand 19")
434 self.combos.addItem("13.5KW Rekkehus m/Elbil Kr.sand 19")
435 self.combos.addItem("13.5KW Rekkehus m/Elbil Solceller Kr.sand 19")
436 self.combos.addItem("36KW Rekkehus m/Elbil Kr.sand 19")
437 self.combos.addItem("36KW Rekkehus m/Elbil Solceller Kr.sand 19")
438 self.combos.addItem("13.5KW Enebolig DK1 21")
439 self.combos.addItem("13.5KW Enebolig Solceller DK1 21")
440 self.combos.addItem("13.5KW Enebolig Tr.heim 21")
441 self.combos.addItem("13.5KW Enebolig Solceller Tr.heim 21")
442 self.combos.addItem("13.5KW Hytte Solceller Tr.heim 21")
443 self.combos.addItem("13.5KW Enebolig Solceller Kr.sand 20")
444 self.combos.addItem("13.5KW Enebolig Solceller og Vind Kr.sand 21")
445 self.combos.addItem("13.5KW Hytte Solceller Kr.sand 21")
446 self.combos.currentIndexChanged.connect(self.rullegardins)
447 self.verticalLayout_2.addWidget(self.combos, 7, 1)
448
449
450 self.combos2 = QtWidgets.QComboBox(self.frame_4)
451 self.combos2.setObjectName('Combos')
452 self.combos2.setStyleSheet("font-size: 14pt;")
453 self.combos2.addItem("None")
454 self.combos2.addItem("13.5KW Enebolig Kr.sand 21")
455 self.combos2.addItem("13.5KW Enebolig Solceller Kr.sand 21")
456 self.combos2.addItem("36KW Enebolig Kr.sand 21")
457 self.combos2.addItem("36KW Enebolig Solceller Kr.sand 21")
458 self.combos2.addItem("13.5KW Rekkehus m/Elbil Kr.sand 21")
459 self.combos2.addItem("13.5KW Rekkehus m/Elbil Solceller Kr.sand 21")
460 self.combos2.addItem("36KW Rekkehus m/Elbil Kr.sand 21")
461 self.combos2.addItem("36KW Rekkehus m/Elbil Solceller Kr.sand 21")
462 self.combos2.addItem("13.5KW Jakob Kr.sand 21")
463 self.combos2.addItem("13.5KW Jakob Solceller Kr.sand 21")
464 self.combos2.addItem("36KW Jakob Kr.sand 21")
465 self.combos2.addItem("36KW Jakob Solceller Kr.sand 21")
466 self.combos2.addItem("13.5KW Enebolig Kr.sand 19")
467 self.combos2.addItem("13.5KW Enebolig Solceller Kr.sand 19")

```



```

468 self.combos2.addItem("36KW Enebolig Kr.sand 19")
469 self.combos2.addItem("36KW Enebolig Solceller Kr.sand 19")
470 self.combos2.addItem("13.5KW Rekkehus m/Elbil Kr.sand 19")
471 self.combos2.addItem("13.5KW Rekkehus m/Elbil Solceller Kr.sand 19")
472 self.combos2.addItem("36KW Rekkehus m/Elbil Kr.sand 19")
473 self.combos2.addItem("36KW Rekkehus m/Elbil Solceller Kr.sand 19")
474 self.combos2.addItem("13.5KW Enebolig DK1 21")
475 self.combos2.addItem("13.5KW Enebolig Solceller DK1 21")
476 self.combos2.addItem("13.5KW Enebolig Tr.heim 21")
477 self.combos2.addItem("13.5KW Enebolig Solceller Tr.heim 21")
478 self.combos2.addItem("13.5KW Hytte Solceller Tr.heim 21")
479 self.combos2.addItem("13.5KW Enebolig Solceller Kr.sand 20")
480 self.combos2.addItem("13.5KW Enebolig Solceller og Vind Kr.sand 21")
481 self.combos2.addItem("13.5KW Hytte Solceller Kr.sand 21")
482 self.combos2.currentIndexChanged.connect(self.rullegardins2)
483 self.verticalLayout_2.addWidget(self.combos2, 8, 1)
484
485
486 self.verticalLayout.addWidget(self.frame_4, 0, QtCore.Qt.AlignBottom)
487 self.horizontalLayout.addWidget(self.frame)
488 MainWindow.setCentralWidget(self.centralwidget)
489 self.menubar = QtWidgets.QMenuBar(MainWindow)
490 self.menubar.setGeometry(QtCore.QRect(0, 0, 688, 22))
491 self.menubar.setObjectName("menubar")
492 MainWindow.setMenuBar(self.menubar)
493 self.statusbar = QtWidgets.QStatusBar(MainWindow)
494 self.statusbar.setObjectName("statusbar")
495 MainWindow.setStatusBar(self.statusbar)
496
497 self.retranslateUi(MainWindow)
498 QtCore.QMetaObject.connectSlotsByName(MainWindow)
499
500 def retranslateUi(self, MainWindow):
501     _translate = QtCore.QCoreApplication.translate
502     MainWindow.setWindowTitle(_translate("MainWindow", "ESS"))
503     self.label.setText(_translate("MainWindow", "Oversikt over ESS"))
504     self.label1.setText(_translate("MainWindow", f"Størrelse batteripakke {
505     kapasitet_batteri} KWh"
506     f"\nStart dato {alt_data['
507     Dato'].iloc[0]}")
508     f"\nSlutt dato {alt_data['
509     Dato'].iloc[-1]}")
510     f"\nTotalt forbruk {round(
511     sum(alt_data['Forbruk']), 1)} KWh"
512     f"\nGjennomsnittlig forbruk
513     per dag er {round(sum(alt_data['Forbruk'])/(len(alt_data.index)/24), 2)}
514     KWh"
515     f"\nGjennomsnittlig strø
516     mpris i perioden {round((alt_data['Strompris'].mean()),2)} EUR"
517     f"\nStandardavvik strømpris
518     {round(alt_data['Strompris'].std(),2)} EUR"
519     f"\nTotal oppladning på
520     Batteri {round(sum(alt_data['Flyt']),2)} KWh"
521     f"\nTotal direkte forbruk {

```

