MASTER THESIS

Modelling the LIFE project using DEMKit

Yinping Dai

Sustainable Energy Technology
Faculty of Engineering Technology

EXAMINATION COMMITTEE
Prof. dr. J.L. Hurink
Dr. ir. G. Hoogsteen
Dr. ir. P.W. De Vries

13 January 2020

UNIVERSITY OF TWENTE.

ABSTRACT

With the depletion of fossil fuel and the acceleration of climate change, sustainability is more valued by the public and governments. Sustainable technologies, such as renewable energy technologies and smart appliances, are acknowledged as promising solutions to reduce carbon footprint. The University of Twente initiated a project named the LIFE with the intention to research residential energy and water system by incorporating various sustainable technologies. In this thesis, we explore the possibility of the LIFE microgrid to operate in a near-autarkic condition by DEMKit.

The LIFE as envisioned consists of a 3 kW wind turbine, an EV parking lot with 25 kWp PV panels, a hybrid storage system (a short-term and seasonal buffer), and three tiny houses (including underfloor infrared heating systems). The models of the first three components are created and integrated into the DEMKit. Also, a long-term planning approach for buffers is developed to support seasonal storage. Besides, the Profile Steering control algorithm (PS) is applied to improve the Degree of Autarky (DoA) of the microgrid. The continuous power mode without loss and discrete power mode with the seasonal buffer conversion efficiencies, 45% for discharging and 65% for charging, are used in the simulation.

The potential interactions between users and sustainable technologies and the consequential user behavior change are studied through literature research. A decrease of 10% is estimated for each house. The annual energy consumption of a normal household and a campus EV are estimated to be 4-4.5 MWh (including heating) and 2-2.5 MWh, respectively. The wind turbine and PV panels generate around 29.3 MWh of electricity a year. Based on this knowledge, we create a normal-behavior scenario and energy-saving scenario based on 10% household consumption decrease).

We studied the impact of potential households' behavior change on the sizing of the storage system, using continuous mode. It is found that PS is capable of improving DoA over 10 percent points alone and around 12 percent points with a hybrid buffer system. With it implemented, the normal-behavior scenario can achieve a 99.8% DoA with a 90 kWh short-term battery and 9000 kWh seasonal storage system. Whereas, a ceiling of 95% DoA exists for the energy-saving scenario under the present storage configuration, predominantly subjected to intentionally introduced prediction error. Nonetheless, a smaller seasonal buffer, 3000 kWh, is enough to reach its maximum DoA.

When exploring the maximum amount of tiny houses that the LIFE can supply with the aforementioned PV and wind turbine, the 95% ceiling appears again (using continuous mode). With a 210 kWh short-term battery and 12000 kWh seasonal storage, six tiny houses plus a campus EV, whose total loads is 27.26 MWh, can achieve 94.7% of DoA. Moreover, the discrete power mode is exerted on the normal behavior scenario of three tiny houses. A 60 kWh short-term battery and 6000 kWh seasonal buffer results in 78.5% for DoA. The relatively low degree of autarky is mainly due to the enormous conversion losses, around 14.18 MWh, which turns the scenario into an extreme case. For a more compelling storage model, integrating loss into the continuous power mode of DEMKit and tackling prediction errors (95% ceiling problem) is desired. It is expected that with these improvements, the normal-behavior scenario may accomplish the target of near autarky with a larger long-term buffer.

CONTENTS

ABBREVIATIONS

AI	Artificial intelligence
ALPG	Artificial load profile generator
Auction	Double-sided auction control algorithm
CCT	Correlated Color Temperature
CHP	Combined heat and power
CoP	Coefficient of Performance
DEMKit	Decentralized Energy Management Simulation and Demonstration Toolkit
DoA	Degree of Autarky
E-boiler	Electric boiler
ES	Energy-saving scenario
EV	Electric vehicle
HB	Hybrid buffer system
HEMS	Home energy management system
HVAC	Heating, ventilation, and air conditioning
IoT	Internet of Things
LIFE	Living Lab for Innovative Future Environments
NB	No battery
NC	No control
NL	No loss
OSST	Occupancy sensing thermostat with schedule-learning algorithms
OST	Occupancy sensing only thermostat
PEM	Proton Exchange Membrane
PlAuc	Planning-based Auction control algorithm
PS	Profile steering control algorithm
PSC	Continuous Profile Steering
PSD	Discrete Profile Steering
PT	Programmable thermostat
PV	Photovoltaic
RE	Renewable energy
R_{LtP}	Ratio of the total load to production
SB	Short- term buffer
SBNC	Short-term buffer not controlled
SDGs	Sustainable development goals
SoC	State of charge of buffer, unit kWh
UIHS	Underfloor infrared heating system
UT	University of Twente

1. INTRODUCTION

1.1 Background

In the context of climate change, the Dutch government set sustainable development goals (SDGs) for 2030 [1]. In the energy sector, a transition from fossil fuels to renewable energy (RE) by promoting RE access for households is emphasized. 16% and nearly 100% sustainable energy are the targets by 2023 and 2050, respectively. An associated UN goal, SDG 11 sustainable cities and communities, expresses the idea of safe, affordable, good-quality housing with adequate room improving the dwellers' sense of well-being. Along with the penetration of RE technologies and relevant smart devices, the way of how people lives will be unconsciously changed, which is an inevitable topic while studying smart energy community.

Living Lab for Innovative Future Environments (LIFE) is initiated by the University of Twente (UT), cooperating with multiple RE companies, e.g., Super B, and aims to explore a residential solution by integrating various sustainable technologies in the aspects of water recycling, energy generation and storage, and sustainable community. In terms of energy demand and supply, the goal is to achieve a 'soft-islanding' (near autarkic behavior) scenario of multiple houses, first stage with three tiny houses followed by six houses, and will expand in an evolutional fashion subsequently. The initial proposition for RE technologies is to include electric boiler (E-boiler), underfloor infrared heating system (UIHS), PV, wind turbine, short-term battery storage, and hydrogen system (seasonal storage). On the other hand, research also covers the synergy among dwellers, smart devices, and RE technologies for energy consumption reduction and efficiency improvement.

The sizing of energy generation and storage assets to achieve soft-islanding for 16 houses in the Netherlands was investigated [2]. This study finds that if each house is equipped with 4 kWh battery and 22.4 m^2 PV panels and shared a 60 $kW_{thermal}$ / 30 $kW_{electricity}$ CHP unit, a Degree of Autarky (DoA) of 99.1% can be achieved. In this case, Profile Steering control methodology (PS) is applied to control the micro-grid. The results shows the potential of such an autarkic solution from the technical perspective. With this result, we involve the aims to achieve similar results from the LIFE project. In order to do so, we require such a model of the tiny houses project to determine the following proper steps for its evolution.

1.2 Problem definition

The LIFE, as a demonstration project, is expected to explore all possibilities of cutting-edge technology application. Aside from the aforementioned RE technologies, other promising technologies, especially smart devices, can be added as testing components. The use of these smart devices is supposed to benefit the DoA and the synergy effect with the user's behavior. An example is that people are willing to engage in load-shifting activity with a visual energy display [3]. One of the main objectives is to study these state-of-the-art technologies, their interaction with users, and, more importantly, the changes they impose on energy conservation and dweller behavior.

Due to the asynchronization of RE production and household consumption (e.g., usage peak in the evening while PV production during the day), a well-designed micro-grid system is necessary. Such a system requires the proper sizing of different RE and storage components, and suitable control strategy, especially for buffers, predominantly seasonal storage. The second main objective of this study is thus to explore what is needed from the technical part to create an autarkic field lab. As mentioned above, user behavior may also impact the efficiency of the system. We explore these options for future expansions of the living lab by means of simulation studies using the Decentralized Energy Management Simulation and Demonstration Toolkit

(DEMKit) [4]. Note that, the social and technical aspects are intertwined with each other, and the thesis will be centered around the ‘human-in-the-loop’ scope.

1.3 Research questions

To reach the objectives and requirements mentioned in the previous section, the following research questions have to be answered:

Main research question:

What is needed from the technical part to create an autarkic field lab?

The main research question can be decomposed into the following sub-questions:

1) What are the state-of-the-art sustainable technologies that can be included to improve system DoA?
2) How would user behavior change with these technologies being applied, and how would these changes influence power balance in the microgrid?
3) What is needed from the technical part to support people in making these social changes?
4) How can the optimal size of these technologies be determined in the light of system integration?

1.4 Approach

As this thesis only contributes to the theoretical study in the preliminary phase of the LIFE project, the first main objective is conducted through a literature study. First of all, promising smart appliances and RE technologies are reviewed. We analysis the role they play in energy saving or efficiency improvement, especially their impact on user behavior shaping. From a practical point of view, the user experience is summarized, and the energy performance results are estimated as references for the testing and analysis phase of the LIFE project.

We use DEMKit to model the micro-grid system, three tiny houses plus a shared electric vehicle (EV) charging parking lot with PV panels on the top. Component models, such as PV, EV, E-boiler, etc., are already available in DEMKit. Instead, the models of a wind turbine, an infrared underfloor heating system, and a hybrid battery system (short-term and seasonal storage system) need to be created for this assignment. The mathematic models are integrated into DEMKit, and thus, the smart grid control algorithm, Profile Steering, can implement seasonal planning. Besides, two scenarios, energy-saving and energy-intense referred to household consumption, are analyzed and compared to explore the different possibilities in reality. The profiles of these scenarios are generated by Artificial Load Profile Generator (ALPG) [5], the result of which is the input to DEMKit.

1.5 Outline of the thesis

In this chapter, we introduced the background of the LIFE project and the technical challenges it is facing. These challenges are translated into research questions, based on which a specific approach is given to conduct the research. A literature study is presented in Chapter 2. We focus on reviewing energy conservation strategies and sustainable technologies, including RE technologies and smart home appliances. Besides, performance indicators are discussed. In Chapter 3, we create a model for the tiny house micro-grid. The layout of the tiny houses is first illustrated, followed by introduction of the used software, ALPG and DEMKit, and profile steering control algorithm. More importantly, we present the modeling details of the three critical components, wind turbine, the infrared heating system, and seasonal battery. Chapter 5 uses this model to simulate the two scenarios, and the results are analyzed by using performance indicators introduced beforehand. Chapter 6 concludes the thesis by answering the introduced research questions, and recommendations for future work are presented.

2. LITERATURE STUDY

Technical resorts are not the exclusive way to energy conservation. Influencing people psychologically and sensually can also lead to behavior change towards a more sustainable pattern. All these methods are potentially carried out in the LIFE project. Therefore, literature research is not only about advanced sustainable technologies, particularly smart home appliances, but also non-technical strategies. The intention is to study the likely consequences so as to conceive a general idea of how to build a model for the energy-saving scenario.

2.1 Energy conservation strategies

Household energy conservation has been a topic of interest in the field of applied social and environmental psychological research for several decades. Along with the focus being shifted to climate change, household energy conservation, as an efficient way, has become a hot topic in the sustainability domain as well. Abrahamse et al. [6] categorized energy conservation strategies into antecedent and consequence strategies. These two strategies clarify various interventions that could potentially help households to reduce energy consumption. Note that a variety of interventions are commonly used in a combined fashion.

2.1.1 Antecedent strategies

Antecedent strategies, as the name suggests, are the interventions used before energy is consumed. This type of intervention would impact the households' determinants (the factors that cause behavior change, e.g., knowledge) and thus lead to behavior alteration. For example, affirmative information can endow knowledge about sustainability and changed knowledge would affect people's lifestyle towards a more green one. These interventions include commitment, goal setting, modeling, and information.

Commitment refers to a pledge or promise to alter behavior, which is always involved in goal setting (e.g., decrease energy consumption by 10%). In the research of Pallak et al. [7], commitment showed better effect if it is made publicly, as social norms (e.g., expectations of neighbors) could exert active causes leading to more significant change. The drawback, nonetheless, is the probable discontinuation behavior of conserving energy after 6 months.

Goal setting is always committed with feedback intervention (a type of consequence strategy). Becker [8] compared different goal-setting levels, 20% and 2% of saving energy in the research. The results shows the 20% goals perform better with 15.1%, whereas the 2% goal is barely useful.

Modeling advises people with examples of recommended behaviors. Winett et al. [9] provided various energy-saving measures through a TV channel that targets middle-class homeowners. The energy use was reduced by 10% through modeling.

Information is the most commonly used strategy. The intention is to impart knowledge and to increase the household's awareness in multiple ways, such as tailored information, workshops, and mass media campaigns. The latter three methods not necessarily lead to behavior change, and in fact, even if the change was triggered, the effect is mild. The tailor information differs in provided information, and thus targets would get overload with general information. Winett et al. [10] provide personalized information on air conditioning and heating to subjects, which resulted in 21% energy being conserved.