```

513     round(sum(alt_data['Flytforbruk']),2)} KWh"
514     solceller {round(sum(alt_data['Produksjon']*sol), 2)} KWh"
515     round(sum(alt_data['Salg']),2)} KWh"
516     round(sum(alt_data['PrisutenBatteri']), 2)} EUR"
517     round(sum(alt_data['PrismedBatteri']), 2)} EUR"
518     f"\nDifferanse {round(sum(alt_data['PrisutenBatteri']) - sum(alt_data['PrismedBatteri']), 2)} EUR i
519     besbarelse"
520     f"\nHvilket betyr en
521     reduksjon på {round(((sum(alt_data['PrisutenBatteri']) - sum(alt_data['
522     PrismedBatteri']))) / sum(alt_data['PrisutenBatteri'])) * 100,2)}%"
523     f"\nEffektivitet {round
524     ((1-((sum(alt_data['Flyt']) + sum(alt_data['Flytforbruk']) + (sum(alt_data['
525     Produksjon']*sol) - sum(alt_data['Forbruk']) - sum(alt_data['Salg']))/sum(
526     alt_data['Forbruk']))) * 100,1)}%"
527     f"\nGjennomsnittsinntjening
528     per dag {round(alt_data['AkkfortBatteriDiff'].mean()*24, 3)} EUR"
529     f"\nEst. besparelse over en
530     10 års periode {round((alt_data['AkkfortBatteriDiff'].mean()*24*365*10)}
531     EUR"
532     f"\nEst. profit over en 10
533     års periode {round(((alt_data['AkkfortBatteriDiff'].mean()*24*365*10) -
534     kapasitet_batteri*120.5) - 1200)} EUR"
535     f"\nSum nettleie differanse
536     {round(sum(alt_data['Nettleieuten']) - sum(alt_data['Nettleie']),2)} EUR"))
537     self.label2.setText(_translate("MainWindow", "Konfigurasjon"))
538     self.label3.setText(_translate("MainWindow", "Sammenlikning"))
539     self.pushButton.setText(_translate("MainWindow", "Oversikt"))
540     self.pushButton1.setText(_translate("MainWindow", "Strømpris"))
541     self.pushButton2.setText(_translate("MainWindow", "Forbruk"))
542     self.pushButton3.setText(_translate("MainWindow", "Akkumulert besbarelse
543     "))
544     self.pushButton4.setText(_translate("MainWindow", "Standardavvik Strø
545     mpris mot derivert fortjenste"))
546     self.pushButton5.setText(_translate("MainWindow", "Variabel nettleie
547     differanse"))
548     self.pushButton6.setText(_translate("MainWindow", "Produksjon fra
549     solceller"))
550     self.pushButton7.setText(_translate("MainWindow", "SOC mot strømpris"))
551     self.pushButton8.setText(_translate("MainWindow", "Tapinverter"))
552     self.pushButton9.setText(_translate("MainWindow", "Belastning på strø
553     mnettet"))
554     self.pushButton10.setText(_translate("MainWindow", "Oppladning og
555     utladning"))
556     self.pushButton11.setText(_translate("MainWindow", "Salg av strøm"))
557
558     def plotOnCanvas(self):
559         ##clear the canvas
560         self.figure.clear()

```

```

543     plt.plot(alt_data.index, alt_data['Strompris'], color='red', linewidth
=2)
544     plt.plot(alt_data.index, alt_data['Strompris'].rolling(window=24).mean()
, color='blue', linewidth=4)
545
546     plt.xlabel('Dato')
547     plt.ylabel('Strømpris EUR/MWh')
548     plt.title('Strømpris')
549     plt.xticks(np.arange(0, len(alt_data['Dato']), 900), alt_data['Dato'].
iloc[:,900], rotation='horizontal')
550     plt.savefig('strompris')
551     #refresh canvas
552     self.canvas.draw()
553
554     def plot0nforbruk(self):
555         ##clear the canvas
556         self.figure.clear()
557
558         if sammenlikning == 1 and sammenlikning2 == 1:
559             plt.plot(alt_data.index, alt_data['Forbruk'], color='brown',
linewidth=2)
560             plt.plot(alt_data.index, alt_data['Forbruk'].rolling(window=24).mean(),
color='blue', linewidth=4, label=navnaltdata)
561             if sammenlikning != 1:
562                 if sum(alt_data['Forbruk']) != sum(alt_datas['Forbruk']):
563                     plt.plot(alt_data.index, alt_datas['Forbruk'].rolling(window=24)
.mean(), color='orange', linewidth=4, label=navnaltdatas)
564             if sammenlikning2 != 1:
565                 if sum(alt_data['Forbruk']) != sum(alt_datas2['Forbruk']) or sum(
alt_data['Forbruk']) != sum(alt_datas['Forbruk']):
566                     plt.plot(alt_datas2.index, alt_datas2['Forbruk'].rolling(window
=24).mean(), color='black', linewidth=4, label=navnaltdatas2)
567
568
569     plt.legend()
570     plt.xlabel('Dato')
571     plt.ylabel("Forbruk i KWh")
572     plt.title('Forbruk')
573     plt.xticks(np.arange(0, len(alt_data['Dato']), 900), alt_data['Dato'].
iloc[:,900], rotation='horizontal')
574     plt.savefig('forbruk')
575     #refresh canvas
576     self.canvas.draw()
577
578     def plot0noversikt(self):
579         ##clear the canvas
580         self.figure.clear()
581
582
583         ax = self.canvas.figure.subplots(2)
584
585         if sammenlikning == 1 and sammenlikning2 == 1:
586             ax[0].plot(alt_data.index, alt_data['Strompris'], color='red',
linewidth=2.0)

```

```

587     ax[0].plot(alt_data.index, alt_data['Strompris'].rolling(window=24).mean
(), color='blue', linewidth=4.0, label=navnaltdata)
588     ax[1].plot(alt_data.index, alt_data['AkkfortjenesteBatteri'], color='
green', linewidth=2.0)
589     ax[1].plot(alt_data.index, alt_data['AkkfortjenesteBatteri'].rolling(
window=24).mean(), color='blue', linewidth=4.0, label=navnaltdata)
590
591     if sammenlikning != 1:
592         if sum(alt_data['Strompris']) != sum(alt_datas['Strompris']):
593             ax[0].plot(alt_datas.index, alt_datas['Strompris'].rolling(
window=24).mean(), color='orange', linewidth=4.0, label=navnaltdatas)
594
595             ax[1].plot(alt_datas.index, alt_datas['AkkfortjenesteBatteri'].
rolling(window=24).mean(), color='orange', linewidth=4.0, label=navnaltdatas
)
596
597     if sammenlikning2 != 1:
598         if sum(alt_data['Strompris']) != sum(alt_datas2['Strompris']) and
sum(alt_data['Strompris']) != sum(alt_datas['Strompris']):
599             ax[0].plot(alt_datas2.index, alt_datas2['Strompris'].rolling(
window=24).mean(), color='black', linewidth=4.0, label=navnaltdatas2)
600
601             ax[1].plot(alt_datas2.index, alt_datas2['AkkfortjenesteBatteri'].
rolling(window=24).mean(), color='black', linewidth=4.0, label=navnaltdatas2
)
602
603     ax[0].legend()
604     ax[1].legend()
605     ax[0].set_ylabel('Strømpris i EUR/MWh')
606     ax[0].set_title('Oversikt')
607     ax[1].set_ylabel('Besbarelse i EUR')
608     ax[1].set_xlabel('Dato')
609
610
611     for ax in ax.flat:
612         ax.set_xticks(np.arange(0, len(alt_data['Dato']), 900), alt_data['
Dato'].iloc[:,900], rotation='horizontal')
613
614     plt.savefig('oversikt')
615     #refresh canvas
616     self.canvas.draw()
617
618     def plotfortjenestesum(self):
619         ##clear the canvas
620         self.figure.clear()
621
622         plt.plot(alt_data.index, alt_data['AkkfortjenesteBatteri'], color='green
', linewidth=2)
623         plt.plot(alt_data.index, alt_data['AkkfortjenesteBatteri'].rolling(
window=24).mean(), color='blue', linewidth=4, label=navnaltdata)
624
625
626     plt.xlabel('Dato')
627     plt.ylabel("Fortjeneste i EUR")

```

```

628     plt.title('Akkumulert Besbarelse')
629     plt.xticks(np.arange(0, len(alt_data['Dato']), 900), alt_data['Dato'].
iloc[:,900], rotation='horizontal')
630
631     plt.savefig('besbarelse')
632     #refresh canvas
633     self.canvas.draw()
634
635
636     def plotstd(self):
637         self.figure.clear()
638
639         alt_data['AvvikMA120'] = alt_data['Strompris'] - alt_data['Strompris'].
rolling(window=200).mean()
640         alt_data['StromprisStd'] = (alt_data['Strompris'] - alt_data['Strompris '
].mean())/alt_data['Strompris'].std()
641
642         ax = self.canvas.figure.subplots()
643
644         lns1 = ax.plot(alt_data.index, abs(alt_data['StromprisStd']).rolling(
window=24).mean()), linewidth=2.0, label='STD Strømpris')
645
646
647         ax2 = ax.twinx() # instantiate a second axes that shares the same x-
axis
648
649         lns2 = ax2.plot(alt_data.index, alt_data['AkkfortBatteriDiff'].rolling(
window=24).mean(), color='orange', linewidth=2, label='Stigningsrate
Fortjenste')
650
651
652         ax.set_ylabel('Standardavvik strømpris')
653         ax2.set_ylabel("Stigningsrate besbarelse")
654         ax.set_xlabel('Dato')
655         align_yaxis_np(ax, ax2)
656
657         lns = lns1+lns2
658         labs = [l.get_label() for l in lns]
659         ax.legend(lns, labs, loc=0)
660         ax.set_title('STD strømpris mot deriverte av besparelse')
661         ax.set_xticks(np.arange(0, len(alt_data['Dato']), 900), alt_data['Dato '
].iloc[:,900], rotation='horizontal')
662
663         plt.savefig('std')
664         #refresh canvas
665         self.canvas.draw()
666     def plotnettleie(self):
667         self.figure.clear()
668
669         for t in range(len(alt_data.index)):
670             alt_data.at[t, 'Akkfortjenestenettleie'] = sum(alt_data['
Nettleieuten'].head(t+1)) - sum(alt_data['Nettleie'].head(t+1))
671         plt.plot(alt_data.index, alt_data['Akkfortjenestenettleie'], color='red '
)

```

```

672     plt.plot(alt_data.index, alt_data['Akkfortjenestenettleie'].rolling(
window=24).mean(), color='blue', linewidth=4, label=navnaltdata)
673
674     if sammenlikning != 1:
675         for t in range(len(alt_data.index)):
676             alt_datas.at[t, 'Akkfortjenestenettleie'] = sum(alt_datas['
Nettleieuten'].head(t+1)) - sum(alt_datas['Nettleie'].head(t+1))
677             if sum(alt_data['Akkfortjenestenettleie']) != sum(alt_datas['
Akkfortjenestenettleie']):
678                 plt.plot(alt_datas.index, alt_datas['Akkfortjenestenettleie'].
rolling(window=24).mean(), color='orange', linewidth=4, label=navnaltdatas)
679
680         if sammenlikning2 != 1:
681             for t in range(len(alt_data.index)):
682                 alt_datas2.at[t, 'Akkfortjenestenettleie'] = sum(alt_datas2['
Nettleieuten'].head(t+1)) - sum(alt_datas2['Nettleie'].head(t+1))
683                 if sum(alt_data['Akkfortjenestenettleie']) != sum(alt_datas['
Akkfortjenestenettleie']) and sum(alt_data['Akkfortjenestenettleie']) != sum
(alt_datas2['Akkfortjenestenettleie']):
684                     plt.plot(alt_datas.index, alt_datas2['Akkfortjenestenettleie'].
rolling(window=24).mean(), color='black', linewidth=4, label=navnaltdatas2)
685
686
687     plt.legend()
688     plt.xlabel('Dato')
689     plt.ylabel("Differanse Nettleie i EUR")
690     plt.title('Differanse nettleie')
691     plt.xticks(np.arange(0, len(alt_data['Dato']), 900), alt_data['Dato'].
iloc[:,900], rotation='horizontal')
692
693     plt.savefig('nettleie')
694     self.canvas.draw()
695
696     def plottap(self):
697         self.figure.clear()
698         plt.plot(alt_data.index, alt_data['TapInverterforbruk'].rolling(window
=24).mean(), color='blue', linewidth=4, label='Tap utladning')
699         plt.plot(alt_data.index, alt_data['TapInverterbatteri'].rolling(window
=24).mean(), color='black', linewidth=4, label='Tap oppladning')
700         plt.xlabel('Dato')
701         plt.ylabel("Tap inverter i kW")
702         plt.title('Tap gjennom inverter')
703         plt.legend()
704         plt.xticks(np.arange(0, len(alt_data['Dato']), 900), alt_data['Dato'].
iloc[:,900], rotation='horizontal')
705
706         plt.savefig('tap')
707         self.canvas.draw()
708
709     def plotdirekteforbruk(self):
710         self.figure.clear()
711
712
713     conprod = pd.read_excel(r"C:\Users\gonna\pythonProject\betingelser\

```