2.1.2 Consequence strategies

Opposite to antecedent strategies, the measures of consequence strategies are based on the preceding energy consumption pattern and influence household based on observed behaviors. The primary consequence interventions are reward and feedback.

Money is the most straightforward reward and also an effective one. Winett et al. [11] offered a monetary reward to households with information and feedback can save about 12% in 6 weeks. On the other hand, the study of McClelland et al. [12] found that the savings would diminish as the experiment progressed. Other types of rewards comprise tax credit, emoticons, and social rewards (performance indicator with a descriptive comment). Pitts et al. [13] utilized tax deduction from total income taxes as the incentive to attract households to insulate their houses. It turns out the tax credit had no effect at all. Handgraaf et al. [14] claim that social rewards are more effective than these financial rewards since social norms are involved. He conducted a comparative experiment that targets employees in a Dutch company for 13 weeks with grade points with a descriptive comment as social rewards. The results reveal that social rewards outperform financial rewards, and public rewards outperform private rewards. Schultz et al. [15], instead, use emoticons as social rewards among neighbors. People receive either "positively (☺) or negatively (☹) emoticons" depending on whether they consumed less or more than average consumption. Households try to obtain or maintain the positively valenced emotions by saving energy. They are sensitive to if they behave appropriate or not.

Feedback can be regarded as the most flexible strategy, as it is related to or can be applied jointly with all other energy conservation strategies. In general, the provision of feedback can save about 5%-15% of energy [16]. We break down the feedback intervention into the following aspects: why (intention), what (content), when (frequency), and how (approach).

First of all, the same as why people participated in Hargreaves smart monitor/display trial [3], it is believed that the motivations to adopt smart monitors are saving money (also frustration on rising energy prices), environmental concerns, the curiosity on the details of energy consumption, the interest in the technology itself.

The feedback contents are diversified [3]. The most basic ones are real-time and accumulated consumption. Bittle et al. [17] found that for high-energy consumers, cumulative consumption is more effective than daily electricity use, but for medium and low consumers, the effect is opposite. Distinguished to frequency varying in feedback content, continuous and periodical feedback differ in feedback frequency. In Houwelingen and Van Raaij's research [18], the continuous feedback on gas consumption can save about 5% more than that of monthly feedback. The content and scale of consumption feedback also matter. Users argue that the consumption in kWh and translated carbon emission are abstract and too small to provoke action. Instead, the financial interpretation, such as pounds or pence, is preferred [3].

Another associated feedback content is about consumption peak indication with a pricing framework that can provoke load shifting [19]. People also emphasized the necessity of identifying high-consumption or greedy devices [3]. A relatively novel feedback is health based, which frame energy conservation as altruistic and raise the moral cost. Asensio and Delmas [20] framed it as 'Last week, you used XX% more/less electricity than you efficient neighbors. You are adding/avoiding XX pounds of air pollutants, which contribute to known health impacts such as childhood asthma and cancer'. As a result, users displayed a more persistent and effective energy-saving behavior of 10% than a typical financial frame. Feedback content can also be comparative [21]. The comparison objects can be historical data and the consumption of neighbors, friends (from social media), etc. Such comparison plays a role in setting a benchmark, involving competition, and boosting the learning-and-improving loop in the new habit formation process. As for goal setting, people welcome the feedback named 'credit' that suggests the difference between consumption limit and accumulated consumption in a period [3]. Furthermore, some households ask more than domestic energy information, but also wish to incorporate data about transportation, water and gas usage [3].

Email is a traditional but high-engagement approach to deliver feedbacks. Although Asensio and Delmas conveyed the feedback through both email and online portal [20], the online portal was most commonly visited through the link in the weekly email. On the other hand, online portal is a friendly approach for research to check the engagement of users (e.g., through login frequency). A more recent alternative is the mobile app that is characterized by higher accessibility. A home monitor/display is also a popular feedback tool and sometimes is treated as a new attractive gadget that brings more engagement. However, users express the concerns about the consumption provoked by the monitor itself [3].

In this delivery agency topic, an important branch is the design of the interface [3]. On the one hand, users complain about some unwelcome information and appeal to the agency whose interface is customizable. On the other hand, the level of interface sophistication can affect the effectiveness of the feedback. Besides, the aesthetic appearance of the monitor is essential, and a touch screen is preferred. Users tend to move and corner their monitor, due to the inconvenience caused by its volume, the mismatch of the style with surroundings, or no demand for already predictable information. Another interesting subject is gender related. Some monitor users reflected that their female family members are either not interested it or do not understand the monitor. Different schemes are supposed to be included not only for females but also for children and elders.

The feedback methods, as mentioned above, are known as factual feedback (e.g., by providing consumption numerically), through which users need to process the information consciously. People, however, typically lack the motivation or ability to engage. Ambient feedback is found to be a more effective approach (approximately 27% saving) that can be handled even without conscious attention [22]. A typical form is a color-changing light, which is cheap, energy-friendly, low-conspicuity, color- and intensity- changeable, and easily-reachable (as long as light can get to). More importantly, this technology offers feedback that already being evaluated based on a benchmark, hence save the user a lot of effort. Aside from consumption, the color of light can be based on a time-of-use tariff [23] to help achieve energy-saving or load-shifting. From a different perspective, color-changing light can influence people’s feelings. 15% of heating energy is found to be saved through regulating light intensity to ‘deceive’ users perceptually [24].

2.1.3 Implications for LIFE

All the strategies mentioned above seem suitable to be applied in the LIFE project, but still, several items need to be addressed. First, the ‘recipe’ of combined applications is essential and should be delicately designed, especially when it comes to the incorporation with RE and smart technologies. And the experience of previous research should be considered in experiment design, particularly the details. More specifically, not many variations are expected for antecedent strategies. Instead, consequence strategies have more room to explore and extend. For example, we should not be limit to ‘reward’, but also to explore the ‘punishment’ as a consequence intervention.

DEMKit can be connected to a third-party HEMS. Figure 2-1 is an example that DEMKit utilizes the user interface of Home Assistant to display its output. Based on section 2.1.2, a more promising display should be capable of: 1) provide tips for potential improvements, 2) indicate high-consumption and greedy devices, 3) include information about transportation, water and gas usage, travel, etc., 4) incorporate financial and health-based feedbacks, 5) create new branch for goal setting and commitment and their feedbacks 6) the dashboard interface should be designed to be customizable, 7) design gender- and age-specific schemes 8) add share function for friends, etc.

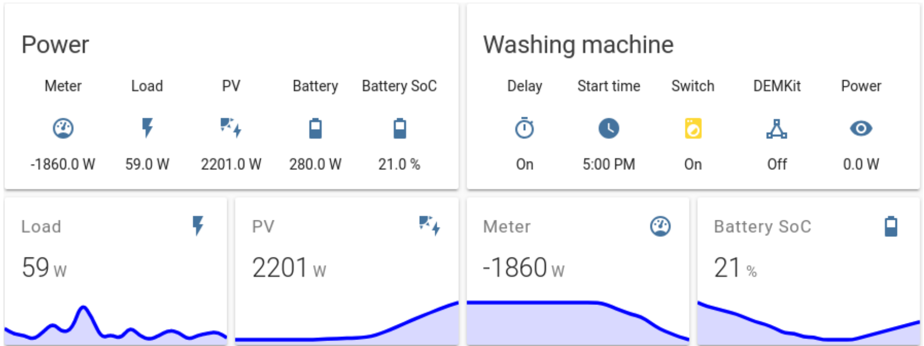


Figure 2-1 The display of DEMKit outputs through third-party (Home Assistant) user interface

Moreover, the novel ambient feedback is not regarded as an alternative or a competitor with factual feedback. They can rather play a symbiotic role together in this project: factual feedback can persuade households to increase their awareness in the early phase; ambient feedback can save user effort and relief or solve the discontinuation problem. Here, we recommend sticking with color- and intensity- changing light indicator as an agency for ambient feedback.

On the other hand, some households express that they would not change certain habits or behavior either by financial incentives or moral pressure to shift the load. These peaks still need to be matched by renewable energy generators or buffers. Meanwhile, some energy conservation strategies, especially the monitor feedback, are reported to be able to prompt users a lot of interest in RE technologies [3]. In addition, the decay of the effectiveness of these strategies over time is normal, which is, however, not wanted.

The majority of the strategies described in this section are all (implicitly or explicitly) based on the assumption that energy-related choices are made after elaborating on the information; the underlying theoretical model would be the Theory of Planned Behavior [25]. The essence here is that consumer's attitudes and social norms play a predominant role in the intention which further alters their behavior. Habituation is one of the causes that often weakens the association between intentions and behavior. McCalley et al. [26] argue that habits are more influential than intentions on everyday life behaviors, and tends to override intentions when the latter goes against the former. As habits happen without conscious deliberation, the LIFE project thus ought to focus on 'conscious behavior' for antecedent and consequence strategies to be effective. While if feedback is presented at the moment the behavioral choice is made, the situation would be different. Confronting people with their prospective energy use at the moment they are setting their washing machine may well work because it interrupts (potentially) routinized behavioral patterns [27]. This beforehand feedback just corresponds to what DEMKit is capable of that predict the event and analysis its potential outcomes.

In terms of the implication for modeling, the effect of these strategies varies a lot, which largely depends on the awareness, knowledge, characteristics and old habits. We take the average value of energy-saving which is around 15% for the LIFE. In the following section, we will start by discussing the interaction between RE technologies and the users.

2.2 Sustainable technologies

2.2.1 Renewable energy technologies

The well-known renewable energy technologies are PV, wind turbine, biomass, etc. Their technologies are mature, and knowledge is abundant. However, the study of their interaction with users is rare. Although these technologies are characterized by distinct generation profiles, the insights of their interaction with humans are consistent. Here, we take the PV panel as an example to illustrate how they alter users' behavior.

The majority of researchers hold that micro generations can enhance the users' awareness and hence favor energy conservation and load shifting [28, 29]. The environmental awareness of these RE technology adopters is more or less stronger than most public even before they decided to purchase a RE product. The employ of these appliances further helps users gain more knowledge about (domestic) energy. Nevertheless, with lower bills, users might start to consume more energy, which is known as the rebound effect in energy economics. Qiu et al. [30] ascribe it to the perceived additional 'income' (from selling productions) or shrunk energy bill. Their research outcome of PV rebound effects is measured to be 18%: 1 kWh PV yield induces an additional 0.18 kWh consumption from 277 solar homes in 4 years. Other researches also reported a similar value of 20% for rebound effect [31-33].

On the other hand, along with the installation of these RE appliances, a home display is usually adopted to monitor the micro-grid. Keirstead [34] found that the monitoring device is mostly used to check the functioning of the PV system, instead of for load-shifting. A wiser option to accomplish load-shifting might be through smart appliances, which we describe in the next section.

2.2.2 HEMS and Smart appliances

Home energy management system (HEMS) is acknowledged to be a handy tool to facilitate households living a sustainable life. Commercialized compatible components are smart plugs, smart lighting systems, smart thermostats, and other smart appliances (e.g., refrigerator). HEMS and its components are often labeled with 'intelligence' or 'smart' because they can turn domestic energy management to (semi-) automatic. Their tasks generally include analyzing device status and environment (e.g., diagnose operation conditions), predicting future energy demand, determining control setting or management strategies (e.g., optimizing device's operation state/efficiency, and interaction with the users (e.g., suggest the device's maintaining schedule and take users' preset).

The HEMS we discussed in this section is different from DEMKit, as these smart appliances are not compatible with DEMKit yet. These market-available smart appliances are usually able to work independently, and the literature about their user interaction study usually targets only one technology. Ford et al.[35] reviewed 308 HEMS products from the market and summarized available functions for each smart appliance category mentioned above. Their study also revealed that the trend of energy portal is shifting from websites and computer software to mobile apps. Lee and Cheng [36] summarized the energy-saving efficacy of individual categories from 305 cases: up to 39.5% averagely for the smart lighting system, around 14.07% for Heating, ventilation, and air conditioning (HVAC), and 16.66% for other products. Alaa et al. [37] reviewed the new and disruptive technology of smart home applications based on Internet of Things (IoT). Hence this literature study no longer repeats these content but focuses on user behavior influence and corresponding energy-saving ratio of varying cases. We aim at the cases in two categories, smart thermostats, and smart lighting systems, as they are the most efficient energy-saving choices.

2.2.2.1 Smart lighting system

Smart lighting systems are more often applied in the office rather than a house or apartment. Nonetheless, similarity exists more or less in the use pattern, user behavior change, and energy use reduction. This part literature study is based on mixed research of household and office smart lighting systems. In this system, color, lighting brightness, and Correlated Color Temperature (CCT) are traditional controllable variables. Conventional techniques are occupancy sensing, daylight harvesting, and dimming. Neida et al. [38] reported that integrated smart lighting systems typically exhibit 17-60% energy savings, and the varying comes from distinct use patterns of the system.