```

ProductionConsumption-2021.xlsx")
714     ax = self.canvas.figure.subplots()
715     lns1 = ax.plot(range(150), (alt_data['Forbruk'].iloc[3000:3150] -
alt_data['Produksjon'].iloc[3000:3150]*sol), color='black', label='Trek fra
strømnettet uten batteri', linewidth=2, alpha=0.5)
716     lns2 = ax.plot(range(150), (alt_data['Flytforbruk'].iloc[3000:3150] +
alt_data['Flyt'].iloc[3000:3150] - alt_data['Salg'].iloc[3000:3150]), color=
'green', label='Trek fra strømnett med batteri', linewidth=2, alpha=0.5)
717
718     ax2 = ax.twinx()
719
720     lns3 = ax2.plot(range(150), conprod['Production'].iloc[3000:3150], label
='Produksjon i hele Norge')
721     lns4 = ax2.plot(range(150), conprod['Consumption'].iloc[3000:3150],
label='Forbruk i hele Norge')
722
723     ax.set_ylabel('Effekt i KW')
724     ax2.set_ylabel("Effekt i MW")
725     ax.set_xlabel('Dato')
726
727     lns = lns1+lns2+lns3+lns4
728     labs = [l.get_label() for l in lns]
729     ax.legend(lns, labs, loc=0)
730     ax.set_title('Belastning på nettet')
731     ax.set_xticks(np.arange(0, 150, 24), alt_data['Dato'].iloc
[3000:3150:24], rotation='horizontal')
732
733     plt.savefig('direkteforbruk')
734     self.canvas.draw()
735
736     def plotproduksjon(self):
737
738         if sol == 0:
739             self.figure.clear()
740             self.canvas.draw()
741         elif sol == 1:
742             self.figure.clear()
743             plt.plot(alt_data.index, alt_data['Produksjon'], color='red')
744             plt.plot(alt_data.index, alt_data['Produksjon'].rolling(window=24).
mean(), color='blue', linewidth=4)
745             plt.xlabel('Dato')
746             plt.ylabel("Produksjon i KWh")
747             plt.title('Produksjon fra Solceller')
748             plt.xticks(np.arange(0, len(alt_data['Dato']), 900), alt_data['Dato'
].iloc[:900], rotation='horizontal')
749
750             plt.savefig('produksjon')
751             self.canvas.draw()
752
753         def plotoppladning(self):
754             self.figure.clear()
755             plt.plot(alt_data.index, alt_data['Flyt'], color='red', label='
Oppladning')
756             plt.plot(alt_data.index, (alt_data['Forbruk'] - alt_data['Flytforbruk'])

```

```

*-1, color='black', label='Utladning')
757     plt.xlabel('Dato')
758     plt.ylabel("KW")
759     plt.title('Oppladning og utladning')
760     plt.legend()
761     plt.xticks(np.arange(0, len(alt_data['Dato']), 900), alt_data['Dato'].
iloc[:,900], rotation='horizontal')

762
763     plt.savefig('oppladning')
764     self.canvas.draw()
765
766     def plotsalg(self):
767         self.figure.clear()
768         plt.plot(alt_data.index, alt_data['Salg'], color='green', label='
Oppladning')
769         plt.xlabel('Dato')
770         plt.ylabel("KW")
771         plt.title('Salg fra batteri')
772         plt.xticks(np.arange(0, len(alt_data['Dato']), 900), alt_data['Dato'].
iloc[:,900], rotation='horizontal')

773
774         plt.savefig('salg')
775         self.canvas.draw()
776
777
778     def plotsoc(self):
779         self.figure.clear()
780         #ax = self.canvas.figure.subplots()
781         #for t in range(200):
782             #     if t <= 95:
783                 #         ax.cla()
784                 #         ax.plot(np.arange(0, t, 1), alt_data['Batteri'].iloc[0:t],
color='green')
785             #     ax.plot(np.arange(0, t, 1), alt_data['Strompris'].iloc[0:t],
color='red')
786             #         plt.xlabel('Tid(Time)')
787             #         plt.ylabel("Strømpris og SOC")
788             #         ax.set_ylim([0, 100])
789             #         self.canvas.draw()
790             #         self.canvas.flush_events()
791             #     else:
792                 #         ax.cla()
793                 #         ax.plot(np.arange(t-95, t, 1), alt_data['Batteri'].iloc[t-95:t
], color='green')
794             #         ax.plot(np.arange(t-95, t, 1), alt_data['Strompris'].iloc[t-95:
t], color='red')
795             #             plt.xlabel('Tid(Time)')
796             #             plt.ylabel("Strømpris og SOC")
797             #             ax.set_ylim([0, 100])
798             #             self.canvas.draw()
799             #             self.canvas.flush_events()
800
801         #self.figure.clear()
802         ax = self.canvas.figure.subplots()

```



```

803     ax.plot(np.arange(0, 200, 1), alt_data['Batteri'].head(200), color='
green')
804     ax.plot(np.arange(0, 200, 1), alt_data['Strompris'].head(200), color='
red')
805     plt.xlabel('Tid(Time)')
806     plt.ylabel("Strømpris(EUR) og SOC(%)")
807     plt.title('SOC mot strømpris for første 200 timer')
808     ax.set_ylim(0, 100)
809     plt.xticks(np.arange(0, 200, 24), alt_data['Dato'].iloc[0:200:24],
rotation='horizontal')
810
811     def update(val):
812         pos = round(slider_position.val)
813         ax.cla()
814         lns1 = ax.plot(np.arange(pos, pos+200, 1), alt_data['Batteri'].iloc[
pos:pos+200], color='green', label="SOC")
815         lns2 = ax.plot(np.arange(pos, pos+200, 1), alt_data['Strompris'].
iloc[pos:pos+200], color='red', label="Strømpris")
816         lns = lns1+lns2
817         labs = [l.get_label() for l in lns]
818         ax.legend(lns, labs, loc=0)
819         ax.set_ylim(0,100)
820         ax.set_xticks(np.arange(pos, pos+200, 24), alt_data['Dato'].iloc[pos
:pos+200:24], rotation='horizontal')
821         ax.set_title(f'Strømpris(EUR) og SOC(%) i perioden {alt_data["Dato
"].iloc[pos]} til {alt_data["Dato"].iloc[pos+200]}')
822
823         plt.savefig('soc')
824         self.canvas.draw_idle()
825         axis_position = plt.axes([0.2, 0.11, 0.65, 0.03], facecolor='white')
826         slider_position = Slider(axis_position, 'Pos', 1, len(alt_data.index)
-200)
827         slider_position.on_changed(update)
828
829         self.canvas.draw()
830
831
832     def solknapp(self, b):
833         global sol
834         if b.isChecked() == True:
835             print(b.text()+" is selected")
836             sol = 1
837             milp(alt_data['StromprisKWhMVA'], kapasitet_batteri, alt_data["
Forbruk"], alt_data['Produksjon'], sol)
838             self.retranslateUi(MainWindow)
839             self.figure.clear()
840             self.canvas.draw()
841
842         else:
843             print(b.text()+" is deselected")
844             sol = 0
845             milp(alt_data['StromprisKWhMVA'], kapasitet_batteri, alt_data["
Forbruk"], alt_data['Produksjon'], sol)
846             self.retranslateUi(MainWindow)

```

```

847         self.figure.clear()
848         self.canvas.draw()
849
850     def settbatteri(self):
851         global kapasitet_batteri
852         input = self.line.text()
853         input = int(input)
854         kapasitet_batteri = input
855         milp(alt_data['StromprisKWhMVA'], kapasitet_batteri, alt_data["Forbruk"
856 ], alt_data['Produksjon'], sol)
857         self.retranslateUi(MainWindow)
858         self.figure.clear()
859         self.canvas.draw()
860
861     def rullegardin(self, i):
862         global alt_data
863         global sol
864         global navnaltdata
865         global kapasitet_batteri
866         if i == 0:
867             alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\
868 betingelser\ferdig\Krsand21\13KW_EB_uS_Krsand21.xlsx")
869             sol = 0
870             kapasitet_batteri = 13.5
871             navnaltdata = self.combo.itemText(i)
872         elif i == 1:
873             alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\
874 betingelser\ferdig\Krsand21\13KW_EB_S_Krsand21.xlsx")
875             sol = 1
876             kapasitet_batteri = 13.5
877             navnaltdata = self.combo.itemText(i)
878         elif i == 2:
879             alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\
880 betingelser\ferdig\Krsand21\36KW_EB_uS_Krsand21.xlsx")
881             sol = 0
882             kapasitet_batteri = 36
883             navnaltdata = self.combo.itemText(i)
884         elif i == 3:
885             alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\
886 betingelser\ferdig\Krsand21\36KW_EB_S_Krsand21.xlsx")
887             sol = 1
888             kapasitet_batteri = 36
889             navnaltdata = self.combo.itemText(i)
890         elif i == 4:
891             alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\
892 betingelser\ferdig\Krsand21\13KW_RH_uS_Krsand21.xlsx")
893             sol = 0
894             kapasitet_batteri = 13.5
895             navnaltdata = self.combo.itemText(i)
896         elif i == 5:
897             alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\
898 betingelser\ferdig\Krsand21\13KW_RH_S_Krsand21.xlsx")
899             sol = 1
900             kapasitet_batteri = 13.5

```

```

894         navnaltdata = self.combo.itemText(i)
895     elif i == 6:
896         alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand21\36KW_RH_uS_Krsand21.xlsx")
897         sol = 0
898         kapasitet_batteri = 36
899         navnaltdata = self.combo.itemText(i)
900     elif i == 7:
901         alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand21\36KW_RH_S_Krsand21.xlsx")
902         sol = 1
903         kapasitet_batteri = 36
904         navnaltdata = self.combo.itemText(i)
905     elif i == 8:
906         alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand21\13KW_JA_uS_Krsand21.xlsx")
907         sol = 0
908         kapasitet_batteri = 13.5
909         navnaltdata = self.combo.itemText(i)
910     elif i == 9:
911         alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand21\13KW_JA_S_Krsand21.xlsx")
912         sol = 1
913         kapasitet_batteri = 13.5
914         navnaltdata = self.combo.itemText(i)
915     elif i == 10:
916         alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand21\36KW_JA_uS_Krsand21.xlsx")
917         sol = 0
918         kapasitet_batteri = 36
919         navnaltdata = self.combo.itemText(i)
920     elif i == 11:
921         alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand21\36KW_JA_S_Krsand21.xlsx")
922         sol = 1
923         kapasitet_batteri = 36
924     elif i == 12:
925         alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand19\13KW_EB_uS_Krsand19.xlsx")
926         sol = 0
927         kapasitet_batteri = 13.5
928         navnaltdata = self.combo.itemText(i)
929     elif i == 13:
930         alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand19\13KW_EB_S_Krsand19.xlsx")
931         sol = 1
932         kapasitet_batteri = 13.5
933         navnaltdata = self.combo.itemText(i)
934     elif i == 14:
935         alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand19\36KW_EB_uS_Krsand19.xlsx")
936         sol = 0
937         kapasitet_batteri = 36
938         navnaltdata = self.combo.itemText(i)

```