For occupancy-sensing lighting based systems, energy-saving potentially range from 3-60% [38]. Current occupancy sensors generally are based on single-point detection and thus introduce uncertainty in the sensor feedback data. Preset time delays is a regular solution, and need to be calibrated appropriately. Otherwise, more energy might be consumed than a conventional lighting system. The typical time delays are between 5 and 30 min [39]. Eilers et al. [40] revealed that even with occupancy detection, users are half likely to switch on or off the lights manually of 63 offices, and this behavior had induced 30% additional electricity saving.

Heschong Mahone Group [41] conducted phone surveys to their customers, including schools, offices, and other types of occupancy, to compare the switching system and dimming system. The result shows that although the dimming system has higher performance, the switching system is more welcomed (56% vs. 41%). Meanwhile, users complain about the complexity of system operation, difficulty in initial calibration, and brightness not being kept enough. For a dimming system, if with an occupancy detection sensor, users are more likely to select maximum light output, the possibility increase from 89% to 95% [42].

In terms of the daylight-harvesting system, Chew et al. [39] concluded that reported energy-saving is usually over 40%. Besides, Daylight presented a positive effect on the well-being and health of users and thus is a preferable solution. Moreover, the design of the system is supposed to prevent glare that might cause user discomfort.

Non-visual effects of light is a non-negligible factor when it comes to lighting quality that potentially impacts human well-being [43]. An example is that the diversified color temperatures would influence the human perception of a space. Moreover, the immense impact on the human physiological process was confirmed [44], so as the distinct preferences of color temperatures for different spaces [45]. Higher color temperatures that characterized by greater alertness are more suitable for workspace, while lower color temperatures are usually preferred in bedrooms and living rooms [39].

2.2.2.2 Smart thermostat

The smart thermostat is not the traditional energy-consuming devices, but its control subject HVAC is. Different from an occupancy-based lighting system that usually needs only one sensor, a smart thermostat can collect various information, such as occupancy, humidity, temperature, etc. from multiple sensors as inputs to determine the action of HVAC. In this section, we discuss the smart thermostat through pilot project researches findings.

Since the heating or cooling process is time-consuming, pre-cooling or pre-heating is a popular function in smart thermostat products. Schedule-learning algorithms thus could be a helpful auxiliary to enhance the degree of automation. Aarish and Jones [46] evaluated two smart thermostat pilots that compare occupancy sensing only thermostat (OST), occupancy sensing thermostat with schedule-learning algorithms (OSST), and programmable thermostat (PT). In terms of energy saving, OSST resulted in averagely 13.3% gas saving for heating and 14.5% electricity saving for cooling, which outperformed PT with 7.8% gas reduction, and performs slightly better than OST for heating and saved around 10% more for cooling. As for heating season comfort, majority users reflected that they did not notice a change in comfort level, 57% for OST, 65% for OSST. About 20% of user feedback on the comfort level is acceptable. According to the test, precooling can cut 7.9 and 3.6 mins running time from the first and second events. The energy-saving is not significant about 0.847 and 0.472 kW, but saved time potentially contributes to peak load reduction.

Lieb et al. [47] evaluated a smart thermostat pilot that compares two products that are both occupancy-based with remote control options for gas heating systems. For occupancy detection, two technologies are available, motion sensor and GPS. 88% of users kept the sensor on as default, and less than 50% of users turned on GPS. Besides, users are less willing to override the motion sensor. Over 4 months, both product users showed a decline in manual engagement, and the ratios are 60% and 35% individually. Only a few participants disabled the occupancy detection as they claimed that it did not work well for their home. Few users also replaced thermostat to programmable ones. It turns out manual thermostats realized higher gas-saving than programmed ones. Furthermore, users' major complaints are about operational issues, scheduling adjustments, Wi-Fi connectivity, and occupancy detection.

2.2.3 Implications for LIFE

RE technologies are often more acknowledged or realized by the public, but its limitations are obvious too, especially the mismatch between production and demand, and sole solution by providing green energy. However, HEMS and the smart appliance can not only relieve households from trifles; more importantly, it can save a considerable amount of energy and shift peaks if it is well-designed. Also, IoT would also be an indispensable element in the future. These three technologies are essentially mutually beneficial and should be well balanced in the LIFE project.

Regarding to implications of specific technology, the rebound effect of the integration of RE technologies is too significant to be not underscored in the LIFE project. Potential solutions, such as warning rebound effect when it occurred and cautiously presenting financial data. On the flip side, previous research about the rebound effect did not adopt smart appliances, which might mitigate or even eliminate this side effect. Therefore, the effect of (semi-) automated HEMS on the RE rebound effect is worthy of the effort to explore. While for modeling, we estimate the rebound effect to be 15%, considering the knowledge and awareness of the dweller would be higher than average.

For smart appliances, several commons should be realized and accounted for in the LIFE projects. First, the efficacy of smart appliances varies a lot in different scenarios, mostly depend on the use pattern. Hence, the product selection and delicate calibration are critical as it would directly influence the outcome. Second, the service or device accessibility and robustness are essential, sometimes even play a more significant role than energy saving for user adoption and behavior change. Third, the simplicity or the degree of automation of the system more or less determines the continuation of sustainable behavior. Last but not least, the application of HEMS and smart devices is supposed to coordinate with the use of energy conservation strategies, as mentioned above. Last but not least, energy saved by smart appliances can be very high, but probably it partly overlaps with that of energy conservation strategies. Based on the estimation of energy conservation strategies in section 2.1.3 and the rebound effect, we conservatively reckon the average energy saving per household for the LIFE is around 10%.

2.3 Performance indicator

As mentioned in the Introduction, the goal of this thesis project is to achieve near autarkic scenarios for the LIFT project. In order to so, proper evaluation method needs to be introduced. In accordance with previous work of 16 soft-islanding houses [2], the performance indicator used here is DoA, instead of widely-used self-sufficiency, self-consumption, or Demand Cover Factor.

$$DoA = \frac{E_{consumption\_yearly} - E_{import\_yearly}}{E_{consumption\_yearly}} \times 100\%$$

Where E_{import} is the total amount of electricity imported from the grid. In contrast with self-sufficiency, DoA accounts for the part of surplus self-produced energy that stored in buffers and would be consumed by microgrid but at different time intervals. While self-sufficiency is defined as the ratio between the energy directly from RE production and total energy consumption.

The ideal situation is the microgrid to be fully islanded. However, the cost to assure the last few percent points of DoA is too high to be cost-efficient. Therefore, in the case of the LIFE project, we consider 98% as the lower limit for the microgrid to be near autarky.

3. MODELING

3.1 The layout of tiny houses

In the LIFE project, multiple tiny houses along with RE technologies, a wind turbine and an EV parking lot with PV panels, is planned to be situated near the front gate of the University of Twente. The tiny houses are manufactured by EcoCabins [48], and manifold models are available. The model for the LIFE project is not decided yet. Here, we choose the unit of 32 m^2 for modeling. Saxion University of Applied sciences also gets involved in the LIFE project, and they will tackle the application of artificial intelligence (AI) technology to improve the performance of the microgrid, especially to perfect the prediction of the consumption. For this reason, even if different tiny houses would be added in the future, establishing new models is not necessary. Instead, AI technology can solve the problem of demand difference through scaling. Figure 3-1 shows the approximate area of the project inside the campus. Figure 3-2 sketches the layout of six tiny houses and RE technologies.

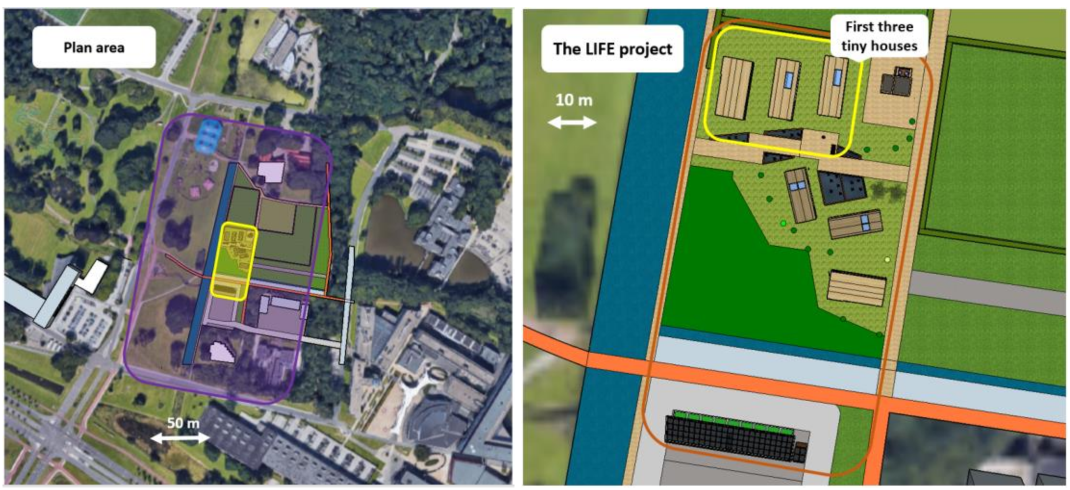


Figure 3-1 The satellite image of tiny houses' location in University of Twente (left) and the arrangement of tiny houses and other components

Figure 3-2 The design sketch of a 32m² tiny house

Figure 3-3 gives a schematic representation of the composition of the individual houses. The square icons represent the devices and energy demands. Each house is equipped with an underfloor infrared heating system and an E-boiler with a buffer/water tank to fulfill the demand of spacing heating and tap water, respectively. The loads are categorized to fixed and flexible. Flexible loads are devices that can be switched on or off at specific times, as long as their function fulfils the requirements of the users (e.g., an EV should be finished charging at the time specified by the user). Whereas fixed loads can not be compromised, for example, watching TV or charging a mobile phone is not supposed to be delayed.

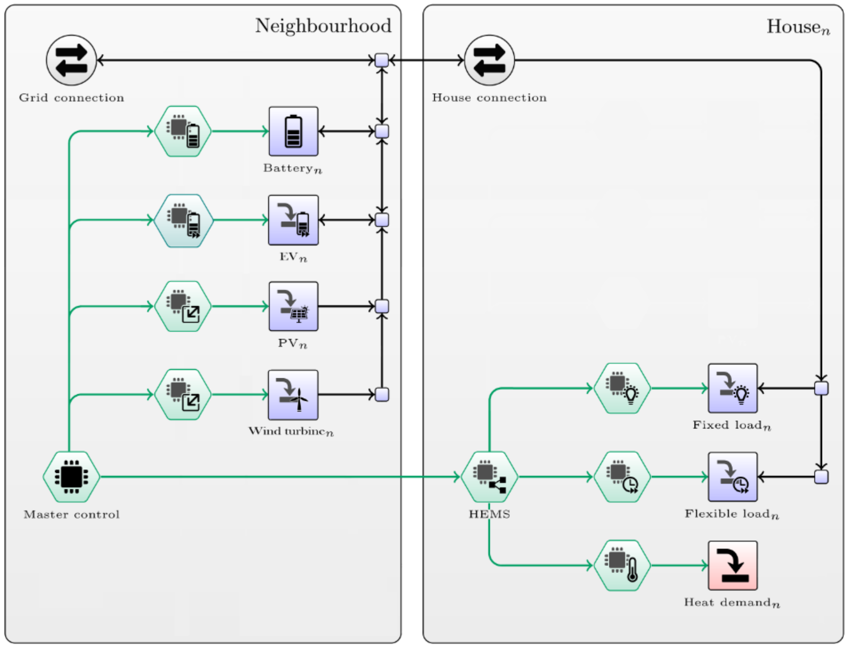


Figure 3-3 Schematic representation of an individual house and its connection to the central battery, EV parking lot and control system.

3.2 Methods

3.2.1 ALPG

As mentioned in the introduction, DEMKit is used to model and simulate the micro-grid of the tiny houses. However, DEMKit cannot work alone; it requires specific inputs data for simulation, so-called scenarios. ALPG, an open-source software, is created to generate such data and is compatible with DEMKit [5]. We use it to generate data, such as load profiles, flexibility details, start/end times of EV, in 1 min interval as inputs for DEMKit. For the user, the most crucial step is to determine the input parameters for the ALPG:

- Simulation parameters: time base, the start day, the number of days to be simulated, and so on.
- Emerging (smart grid) technology penetrations (percentages): EV, PV, heat pump, and so on.
- Power consumption of devices: induction stoves, microwaves, and so on.
- Geographical location: to obtain sunrise and sunset times
- Weather data: temperature and irradiance hourly data
- Household types: SingleWorker, DualWorker, FamilyDualWorker, DualRetired, and so on.