```

939         elif i == 15:
940             alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand19\36KW_EB_S_Krsand19.xlsx")
941             sol = 1
942             kapasitet_batteri = 36
943             navnaltdata = self.combo.itemText(i)
944         elif i == 16:
945             alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand19\13KW_RH_uS_Krsand19.xlsx")
946             sol = 0
947             kapasitet_batteri = 13.5
948             navnaltdata = self.combo.itemText(i)
949         elif i == 17:
950             alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand19\13KW_RH_S_Krsand19.xlsx")
951             sol = 1
952             kapasitet_batteri = 13.5
953             navnaltdata = self.combo.itemText(i)
954         elif i == 18:
955             alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand19\36KW_RH_uS_Krsand19.xlsx")
956             sol = 0
957             kapasitet_batteri = 36
958             navnaltdata = self.combo.itemText(i)
959         elif i == 19:
960             alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand19\36KW_RH_S_Krsand19.xlsx")
961             sol = 1
962             kapasitet_batteri = 36
963             navnaltdata = self.combo.itemText(i)
964         elif i == 20:
965             alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\DK1_21\13KW_EB_uS_DK121.xlsx")
966             sol = 0
967             kapasitet_batteri = 13.5
968             navnaltdata = self.combo.itemText(i)
969         elif i == 21:
970             alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\DK1_21\13KW_EB_S_DK121.xlsx")
971             sol = 1
972             kapasitet_batteri = 13.5
973             navnaltdata = self.combo.itemText(i)
974         elif i == 22:
975             alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\N02_21\13KW_EB_uS_N0221.xlsx")
976             sol = 0
977             kapasitet_batteri = 13.5
978             navnaltdata = self.combo.itemText(i)
979         elif i == 23:
980             alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\N02_21\13KW_EB_S_N0221.xlsx")
981             sol = 1
982             kapasitet_batteri = 13.5
983             navnaltdata = self.combo.itemText(i)

```

```

984         elif i == 24:
985             alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\N02_21\13KW_R_S_N021.xlsx")
986             sol = 1
987             kapasitet_batteri = 13.5
988             navnaltdata = self.combo.itemText(i)
989         elif i == 25:
990             alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand20\13KW_EB_S_Kr20.xlsx")
991             sol = 1
992             kapasitet_batteri = 13.5
993             navnaltdata = self.combo.itemText(i)
994         elif i == 26:
995             alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\13KW_EB_SW_Kr21.xlsx")
996             sol = 1
997             kapasitet_batteri = 13.5
998             navnaltdata = self.combo.itemText(i)
999         elif i == 27:
1000            alt_data = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand21\13KW_R_S_Kr21.xlsx")
1001            sol = 1
1002            kapasitet_batteri = 13.5
1003            navnaltdata = self.combo.itemText(i)
1004
1005            self.retranslateUi(MainWindow)
1006            self.figure.clear()
1007            self.canvas.draw()
1008            print(self.combo.itemText(i))
1009
1010    def rullegardins(self, i):
1011        global alt_datas
1012        global sol
1013        global sammenlikning
1014        global navnaltdatas
1015        global kapasitet_batteri
1016        if i == 0:
1017            sammenlikning = 1
1018        else:
1019            sammenlikning = 0
1020        if i == 1:
1021            alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand21\13KW_EB_uS_Krsand21.xlsx")
1022            sol = 0
1023            kapasitet_batteri = 13.5
1024            navnaltdatas = self.combos.itemText(i)
1025        elif i == 2:
1026            alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand21\13KW_EB_S_Krsand21.xlsx")
1027            sol = 1
1028            kapasitet_batteri = 13.5
1029            navnaltdatas = self.combos.itemText(i)
1030        elif i == 3:
1031            alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\

```

```

1032         betingelser\ferdig\Krsand21\36KW_EB_uS_Krsand21.xlsx")
1033             sol = 0
1034             kapasitas_batteri = 36
1035             navnaltdatas = self.combos.itemText(i)
1036         elif i == 4:
1037             alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\
1038 betingelser\ferdig\Krsand21\36KW_EB_S_Krsand21.xlsx")
1039             sol = 1
1040             kapasitas_batteri = 36
1041             navnaltdatas = self.combos.itemText(i)
1042         elif i == 5:
1043             alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\
1044 betingelser\ferdig\Krsand21\13KW_RH_uS_Krsand21.xlsx")
1045             sol = 0
1046             kapasitas_batteri = 13.5
1047             navnaltdatas = self.combos.itemText(i)
1048         elif i == 6:
1049             alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\
1050 betingelser\ferdig\Krsand21\13KW_RH_S_Krsand21.xlsx")
1051             sol = 1
1052             kapasitas_batteri = 13.5
1053             navnaltdatas = self.combos.itemText(i)
1054         elif i == 7:
1055             alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\
1056 betingelser\ferdig\Krsand21\36KW_RH_uS_Krsand21.xlsx")
1057             sol = 0
1058             kapasitas_batteri = 36
1059             navnaltdatas = self.combos.itemText(i)
1060         elif i == 8:
1061             alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\
1062 betingelser\ferdig\Krsand21\36KW_RH_S_Krsand21.xlsx")
1063             sol = 1
1064             kapasitas_batteri = 36
1065             navnaltdatas = self.combos.itemText(i)
1066         elif i == 9:
1067             alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\
1068 betingelser\ferdig\Krsand21\13KW_JA_uS_Krsand21.xlsx")
1069             sol = 0
1070             kapasitas_batteri = 13.5
1071             navnaltdatas = self.combos.itemText(i)
1072         elif i == 10:
1073             alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\
1074 betingelser\ferdig\Krsand21\13KW_JA_S_Krsand21.xlsx")
1075             sol = 1
1076             kapasitas_batteri = 13.5
1077             navnaltdatas = self.combos.itemText(i)
1078         elif i == 11:
1079             alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\
1080 betingelser\ferdig\Krsand21\36KW_JA_uS_Krsand21.xlsx")
1081             sol = 0
1082             kapasitas_batteri = 36
1083             navnaltdatas = self.combos.itemText(i)
1084         elif i == 12:
1085             alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\

```