ALPG utilizes probability distributions to determine house occupancy profiles (e.g., When dwellers would be in the house), followed by user behavior (e.g., when to shower, cook, or charge EV, etc.). According to these stochastic profiles, load profiles are yield. The flow-chart below shows the complete simulation process (see Figure 3-4).

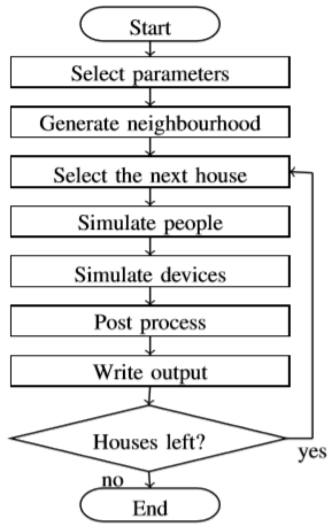


Figure 3-4 ALPG simulation flow chart [5]

To support the functionalities of the DEMKit and Profile Steering algorithm, ALPG needs to provide extra data aside from consumption profiles. In DEMKit, the devices are categorized into eight classes (see Table 3-1). Each class is labeled with different kinds of flexibility. For example, the operation of a washing machine (*Timeshiftable*) can be scheduled from peak hours to off-peak time through the built-in algorithms in DEMKit. The main functionality of PS, in short, is to shift peaks to acquire a profile as flat as possible (details see section 3.2.3). The key resorts are adjusting the operation time, controlling the power of certain devices, and controlling the energy flow of buffers. In order to do so, the flexibility of each device needs to be first identified. Hence, ALPG also yields data for flexible devices, such as start time and end time for timeshiftable devices. Furthermore, other simulation outcomes comprise device parameters, environment data (e.g., ventilation airflow profiles), and so forth.

Due to the feature of reliance on the possibility distribution, the more households being simulated by ALPG the more accurate the outcome would be, and vice versa. Figure 3-5 shows an example of an annual electricity curve of a neighborhood of 81 households in Lochem.

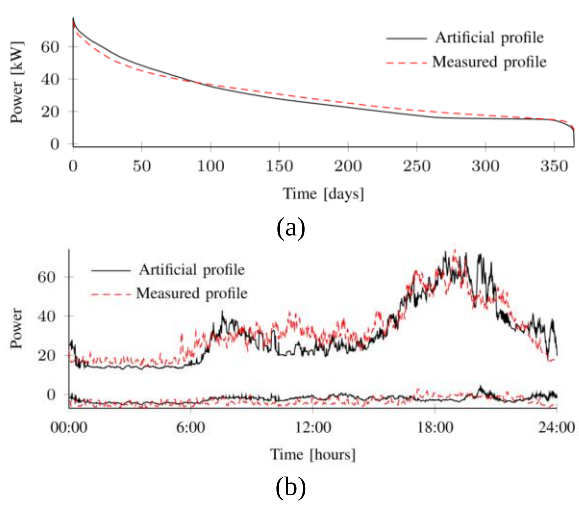


Figure 3-5 (a) Annual electricity duration curve for a neighborhood of 81 households in Lochem; (b) Neighbourhood active power consumption for one day, depicting measurement data and artificial data. The two upper lines depict the active power load in kW; the two lower lines show the reactive power consumption in kvar[5].

3.2.2 DEMKit

DEMKit, developed the University of Twente, is originally designed to test different control algorithms. By further development, it now can be utilized for modeling and simulation. DEMKit is written in python and connected to several open-source software packages, such as InfluxDB (time-series database) and Grafana (data visualization). It can also interact with the API of the open-source home automation software OpenHAB, which can host a user interface to adjust the scenario on the fly. DEMKit applies a cyber-physical systems architecture for a strict separation between control algorithms and physical (device) models (see Figure 3-6).

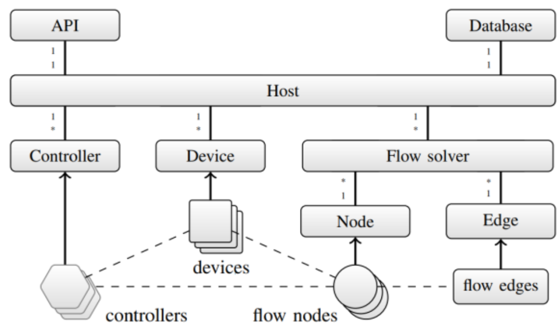


Figure 3-6 Diagram of DEMKit with object references between devices (squares), controllers (hexagons) and infrastructure (circles) in dashed lines[4]

DEMKit uses a set of generic classes to category and distinguish various devices (see Table 3-1). Thanks to this classification schema, new devices can be easily integrated, and it also supports the application of control algorithms. The built-in control algorithms of DEMKit are Profile Steering, Double-sided Auction, and Planning-based Auction. As mentioned before, we only use the Profile Steering algorithm in this thesis. Besides, DEMKit is under continuous development; new functions and components keep being added.

Class	Example devices	Online control	Prediction	Offline optimization	Grid limits	Curtailment / load shedding	Multiple commodities
Uncontrollable	Baseload		1, 3, 4				
Curtailable	Solar panels, wind turbine	2, 3	1, 3, 4	1, 3	1, 2, 3	1, 2, 3	1
Timeshiftable	White goods (e.g. washing machine)	2, 3	1, 3	1, 3	1, 2, 3	1, 2, 3	
Buffer	Battery, heat store	1, 2, 3		1, 3	1, 2, 3	1, 2, 3	1
Buffer-Timeshiftable	Electric vehicle	2, 3, 4	1, 3, 4	1, 3, 4	1, 2, 3	1, 2, 3	1
Converter	Combined heat and power	2, 3			2, 3	2, 3	
Buffer-Converter	Heat pump with storage	2, 3	1, 3	1, 3	1, 2, 3	1, 2, 3	1
Controllers	Home energy management system	1, 2, 3, 4	1, 3, 4	1, 3, 4	1, 2, 3	1, 2, 3	1

Functionality currently supported: 1. Profile Steering [11], 2. Double-sided auction [12], 3. Planning based auction, 4. Valley-filling [13].

Table 3-1 Implemented component classes and supported control and optimization functionality of DEMKit [4]

3.2.3 Profile Steering algorithm

Profile Steering [49] methodology consists of two phases, namely asynchronous scheduling phase, and asynchronous realization phase. The first phase makes use of predictions, based on previous time interval data, to optimize power profile for upcoming time intervals. Subsequently, yield a schedule. The second phase devices exploit their flexibilities, e.g., time-shiftable devices, trying to follow this schedule as good as possible [50]. While prediction errors happen from time to time, for example, an EV needs to finish its charging earlier than predicted, Such that balance between meeting short- and long-term objectives (peak shaving) can be achieved.

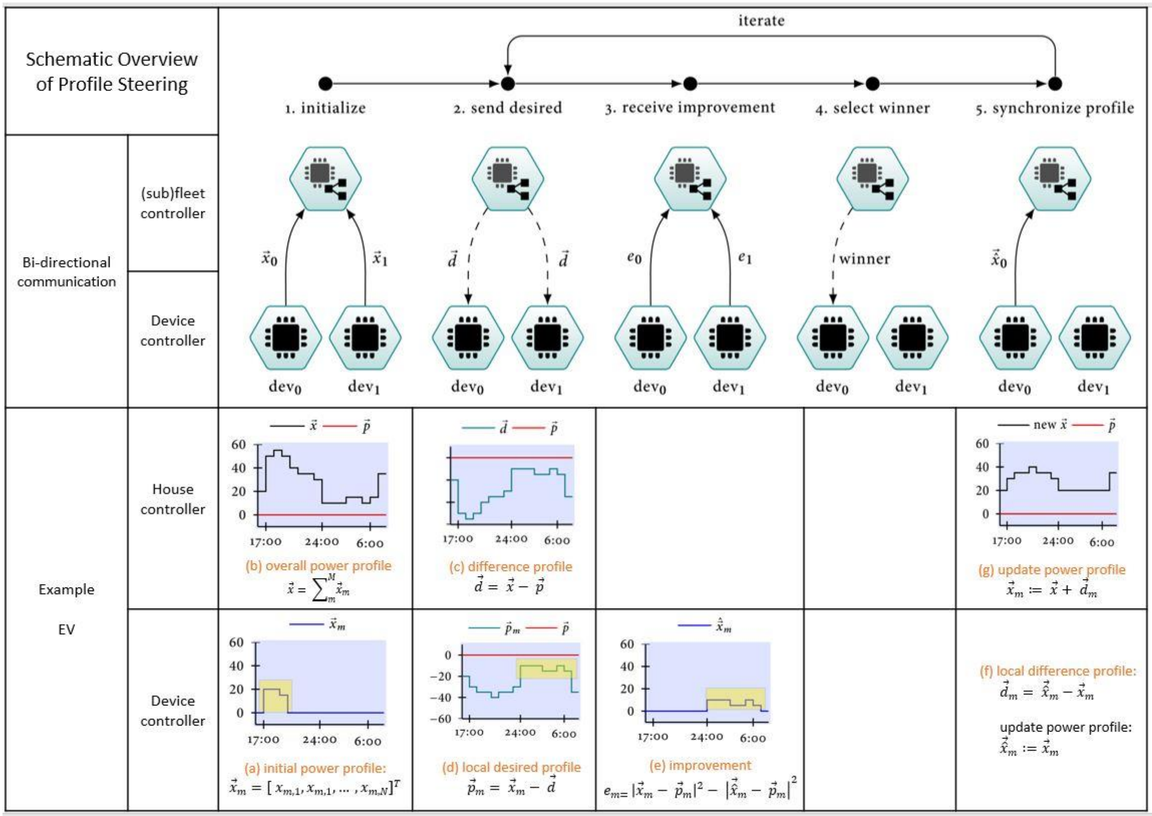


Figure 3-7 Schematic overview of profile steering and example of EV heuristic [51].

How prediction is tackled by DEMKit in a computationally efficient way, will not be explained here, but is addressed in the aforementioned references. The optimization algorithms of synchronous scheduling, yet, will be elaborated to have a better understanding of this approach. The profile steering heuristic consists of initialization and iterative optimization process, coordinated by a (sub)fleet controller, e.g., HEMS (Figure 3-7). Here, take two levels of hierarchy as an example. The first level is HEMS and device controllers at the bottom. An example of an EV arriving at 17:00 with a charging deadline at 7:00 the next day and energy demand of 55 kWh will be explained below.

The main processes:

1. ***Initialize***

(1) HEMS (fleet controller) signals each device controller $m \in \{1,2,\dots,M\}$ to create an initial schedule/power profile $\vec{x}_m = [\, x_{m,1}, x_{m,2}, \dots , x_{m,N}]^{\,T}$ in greedy strategy, e.g., charge an EV as soon as possible. EV power profiles are (a) in the figure.
(2) HEMS receives and aggregates all individual profiles to obtain the overall power profile $\vec{x} = \sum_m^M \vec{x}_m$, see (b).

2. ***Send desired profile***

After initialization, it is likely that the current profile deviations from the desired profile $\vec{p}$ (usually $\vec{p} = [\,0_1, 0_2, \dots , 0_N]^T$, see (b). HEMS will request device controllers to alter their schedules to obtain a better overall profile, minimizing $\|\vec{x} - \vec{p}\|_2$.
(1) HEMS sends difference profile, $\vec{d} = \vec{x} - \vec{p}$ to all devices, see (c).
(2) EV (device) controller(s) calculate a new local desired profile $\vec{p}_m = \vec{x}_m - \vec{d}$, see (d).

3. ***Receive improvement***

(1) EV controller need to construct a new feasible candidate power profile $\vec{\hat{x}}_m$ that minimizes $\left\|\vec{\hat{x}}_m - \vec{p}_m\right\|_2$, see (e).
(2) Each device calculate improvement $e_m = \|\vec{x}_m - \vec{p}_m\|_2 - \left\|\vec{\hat{x}}_m - \vec{p}_m\right\|_2$ and sent back to HEMS.

4. Select winner

(1) HEMS collects all devices improvements and selects the largest (positive) improvement e_m.
(2) HEMS sends requests to corresponding device controller m to replace its scheduled power profile by the candidate power.

5. Synchronize profile

(1) Selected device controller responds with local difference profile $\vec{d}_m = \vec{\hat{x}}_m - \vec{x}_m$ and updates its own power profile $\vec{\hat{x}}_m := \vec{x}_m$.
(2) The fleet controller uses the received local difference profile to update its own power profile $\vec{x}_m := \vec{x} + \vec{d}_m$ (see (g)). This also results in a new difference profile $\vec{d}$ at HEMS.