```

1077         betingelser\ferdig\Krsand21\36KW_JA_S_Krsand21.xlsx")
1078             sol = 1
1079             kapasitas_batteri = 36
1080             navnaltdatas = self.combos.itemText(i)
1081         elif i == 13:
1082             alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\
1083 betingelser\ferdig\Krsand19\13KW_EB_uS_Krsand19.xlsx")
1084             sol = 0
1085             kapasitas_batteri = 13.5
1086             navnaltdatas = self.combos.itemText(i)
1087         elif i == 14:
1088             alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\
1089 betingelser\ferdig\Krsand19\13KW_EB_S_Krsand19.xlsx")
1090             sol = 1
1091             kapasitas_batteri = 13.5
1092             navnaltdatas = self.combos.itemText(i)
1093         elif i == 15:
1094             alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\
1095 betingelser\ferdig\Krsand19\36KW_EB_uS_Krsand19.xlsx")
1096             sol = 0
1097             kapasitas_batteri = 36
1098             navnaltdatas = self.combos.itemText(i)
1099         elif i == 16:
1100             alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\
1101 betingelser\ferdig\Krsand19\36KW_EB_S_Krsand19.xlsx")
1102             sol = 1
1103             kapasitas_batteri = 36
1104             navnaltdatas = self.combos.itemText(i)
1105         elif i == 17:
1106             alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\
1107 betingelser\ferdig\Krsand19\13KW_RH_uS_Krsand19.xlsx")
1108             sol = 0
1109             kapasitas_batteri = 13.5
1110             navnaltdatas = self.combos.itemText(i)
1111         elif i == 18:
1112             alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\
1113 betingelser\ferdig\Krsand19\13KW_RH_S_Krsand19.xlsx")
1114             sol = 1
1115             kapasitas_batteri = 13.5
1116             navnaltdatas = self.combos.itemText(i)
1117         elif i == 19:
1118             alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\
1119 betingelser\ferdig\Krsand19\36KW_RH_uS_Krsand19.xlsx")
1120             sol = 0
1121             kapasitas_batteri = 36
1122             navnaltdatas = self.combos.itemText(i)
1123         elif i == 20:
1124             alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\
1125 betingelser\ferdig\Krsand19\36KW_RH_S_Krsand19.xlsx")
1126             sol = 1
1127             kapasitas_batteri = 36
1128             navnaltdatas = self.combos.itemText(i)
1129         elif i == 21:
1130             alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\

```

```

1122         betingelser\ferdig\DK1_21\13KW_EB_uS_DK121.xlsx")
1123             sol = 0
1124             kapasitet_batteri = 13.5
1125             navnaltdatas = self.combos.itemText(i)
1126         elif i == 22:
1127             alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\
1128         betingelser\ferdig\DK1_21\13KW_EB_S_DK121.xlsx")
1129             sol = 1
1130             kapasitet_batteri = 13.5
1131             navnaltdatas = self.combos.itemText(i)
1132         elif i == 23:
1133             alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\
1134         betingelser\ferdig\N02_21\13KW_EB_uS_N0221.xlsx")
1135             sol = 0
1136             kapasitet_batteri = 13.5
1137             navnaltdatas = self.combos.itemText(i)
1138         elif i == 24:
1139             alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\
1140         betingelser\ferdig\N02_21\13KW_EB_S_N0221.xlsx")
1141             sol = 0
1142             kapasitet_batteri = 13.5
1143             navnaltdatas = self.combos.itemText(i)
1144         elif i == 25:
1145             alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\
1146         betingelser\ferdig\N02_21\13KW_R_S_N021.xlsx")
1147             sol = 1
1148             kapasitet_batteri = 13.5
1149             navnaltdatas = self.combos.itemText(i)
1150         elif i == 26:
1151             alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\
1152         betingelser\ferdig\Krsand20\13KW_EB_S_Kr20.xlsx")
1153             sol = 1
1154             kapasitet_batteri = 13.5
1155             navnaltdatas = self.combos.itemText(i)
1156         elif i == 27:
1157             alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\
1158         betingelser\13KW_EB_SW_Kr21.xlsx")
1159             sol = 1
1160             kapasitet_batteri = 13.5
1161             navnaltdatas = self.combos.itemText(i)
1162         elif i == 28:
1163             alt_datas = pd.read_excel(r"C:\Users\gonna\pythonProject\
1164         betingelser\ferdig\Krsand21\13KW_R_S_Kr21.xlsx")
1165             sol = 1
1166             kapasitet_batteri = 13.5
1167             navnaltdatas = self.combos.itemText(i)
1168
1169         self.retranslateUi(MainWindow)
1170         self.figure.clear()
1171         self.canvas.draw()
1172         print(self.combo.itemText(i))
1173
1174     def rullegardins2(self, i):
1175         global alt_datas2

```



```

1168     global sol
1169     global sammenlikning2
1170     global navnaltdatas2
1171     if i == 0:
1172         sammenlikning2 = 1
1173     else:
1174         sammenlikning2 = 0
1175     if i == 1:
1176         alt_datas2 = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand21\13KW_EB_uS_Krsand21.xlsx")
1177         sol = 0
1178         navnaltdatas2 = self.combos2.itemText(i)
1179     elif i == 2:
1180         alt_datas2 = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand21\13KW_EB_S_Krsand21.xlsx")
1181         sol = 1
1182         navnaltdatas2 = self.combos2.itemText(i)
1183     elif i == 3:
1184         alt_datas2 = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand21\36KW_EB_uS_Krsand21.xlsx")
1185         sol = 0
1186         navnaltdatas2 = self.combos2.itemText(i)
1187     elif i == 4:
1188         alt_datas2 = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand21\36KW_EB_S_Krsand21.xlsx")
1189         sol = 1
1190         navnaltdatas2 = self.combos2.itemText(i)
1191     elif i == 5:
1192         alt_datas2 = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand21\13KW_RH_uS_Krsand21.xlsx")
1193         sol = 0
1194         navnaltdatas2 = self.combos2.itemText(i)
1195     elif i == 6:
1196         alt_datas2 = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand21\13KW_RH_S_Krsand21.xlsx")
1197         sol = 1
1198         navnaltdatas2 = self.combos2.itemText(i)
1199     elif i == 7:
1200         alt_datas2 = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand21\36KW_RH_uS_Krsand21.xlsx")
1201         sol = 0
1202         navnaltdatas2 = self.combos2.itemText(i)
1203     elif i == 8:
1204         alt_datas2 = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand21\36KW_RH_S_Krsand21.xlsx")
1205         sol = 1
1206         navnaltdatas2 = self.combos2.itemText(i)
1207     elif i == 9:
1208         alt_datas2 = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand21\13KW_JA_uS_Krsand21.xlsx")
1209         sol = 0
1210         navnaltdatas2 = self.combos2.itemText(i)
1211     elif i == 10:
1212         alt_datas2 = pd.read_excel(r"C:\Users\gonna\pythonProject\

```