6. Repeat for all devices

Another device may have a significant improvement based on this new difference profile. Hence we repeat this process iteratively until none of the candidate profiles result in significant improvements or a predefined maximum number of iterations have been exhausted. Note that only one device controller is selected per iteration to prevent possible oscillation and overshoot problems.

In a nutshell, the optimization process tries to reshape and relocate the device power profile (see yellow highlights in the figure) to minimize $\|\vec{x} - \vec{p}\|_2$ with $\vec{p} = [\,0_1, 0_2, \dots, 0_N]^T$. The drawback that PS requires too much computational power to be employed in real-time control is solved. Overall, the Profile Steering is a promising approach to help the tiny house microgrid to achieve the target of the near autarkic operation.

3.3 Components modeling

The major components of the LIFE project are three tiny houses, EV parking lot with PV panels, wind turbine, and storage system. Except for the basic houses and PV, all other components do not have a generic model in DEMKit. Also, the built-in DEMKit heating systems are the gas boiler, electric boiler, heat pump, and CHP. Whereas the tiny house takes advantage of the underfloor heating system. In short, we need to create four models with DEMKit. To be noticed, in the modeling results of component profiles, positive values are the consumption, and the negative value represents production or the electricity export to the grid.

3.3.1 Underfloor infrared heating system

The conventional underfloor heating system is powered by hot water, while cutting-edge technology is infrared heating. An easy-deployment product is the infrared heating panel, which is usually attached to a wall. The heat source of this underfloor heating system is infrared heating films powered by electricity. The film works based on electrical resistance by emitting far-infrared rays and far anionic rays [52]. These harmless rays directly heat people and objects (e.g., walls and furniture), similar to the working of the sun rays. The heated objects then further warm up the room air. The modeling of a heating system always consists of demand and supply. We first discuss the way to modeling the heating supply as it shapes the model of demand.

Figure 3-8 An example of the infrared heating film [53]

To the best of our knowledge, no literature exists on the modeling of UIHS. Only a few are about infrared heaters for greenhouses [54, 55], but the heating media is not thin film, and the objects heated are plants. Another paper is about the numerical model and thermal behavior of electric radiant heating panels. Even these film and panel products both take advantage of radiation technology, their mathematical models still differ, and the parameters of the infrared film (e.g., thermal resistance) are not disclosed. Despite the working principle differs, IUHS inherits the controlling method of using thermostats from traditional heating systems. Plus, the main task of this thesis is not to model a heating system, but more on seasonal storage. We consider simplifying the heating system modeling by using an available model in DEMKit. The idea is that we do not model the psychical processes (e.g., radiation). Instead, we directly use the heat input that is converted from electricity by Coefficient of Performance (CoP). This way, we end up with an electricity demand profile that matches the thermal energy used by a tiny house.

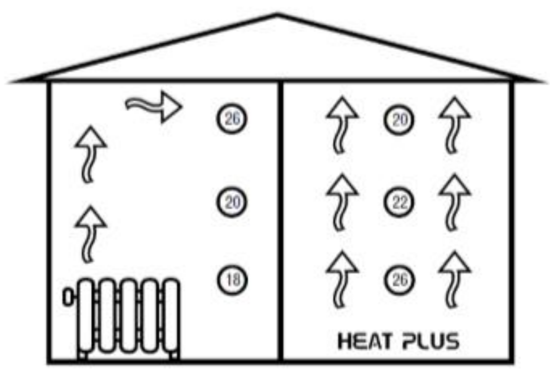


Figure 3-9 The schematic diagram of the temperature distribution comparison between a heat pump (left) and UIHS (right) [56]

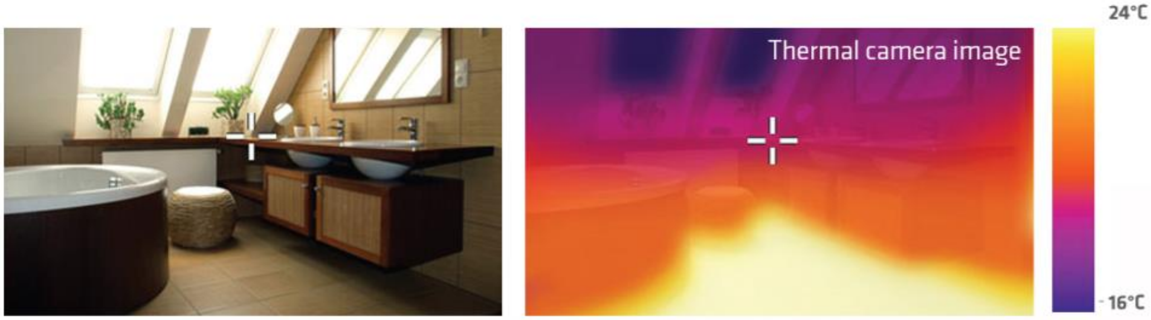
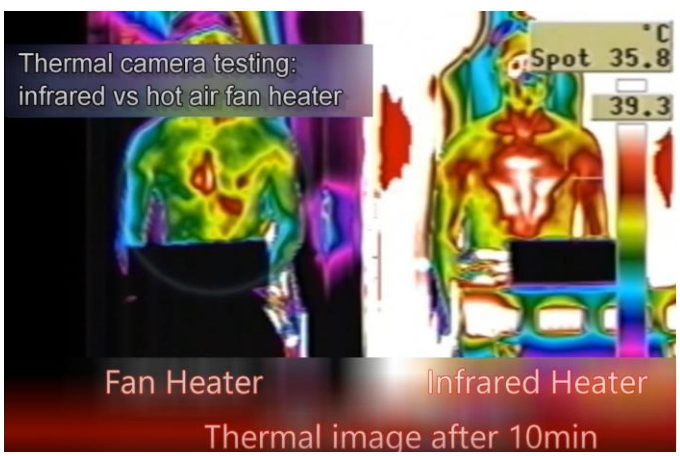


Figure 3-10 An example of underfloor infrared heating system of Heat Décor [57]

The features of IUHS and the corresponding modeling solutions are presented next. In contrary to traditional heating that directly heats the air, UIHS reduces the loss through radiation, conduction, and convection. Arkon [58] and Termofol [59] claim that their products can save 50% energy compared to conventional heatings (e.g., gas boiler). Besides, the temperature stratification of UIHS is opposite to that of traditional heating systems (see Figure 3-9 and Figure 3-10). With surface heating, the average temperature of the floor and wall surfaces remains around 1-2 ℃ higher than with air heating. Every one-degree drop in air temperature saves about 6% of energy [60]. Therefore, we decrease the temperatures of the thermostat at 2 ℃, in contrast to the default ALPG output, in the modeling. This descent also ascribes to the effect of radiation on the human body, which would lower the heating demand.

In terms of the efficiency of UIHS, Flickstein [61] stands that with tuned energy output, the human body can absorb 93% of the infrared waves that reach the skin. The heating effectiveness of UIHS reaches about 99% [52]. Accordingly, the CoP is set to 100% in DEMKit. The operation of UIHS is confirmed by a product company through emails that the thin film is an on-off device. Unlike conventional electric heating devices, UIHS would switch off as long as the set temperature reached and switch on again until the room temperature drops its tolerance temperature (e.g., 2 ℃). Furthermore, UIHS would bring a more pleasing experience to users [56]. It can accomplish the set temperature in 5 min, much quicker than traditional heating systems. UIHS also keeps walls from impairment by humidity, users from arthritis, and muscle pain from humidity.



Thermal camera testing, infrared vs. hot air fan heater [62]

DEMKit is embodied with two zone demand models, namely 1R1C and 2R2C [63]:

- 1R1C model: contains one thermal mass for the zone and one overall building thermal resistance. All heating inputs and direct losses are defined at the zone thermal mass.

- 2R2C model: besides the zone thermal mass, the variety one additionally contains a thermal mass for the floor heating with connected heating input. Variety two contains a thermal mass for the interior house structure, and the heating input is then defined at the zone thermal mass.

Heat pump and E-boiler can use the 2R2C model, as their heater thermal resistance value is easy to access. As for IUHS, neither its thermal resistance or the thermal capacity of its heating objects can be obtained. Hereby, we stick with a simplified demand model, 1R1C. The rest is to calculate the input data for tiny houses, especially (envelope) thermal resistance and thermal mass/capacity.

Table 3-2 The table of (zone) thermal capacity calculation data of 32 m² tiny houses

	Material	Area [m2]	Thickness [mm]	Reference material	gross density r [kg/m³]	spec. heat capacity C [J/kg.K]	heat capacity [J/K]
Façade	wood frame	58.5	285	Wood 500 kg/m³	500	1600	13338000
	softwood covering	60.4	18	Wood 500 kg/m³	500	1600	869760
	Threefold Glass	14.3	16	Quartz glass	2200	1050	528528
	Chipboard window sills	6.8	30	Chipboard(wood)	500	2500	255000
Floor	wood	36.5	280	Wood 500 kg/m³	500	1600	8176000
	cement tiles	2.4	15	Tiles, concrete	2100	1000	75600
Roof	wood (R-6)	47.6	120	Wood 500 kg/m³	500	1600	4569600
	softwood covering	51.6	30	Wood 500 kg/m³	500	1600	1238400

Eco-Cabins provided us with the explicit building material of 32 m² tiny house type. In Table 3-2, we use specific heat capacity to calculate heat capacity for each item, and the sum, $29 \cdot 10^6$ J/K, is the thermal mass of the tiny house. As for windows and glass facades, their total surface is 3.84 m², 10.5 m², 6.91 m² in North, South, East, respectively. According to the window type-triple glazing, 0.7 is chosen as the shading coefficient [64]. Other parameters include 35 dm³/s for ventilation and 0.4 dm³/s per m² for infiltration. In Table 3-3, the thermal parameter, either thermal insulance or thermal transmittance, is known for each construction material. The transmittance is the reciprocal of insulance. Thermal conductance is the product of Area and transmittance. Then we can calculate the envelope thermal resistance, Re, the reciprocal of average thermal conductance, as a result of 0.01817 K/W.

Table 3-3 The table of thermal resistant calculation data of 32 m² tiny houses

Direction	Construction	Structure	Area [m2]	Thermal insulance R [m²K / W]	thermal transmittance U [W / m²K]	Thermal conductance [W/K]
	Floor,33.1 m²	Floor	33.06	3.50	0.29	9.45
North (front)	Front façade, N - 24.5 m² - 90 °	Façade	20.65	4.50	0.22	4.59
		window	1.6		1.20	1.92
			0.64		1.20	0.77
			1.6		1.20	1.92
	Front Roof (Façade), N - 21.5 m² - 14 °	Pitched roof	21.53	6.00	0.17	3.59

West (right)	Right façade, W - 14.8 m² - 90 °	Façade	14.77	4.50	0.22	3.28
South (rear)	Rear facade, Z - 24.5 m² - 90 °	Façade	13.99	4.50	0.22	3.11
		window	1.6		1.20	1.92
		Door with glass	1.98		1.30	2.57
		Terrace door	6.92		1.20	8.30
	Roof Rear facade, Z - 21.5 m² - 14 °	Pitched roof	21.53	6.00	0.17	3.59
East (left)	Left facade, O - 14.8 m² - 90 °	Façade	7.86	4.50	0.22	1.75
		window	6.9		1.20	8.28

Figure 3-11 shows the simulation results of UIHS for three days with the simulation time base set to 1 min. We set the constant power output of UIHS to 5120 W, and the temperature tolerance to 0.2 ℃. ALPG generates the profile of thermostats temperature setpoints according to the occupancy of the house. It can be seen that as long as the room temperatures drop further than the tolerance, the UIHS would be started. When the room temperature reaches to setting point, the UIHS will shut down.

Furthermore, the room temperature not always vary linearly. It is also affected by the environment, such as sunshine and cold air infiltration (see the blue line from 9:00 to 13:00). As for control, UIFS does not have a conventional heat buffer like a water tank. Although the house structure can act as a free buffer; considering the heat capacity and resistance of wood as well as the complexity, UIFS is set to be uncontrollable in DEMKit. However, when applying PS control for the micro-grid, DEMKit can still predict the consumption of UIFS.