```

1213         betingelser\ferdig\Krsand21\13KW_JA_S_Krsand21.xlsx")
1214             sol = 1
1215             navnaltdatas2 = self.combos2.itemText(i)
1216             elif i == 11:
1217                 alt_datas2 = pd.read_excel(r"C:\Users\gonna\pythonProject\
1218                 betingelser\ferdig\Krsand21\36KW_JA_uS_Krsand21.xlsx")
1219                 sol = 0
1220                 navnaltdatas2 = self.combos2.itemText(i)
1221                 elif i == 12:
1222                     alt_datas2 = pd.read_excel(r"C:\Users\gonna\pythonProject\
1223                     betingelser\ferdig\Krsand21\36KW_JA_S_Krsand21.xlsx")
1224                     sol = 1
1225                     navnaltdatas2 = self.combos2.itemText(i)
1226                     elif i == 13:
1227                         alt_datas2 = pd.read_excel(r"C:\Users\gonna\pythonProject\
1228                         betingelser\ferdig\Krsand19\13KW_EB_uS_Krsand19.xlsx")
1229                         sol = 0
1230                         navnaltdatas2 = self.combos2.itemText(i)
1231                         elif i == 14:
1232                             alt_datas2 = pd.read_excel(r"C:\Users\gonna\pythonProject\
1233                             betingelser\ferdig\Krsand19\13KW_EB_S_Krsand19.xlsx")
1234                             sol = 1
1235                             navnaltdatas2 = self.combos2.itemText(i)
1236                             elif i == 15:
1237                                 alt_datas2 = pd.read_excel(r"C:\Users\gonna\pythonProject\
1238                                 betingelser\ferdig\Krsand19\36KW_EB_uS_Krsand19.xlsx")
1239                                 sol = 0
1240                                 navnaltdatas2 = self.combos2.itemText(i)
1241                                 elif i == 16:
1242                                     alt_datas2 = pd.read_excel(r"C:\Users\gonna\pythonProject\
1243                                     betingelser\ferdig\Krsand19\36KW_EB_S_Krsand19.xlsx")
1244                                     sol = 1
1245                                     navnaltdatas2 = self.combos2.itemText(i)
1246                                     elif i == 17:
1247                                         alt_datas2 = pd.read_excel(r"C:\Users\gonna\pythonProject\
1248                                         betingelser\ferdig\Krsand19\13KW_RH_uS_Krsand19.xlsx")
1249                                         sol = 0
1250                                         navnaltdatas2 = self.combos2.itemText(i)
1251                                         elif i == 18:
1252                                             alt_datas2 = pd.read_excel(r"C:\Users\gonna\pythonProject\
1253                                             betingelser\ferdig\Krsand19\13KW_RH_S_Krsand19.xlsx")
1254                                             sol = 1
1255                                             navnaltdatas2 = self.combos2.itemText(i)
1256                                             elif i == 19:

```

```

1256         alt_datas2 = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\DK1_21\13KW_EB_uS_DK121.xlsx")
1257         sol = 0
1258         navnaltdatas2 = self.combos2.itemText(i)
1259         elif i == 22:
1260             alt_datas2 = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\DK1_21\13KW_EB_S_DK121.xlsx")
1261             sol = 1
1262             navnaltdatas2 = self.combos2.itemText(i)
1263             elif i == 23:
1264                 alt_datas2 = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\N02_21\13KW_EB_uS_N0221.xlsx")
1265                 sol = 0
1266                 navnaltdatas2 = self.combos2.itemText(i)
1267                 elif i == 24:
1268                     alt_datas2 = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\N02_21\13KW_EB_S_N0221.xlsx")
1269                     sol = 0
1270                     navnaltdatas2 = self.combos2.itemText(i)
1271                     elif i == 25:
1272                         alt_datas2 = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\N02_21\13KW_R_S_N021.xlsx")
1273                         sol = 1
1274                         #kapasitet_batteri = 13.5
1275                         navnaltdatas2 = self.combos2.itemText(i)
1276                         elif i == 26:
1277                             alt_datas2 = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand20\13KW_EB_S_Kr20.xlsx")
1278                             sol = 1
1279                             navnaltdatas2 = self.combos2.itemText(i)
1280                             elif i == 27:
1281                                 alt_datas2 = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\13KW_EB_SW_Kr21.xlsx")
1282                                 sol = 1
1283                                 navnaltdatas2 = self.combos2.itemText(i)
1284                                 elif i == 28:
1285                                     alt_datas2 = pd.read_excel(r"C:\Users\gonna\pythonProject\
betingelser\ferdig\Krsand21\13KW_R_S_Kr21.xlsx")
1286                                     sol = 1
1287                                     navnaltdatas2 = self.combos2.itemText(i)
1288
1289                 self.retranslateUi(MainWindow)
1290                 self.figure.clear()
1291                 self.canvas.draw()
1292                 print(self.combo.itemText(i))
1293
1294
1295 if __name__ == "__main__":
1296     app = QtWidgets.QApplication(sys.argv)
1297     MainWindow = QtWidgets.QMainWindow()
1298     ui = Ui_MainWindow()
1299     ui.setupUi(MainWindow)
1300     MainWindow.show()
1301     sys.exit(app.exec_())

```