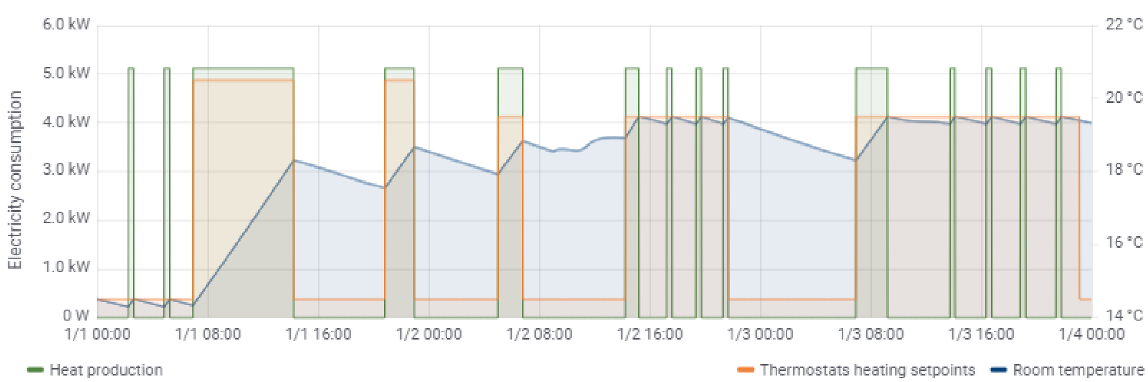


Figure 3-11 The simulation results of the underfloor infrared heating system for 3 days in 2017 January

3.3.2 EV charging parking lot with PV panels

In LIFE, an EV charging parking lot with PV panels is along with the tiny houses. This parking lot will not only serve the residents of tiny houses but also open for the public, especially for university employees. Amperapark is responsible for providing the main parking lot equipment – the Amperaport, a unit with the roofs built of PV panels (see Figure 3-12). 85 PV panels, each in 300 Wp, will be used to support the LIFE project. The quantity in Wp is translated to area and efficiency as inputs to DEMKit by using the following equations:

$$P_{nom} = P * A$$

$$\eta = \frac{p}{A * G}$$

Where η is efficiency, p is power, P_{nom} is nominal power, A is area, and G is the standard (testing) irradiance, 100W/m^2. We first estimated the size of one PV penal to be 1.6 m^2, leading to 187.5 W/m^2 and 18.75% of efficiency. Totally, 136 m^2 PV panels with 18.75% of efficiency, a 10° inclination angle, and a 193° azimuth angle would yield green energy for the tiny houses. The hourly time-series data of solar irradiation is obtained from KNMI [65], as well as other environmental data, such as wind speed for the next section. This results in a total PV yield of 22.42 MWh in 2017, and the yearly profile can be seen in Figure 3-13.

Figure 3-12 The Amperaport – EV parking lot built by PV panels [66]

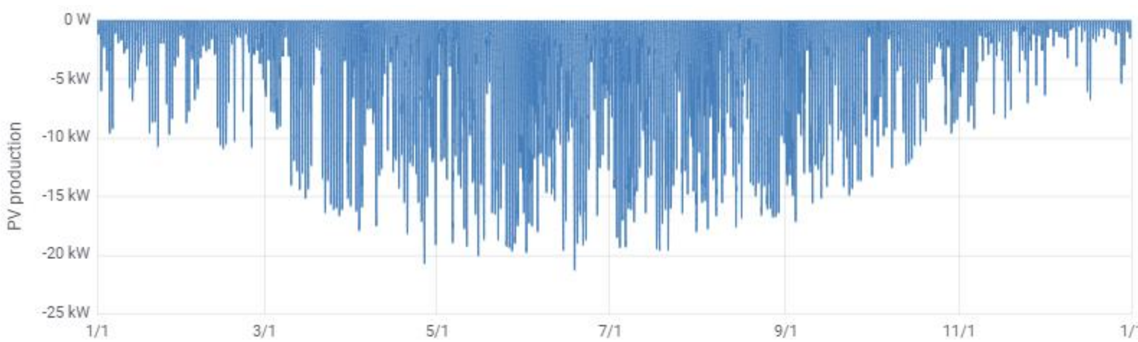


Figure 3-13 The PV production profile of 2017

To model the parking lot as an independent unit, the house concept in the DEMKit default model is borrowed. The production units in this parking lot unit include PV and a wind turbine. A university EV is the only fixed consumption device in this parking lot. This EV shared by the employees of the university for outings (mainly meetings). Therefore, we wrote a new algorithm to create the university EV profile based on built-in EV code in ALPG (see algorithm below). This algorithm utilizes a modified probability distribution of charging events. It is adapted to reflect the expenditure of EV usage for the UT. The input parameter is the range of outing possibility on weekdays. Here, we use a 40%-50% probability. The whole year's energy consumption would vary in each simulation, but around 2 MWh in general.

Algorithm: University EV consumption profile generation

```
function EVoutings(startDay, endDay, outingFreq)
    for day in range (startDay, endDay)
        if day in weekdays, and random() < outingFreq          ➢ determine if there is outings
            if random() < 0.6                                  ➢ Long event
                eventStart = randint (8.5*60, 11*60)           ➢ Event start time in morning hours
                eventDura = randint (5*60, 11*60)              ➢ Event duration in minutes
            else                                               ➢ Short event
```

```
            If random() < 0.5                                  ➢ Start in morning or early afternoon
                eventStart = randint (9*60, 11*60)
            Else
                eventStart = randint (13*60, 14*60)
            eventDura = randint (2*60, 5*60)

        if eventStart + eventDura > 19*60                      ➢ Event must end before 19:00
            eventDura = max (0, 19*60 – eventStart + randint (-20 , 0))

    generateProfile(day, eventStart, eventDura)                ➢ Run function to yield profile
end                                                            ➢

function generateProfile(day, eventStart, eventDura)
    dis=np.interp(eventDura,[2,6],[5,60])+randint (-10,10)     ➢ Calculate drive distance in km
    dis = max (5, dis)                                         ➢ Ensure distance is larger than 5 km
    cons = round (dis / (5+random()) * 1000                    ➢ Energy consumption, 1kWh/5km

    if random() < 0.4                                          ➢ If destination has charging facility
        cons += max ( 0, cons-eventDura * random.uniform(0.5,07) * random.choice([3.5,7,11,22]))
    else
        cons *= 2                                              ➢ No charging, double for return trip

    cons = max (cons, battery)                                 Ensure cons smaller than battery
end
```

3.3.3 Wind turbine

Other than an EV parking lot, a wind turbine is also envisioned to power the LIFE project. A specific wind turbine not determined yet. However, the choices are relatively limited. An airport is located within a 4km radius from the university campus, and thus 45m is the limitation of the highest position of turbine blade tips by regulation [67]. Also, the estimated PV production is already much higher than the annual consumption of three tiny houses. We, therefore, consider a small wind turbine whose parameters are disclosed. As a result, the wind turbine, ZEFIR D7-P3-T10, from Dr Zaber is chosen to be modeled in this thesis (see Figure 3-14). The nominal power of the ZEFIR D7-P3-T10 is 3 kW; rotor diameter is 10m; the hub height is 10m. An advantage of this company is that their products are multifarious, and a lot of information avaible for modelling a wind turbine in DEMKit.

Figure 3-14 wind turbine ZEFIR D7-P3-T10 [68]

A commonly used mathematical model for a wind turbine is based on the power coefficient of the blade that further depends on the tip speed ratio and blade angle [68]. Considering the computational power and accessibility of parameters, we decide to adopt a relatively straightforward approach. We take advantage of the power curve of the wind turbine (see Figure 3-15) to estimate the output power. The nominal power of each integer wind speed is taken, and then use interpolation to compute time-series wind turbine production, based on wind measurements. Before that, the wind speed at hub height needs to be obtained. First, we calculate the roughness length, Z_o, by average wind speed over at different heights by the following equation:

$$\frac{V}{V_{ref}} = \frac{Log_{10}(\mathrm{H}/Z_o)}{Log_{10}\left(H_{ref}/Z_o\right)}$$

Where V and V_{ref} are the wind speeds at height H and H_{ref}, respectively. From the IRENA database [69], we obtained 5.6365 and 4.8072 m/s average wind speeds in 2015 for a height of 100m and 50m respectively and for the location of UT campus. The Z_o results in 0.9, which conforms to the terrain surface characteristics description of roughness class, between forest and city [70].

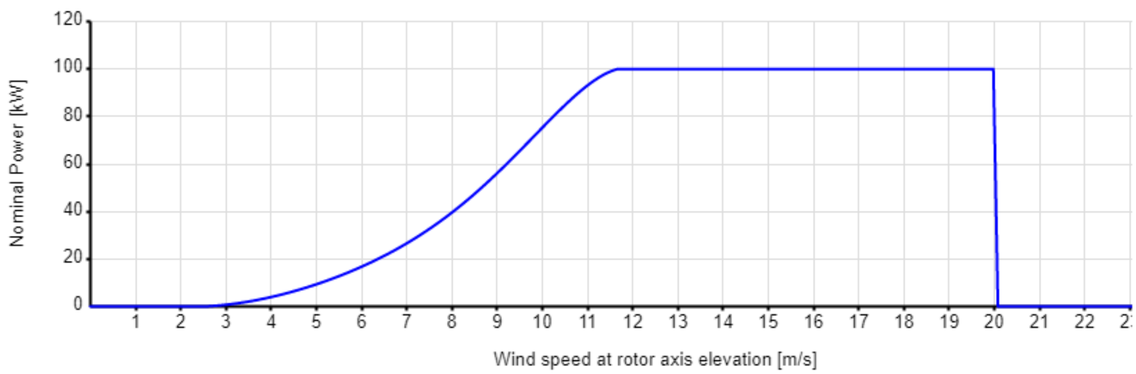


Figure 3-15 the power curve of wind turbine ZEFIR D7-P3-T10 [68]

We can use the same formula to calculate the hub height wind speed, knowing the roughness length, and the wind speed at reference height. The time-series data of wind speed at 10 m height at location 52 ° 16 'NB 06 ° 53'OL (nearby airport) is acquired from the KNMI [71]. Additionally, a scale factor is used to offset the wind speed difference between the locations of the wind turbine and the meteorological station. With the average wind speed at 50m of the Twenthe weather station,4.9292 m/s from the IRENA [69], the scale factor results in 0.975. The total production of the wind turbine in 2017 is 6.88 MWh, and the profile is shown in Figure 3-16. The maximum energy output is fixed at 3 kW, as the limitation of the cut-out speed of 20 m/s. In terms of the model implemented in DEMKit, we wrote a class called windEnv, similar to sunEnv for PV panels. It is in charge of reading and transforming wind data. The model uses this input to yield a production profile of the wind turbine.

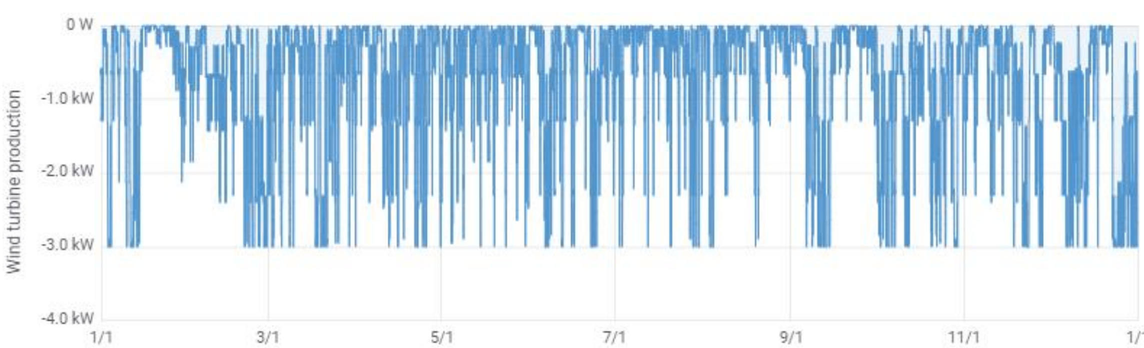


Figure 3-16 The Wind turbine production profile of 2017

3.3.4 Hybrid storage system

3.3.4.1 Introduction

In order to approach the target of a near-autarkic community, the deployment of RE technologies is necessary but not cost-efficient if not with a battery system. Otherwise, the size of the RE appliances would be substantial, as well as overproduction. The seasonal difference of the supply and demand of the micro-grid need to be overcome, the solution depends heavily on the specific case. While, one of the promising solutions is to use a hybrid storage system, storing the overproduction and extracting them when generation cannot meet the demand. Such a system consists of a short-term battery and seasonal storage. The short-term battery mainly tackles short-term fluctuation (day-to-day). The seasonal one is responsible for storing excess RE yield in production peak time (e.g., summer) and supplying energy in high demand periods (e.g., winter). In this thesis, we develop a general control methodology to this hybrid storage system, especially for seasonal/long-term storage, based on the part of the built-in buffer optimization algorithm [72] implemented in DEMKit. A conventional short-term battery and hydrogen storage are expected to be adopted for the LIFE project.

The built-in buffer planning algorithm in DEMKit does not account for conversion losses and losses over time/storage losses. The feasible solution yet to integrate the losses is by the discrete power mode in the PS algorithm rather than normally adopted continuous power mode. In discrete power mode, the buffer can only operate at a set of predefined powers, in contrast to a power range of continuous mode. As this option is also not how seasonal storage works in reality, we will present both models, providing concepts for future work.

In order to test the performance of the hybrid storage system, we generate a different scenario where the total load approach to the generations in 2017. The loads consist of a campus EV and three tiny houses with the yearly consumptions of 1.99 MWh and 12.68 MWh (for white goods and heating), respectively. The yearly renewable energy production is composed of 7.91 MWh PV yield and 6.88 MWh wind turbine generation. The composition of the hybrid storage system is a short-term battery with 20 kWh capacity and 20 kW power and seasonal storage with 300 kWh capacity and 150 kW power. The initial SoC for both buffers is 80% of its capacity. Unless specially specified, all simulations in this section use this scenario.

3.3.4.2 Hydrogen storage

The Hygear will provide a hydrogen storage system, but the product and parameters are not certain yet. Hereby, (operational) criterions in the model are based on typical hydrogen storages. A hydrogen storage system comprises three main components, an electrolyzer that converts electricity into chemical energy (charging), a storage container, and a fuel cell that reverses the process of the electrolyzer (discharging). Typical technologies of electrolyzer and fuel cell are alkaline and proton Exchange Membrane (PEM). The approaches to store hydrogen energy are compression in gas cylinders, as a cryogenic liquid, and as a reversible metal hydride. Each technology is characterized by unique strengths, weaknesses, and operational conditions. Instead of determining a specific system, we utilized the average parameters (e.g., conversion efficiency) and common operational rules to create a general model.

During the conversion between chemical energy and electricity, the losses are high in contrast with near 100% efficiency of the storage phase. The theoretical conversion efficiencies of an electrolyzer, η_{Elect}, and fuel cell, η_{FC}, are higher than those measured ones in demonstration projects or lab experiments. Theoretically, η_{Elect} is around 74% and η_{FC} is around 50% [73, 74]. In practice, η_{Elect} is around 65% and η_{FC} is around 45% [75-79]. In consequence, the latter one is adopted for the LIFE modeling.

When no control applied, the operation of a hydrogen storage system depends on the SoC of short-term battery for a hybrid buffer system, namely on-off cycling limitations [78]. The adopted limitations is 10%

and 90%, which means, as long as short-term battery SoC is higher or lower than 90%, the electrolyzer will start or shut down, the same principle for the fuel cell.

3.3.4.3 The algorithm for no control without losses

In this case, seasonal storage is integrated by utilizing the generic buffer device type into DEMKit. The task is then to link two devices and to coordinate them to work together. The short-term battery will try to achieve balance in the first place. The operation of long-term storage depends on the ratio of short-term buffer SoC to its capacity, known as the on-off condition. 10% and 90% are used as the setpoints for discharge and charge of the seasonal buffer respectively. Once the on-off condition of seasonal storage is triggered, the seasonal storage will response (see algorithm below).

Algorithm: The consumption calculation of seasonal buffer in a hybrid buffer system

```
function simulate ()
    calcCons = True                                              ➢ Set calculateConsumption to True
        if hybridStorage == True:                                ➢ Check if hybrid buffer system
            if onOff[1] < stbuffer.soc/ stbuffer.capa < onOff[-1]:  ➢ Check short-term buffer SoC/capacity
                cons = 0.0                                       ➢ Seasonal buffer is off
                calCons = False                                  ➢ Will not participate balancing grid

        if calcCons == True                                      ➢ Seasonal buffer is on
            calculatingCons()                                    ➢ DEMKit built-in function calc cons
end
```

Figure 3-17 shows how hybrid storage works under no control (the seasonal storage capacity is lowered to 50 kWh in this case). As long as the SoC of short-term battery stays in the range of the on-off condition, the seasonal buffer would stay off. Also, in this scenario, the short-term battery a lot of the time is either fully charged or fully discharged over a year (see Figure 3-18). In other words, the seasonal buffer is forced to take the job that it is not supposed to, dealing with daily fluctuations. This would speed up degradation and decrease its lifetime. An appropriate control algorithm thus is noteworthy.

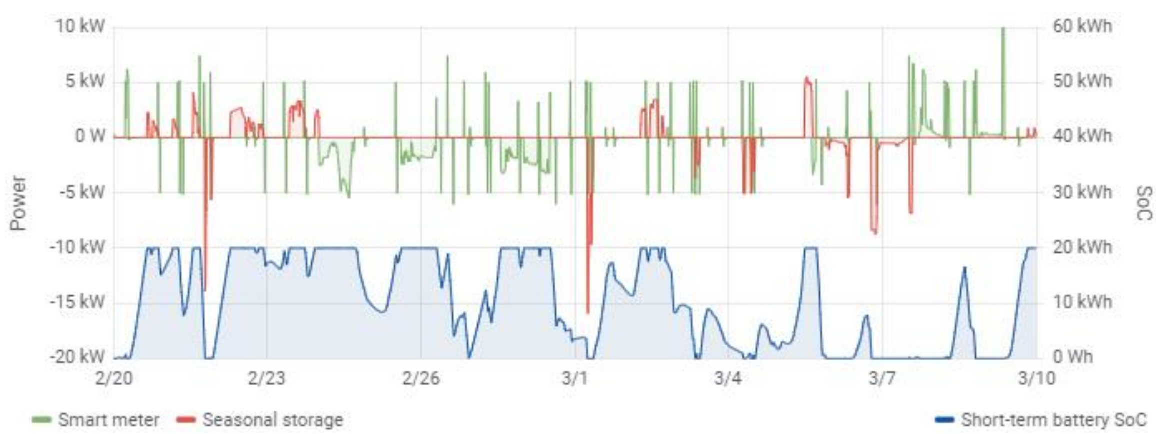


Figure 3-17 The operation of the hybrid storage system based on the on-off condition for no control

Figure 3-18 Control group – the hybrid storage system with no control

3.3.4.4 The algorithm for Profile Steering algorithm

3.3.4.4.1 The algorithm for a sole seasonal storage without losses

The literature regarding the control algorithm of seasonal storage is all about economic conservation, improving the lifetime, and robustness of the system, especially hydrogen storage [80, 81]. Therefore, we need to develop a novel seasonal/long-term optimization approach. This approach requires to be compatible with the built-in Profile Steering algorithm of DEMKit, and able to be integrated into DEMKit. Our solution is based on this existing buffer planning. The basic idea is that we conduct the planning twice, and both use the built-in buffer algorithm for both plannings. The first planning is for the long-term, on which the second (short-term) plan is based. We will illustrate how it works with a single seasonal battery, followed by with the hybrid storage system within the Profile Steering algorithm (PS).

First of all, we need to understand the buffer planning algorithm in DEMKit. Beforehand, two parameters must be known, plan interval that refers to the time interval between two planning sessions and plan horizon that is the period being planned. DEMKit precedes the simulation in a rolling-horizon fashion (see algorithm below). In the default settings, PS make a plan for next two days in day one (plan horizon equals to two days); in day two, new planning would be triggered (plan interval is one day); the original plan for day three is then overwritten by the plan yielded for next two days. The time base is usually set to be 15 mins in simulation and 1 min for real-time operation. One of the reasons to set the default plan horizon to two days is to schedule EVs being charged overnight and thus enhance the load shifting. As for the default plan interval, 1 day, the purpose is the reduce the prediction and scheduling time in reality and also limit the prediction of errors [51].

Algorithm: The rolling-horizon fashion of DEMKit simulation for built-in buffer short-term planning

```
function simulate(time)
    …

    if time >= nextPlan:                          ➢ Check if do new plan this time interval
        currentPlan = doPlanning(time)            ➢ Trigger function to do planning
        nextPlan = time + planInterval * timebase ➢ Set time for next planning (1 day later)
    …

end

function doPlanning (time)
    planStart = time                              ➢ Define the start time of the plan
    planEnd = planStart + planHorizon * timebase  ➢ Plan end time is 2 days later
```

```
    plan = bufPlanning (time planStart, planEnd)      ➢ Call built-in function of buffer planning
return plan                                            ➢ The plan is for next 2 days
```

In the built-in buffer planning algorithm of DEMKit, firstly, the profile of energy exchange with the grid (aggregated profile) is predicted to the planning horizon. Based on the current status and pre-set parameters (e.g., charging powers and capacity) of the buffer, the aggregated profile is translated to the target SoC. This target SoC, rather than the aggregated profile, is an essential input to the planning algorithm. The output of the buffer algorithm is the profile of a plan. Inspired by this, we first simulate for 366 days to gain the yearly aggregated profile, a vector of 366*(24*60/15) elements. We resample this profile based on the time bases of the short-term and seasonal planning, which differs and in this case are 15min and 2 days respectively. The resampled profile is a vector of 183 elements; each element represents the average electricity exchange with the grid of 2 days. Then, we conduct the seasonal buffer planning with the resampled profile, resulting in a profile of the seasonal plan of the (daily average) electricity exchange with the grid. Finally, this profile is translated to the seasonal SoC, also a vector of 183 elements. The seasonal SoC will overwrite the short-term target SoC to lead short-term buffer planning.

Here, the seasonal time base must equal to the short-term plan interval. This is because we try to obtain a target SoC for each short-term planning through seasonal planning, and the premise is the number the seasonal SoC matches with the seasonal simulation interval (simulation time/seasonal time base; e.g., 366 days/1day). On the other hand, to ensure the short-term planning can satisfy the seasonal SoC, the short-term plan interval and horizon need to be equal. If short-term planning continues with the rolling-horizon approach, its short-term prediction would be contradictory to seasonal planning. Plus, with the same reasoning for the default short-term plan horizon, as mentioned above, we change the short-term plan interval to 2 days.

Figure 3-19 The comparison between the on (lower) and off (upper) of the balancing mode for solo seasonal storage with PS control algorithm

Furthermore, in DEMKit, a buffer can either be fully dedicated to follow the plan or can deviate slightly to make up for the uncertainties (balancing mode). Figure 3-19 shows the simulation results of both options. Note that the capacity for seasonal storage is raised to 320 kWh, and its power is elevated to 170 kW, in order to compare with no battery scenario. Though the profiles of seasonal storage in two balancing modes appear to be similar, the smart meter profile of the balancing case is much less violent. In contrast with no control scenario, the balancing case shaves most of the peaks and results in a smooth and flat smart meter profile. Hereby, our seasonal planning approach is viable to some extent, and further (statistical) evidence is given in section 3.3.4.5 and Chapter 4. Notice that in the simulation, we use the perfect prediction, which means that the prediction is the same as the future. In practice, DEMKit can do these predictions based on historical data.

3.3.4.4.2 The algorithm for hybrid seasonal storage without losses

As for the hybrid storage system, the planning algorithm for seasonal storage remains the same, but a link to the short-term battery needs to be established. Short-term buffer takes the responsibility to smooth the daily fluctuation, while long-term storage only requires to focus on fulfilling the seasonal target SoC. The realization of seasonal planning is the first priority. Therefore, the balancing mode is disabled for seasonal buffer and enabled for the short-term battery. The consumption and prediction of the seasonal storage have to be communicated to the short-term before the latter makes any decision. The seasonal storage thus must be simulated in advance to the short-term storage in each time interval, in contrast to the case of no control (see section 3.3.4.4.3).

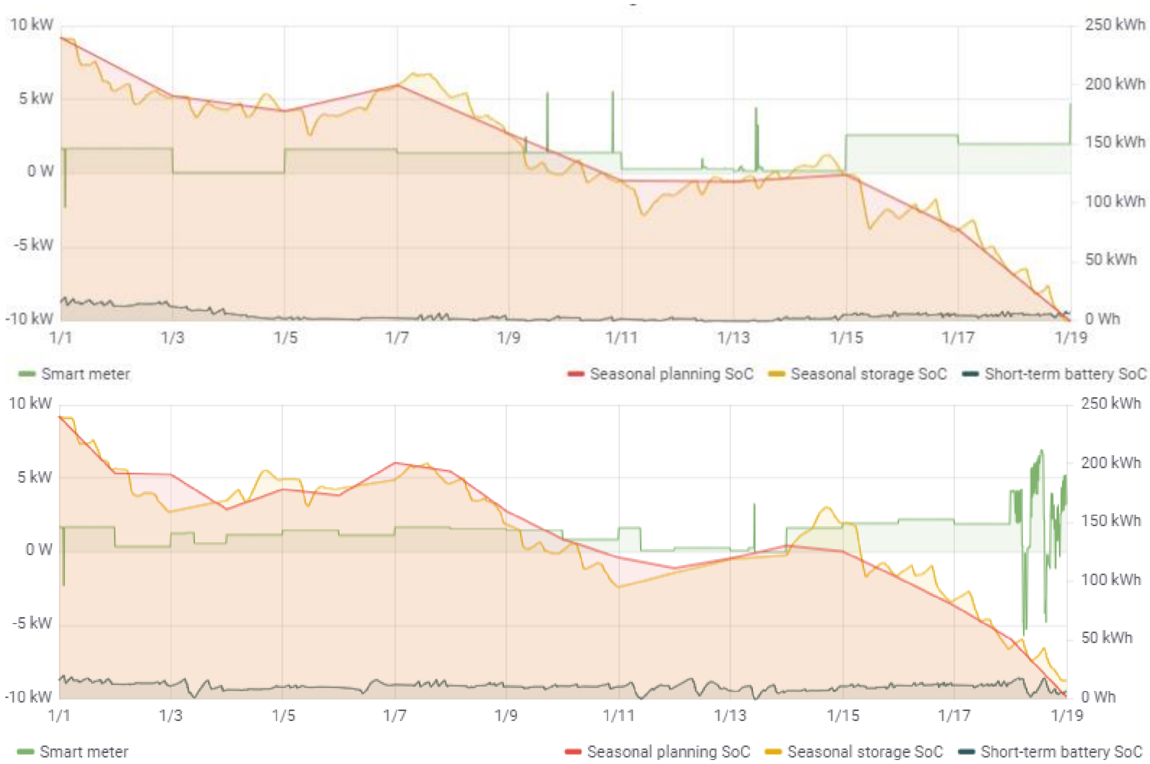


Figure 3-20 The comparison between the perfect realization of the seasonal target (upper) SoC and short-term rolling-horizon simulation method (lower) for hybrid storage system

Notice that during the long-term planning of the hybrid system, the capacity of the short-term battery is not considered. We deem this approach is not only more straightforward but also more robust. Besides, in case of prediction deviation in reality, DEMKit uses parameters namely, plan capacity and plan power, which specifies the proportion of the buffer capacity and power is exploited in the planning. In other words, partial buffer capability is reserved for unexpected peaks. The default values for plan capacity and power are 80% and 60% respectively. The drawback of this setting is that in some cases, the buffer with no control would perform better than that of PS case, as these two default values are both 100% for no control. In the LIFE project, we keep the flexibility for the short-term battery to resolve prediction errors, but adjust both planning capacity and power to 100% for seasonal storage. Even so, the worse results cannot be eradicated, particularly when no battery scenario is already near-autarkic.

The conflict between the short-term rolling-horizon simulation method and the realization of seasonal target SoC, as mentioned above, is comprehended as a beneficial characteristic rather than a problem. We compared the two modes (see Figure 3-20). The only difference is the short-term plan interval; 2 days for the upper figure that represents the perfect realization; 1 day for the lower one that is the rolling-horizon method. The results suggest that the perfect realization mode can indeed achieve the seasonal target SoC at the end of every 2 days. Despite that the rolling-horizon model failed to do so, it still follows the trend of seasonal target SoC. This deviation somehow can be utilized to make up the uncertainties for the model, as this is what would happen in reality. As we want to create a model as real as possible, 1 day and 2 days are adopted for short-term plan interval and horizon, respectively. It can also be seen that the planned seasonal SoC of the two modes slightly differs. The reason is that the resampled aggregated profiles are distinct owing to short-term plan interval or the time base of seasonal planning.

Similarly, the seasonal plan interval and horizon can be either identical or distinct. Notwithstanding, they play a different role in long-term planning. We set the plan interval to a half-year for seasonal rolling-horizon mode, and 1 year for comparison; the seasonal plan horizon equals to 1 year for both cases. In order to do simulation, more data at least for an additional half-year aggregate profile is needed. The results can be seen in Figure 3-21. The main difference lies in planned seasonal target SoC in last season, winter. The seasonal rolling-horizon mode makes an additional plan in the middle of the year, and this planning takes the next spring into account. Consequently, its SoC is not fully consumed or exported at the end of the year. This seasonal rolling-horizon mode is more realistic and will also be used for the subsequent simulations.

Figure 3-21 The comparison of plan interval and plan horizon for long term planning (upper: interval = horizon = 1 year; lower: interval = ½ horizon = half year)

In short, the agreed settings for the hybrid storage system include 1 day and 2 days for short-term plan interval and horizon, half-year and 1 year for seasonal plan interval and horizon, balancing mode for short-term battery but not for seasonal buffer. There is so much to play with DEMKit, e.g., a buffer can be either controlled or not in a hybrid storage system, even the PS will be exerted on all other components.

3.3.4.4.3 The discrete-mode algorithm for hybrid seasonal storage with losses

In previous sections of this chapter, we have discussed the modeling of the hybrid storage system, while without loss considered. The only way yet to involve the loss or the efficiency of seasonal buffer in DEMKit is through discrete mode. The discreteness, here, refers to the nature of the power and the corresponding efficiency. The seasonal storage can only operate at specified powers (a vector in DEMKit). In contrast, the continuous mode specifies a range for power. Following the scattered power, the efficiency needs to be defined in a discrete fashion. Each power can be assigned with unique efficiency. So far, only conversion efficiencies can be appointed in DEMKit. Nonetheless, the tuning on conversion efficiencies can be done to take into account the storage efficiency. In this thesis, we allocate coherent conversion efficiencies, 45% and 65%, for the discharging and charging respectively. Notice that the discrete mode is only for seasonal storage, and we ignore the losses for the short-term battery.

The specification of the power is, however, tricky. First of all, the length of the discrete power vector ought to be a compromise between a computationally efficient simulation and decent simulation results. Moreover, the long-term planning and ultimate realization of the power shall approach that of the continuous mode as much as possible. Figure 3-22 shows an example in continuous mode. The power of long-term planning mostly lands on relatively small values, compared to the ultimate realization. That is because long-term planning is based on average consumption each day, whereas the realization deals with short-term fluctuations. Besides, the violent realization is more ascribed to the way of seasonal planning being modeled. As we described in section 3.3.4.4.2, the seasonal storage is always simulated before the short-term battery. This means that the short-term planning of seasonal storage is conducted first and thus, the seasonal storage would deal with the fluctuations first.

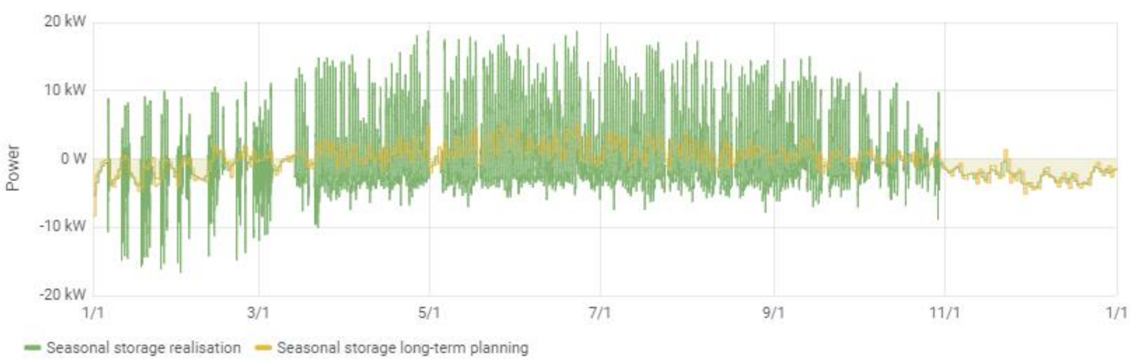


Figure 3-22 An example of the realization (timebase =15 min) and long-term planning (timebase = 1 day) of the seasonal storage power in continuous mode for three tiny houses

Considering the power characteristics of seasonal planning, we use two approaches to specify the discrete power options, namely, evenly-spaced and log-scale methods. The first method returns evenly spaced values within a given interval, and the latter function returns numbers spaced evenly on a log scale. Figure 3-23 gives an example of two methods. As shown in the figure, we first state the ratio of discrete powers to the maximum one. Then we decide the maximum power to create the discrete power vector.

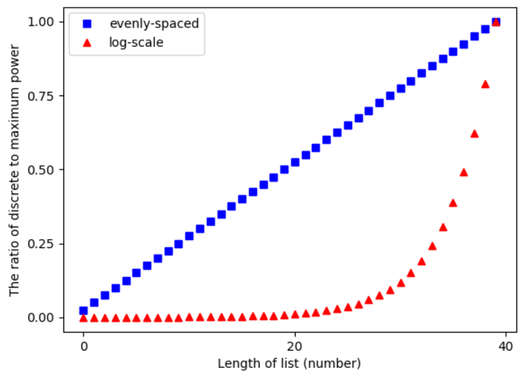


Figure 3-23 The approaches of specifying discrete power, evenly-spaced vs. log-scale (positive part)

The maximum power here is determined to be 20 kW, as all power in continuous mode is below it (see Figure 3-19). The complexity of the discrete power mode method is *O(TM)* where T denotes the number of time intervals, and M denotes the total number of pieces in the piecewise linear approximation of the discrete versions of the problems [72]. The used length of the power vector is 80, which can lead to proper results with adequate simulation speed. To be noticed, the specified power, as input parameters to DEMKit, does not need to scale according to the efficiency, as the unique way of discrete mode dealing with losses. Figure 3-24 presents the simulation results of two methods (both no losses). The evenly-spaced method barely works, at least on seasonal planning with this limited vector length. We deem it is because that the specified power is sparse in the range of seasonal planning powers, when the maximum power is enormous, 20kW, here. Long-term planning power is the average of the whole day ones, which include negative and positive values for import and export power. Plus, the system barely runs the full day and thus leads to a lower power value.

The log-scale one, while, gives similar long-term planning to that of the continuous mode. Moreover, the seasonal buffer SoC follows the plan quite well in general, though the short-term deviation is more significant than that of continuous mode. More peaks of the electricity exchange with the grid appears as well, but the overall profile is more or less the same as that of continuous mode. Therefore, we adopt the log-scale method to set discrete power for the subsequent simulations. The interesting point is that the SoC of seasonal buffer deviates more in summer than winter from the long-term planning. The cause is held to be the fluctuation nature of PV production.

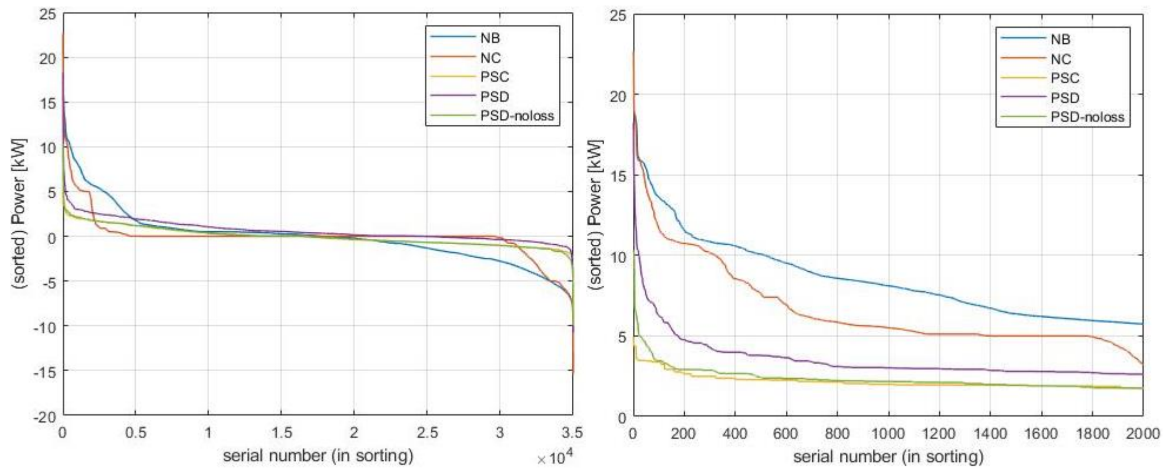
Figure 3-24 The comparison between the simulation profiles of evenly-spaced (upper) and log-scale (lower) discrete power of the seasonal storage with Profile steering and without losses

The simulation result with losses considered is given in Figure 3-25. The long-term planning of seasonal buffer SoC is not well followed in summer. The reason is the priority of the buffer is to balance to the load to some extent and then try to achieve the plan, which can also be seen in the case of evenly-spaced discrete power (see the upper image of Figure 3-24). The SoC following in winter can also confirm the reason. We expect a much smaller deviation between the plan and realization in near-autarkic scenarios.

Figure 3-25 The simulation profiles of the seasonal storage with Profile steering and with losses and log-scale (lower) discrete power

3.3.4.5 Modeling results

Lastly, we compare the electricity exchange with the grid of five scenarios, namely no battery (NB), no control (NC), continuous Profile Steering (PSC), discrete Profile Steering (PSD) and discrete Profile Steering without losses (PSD-noLoss) with values being sorted from high to low (see Figure 3-26). With a hybrid storage system, either controlled or not, the import power is much less. The NC stores and supplies the electricity in a greedy way and therefore leave the head and tail of the curve still steep. While the Profile steering allocates the storage and supply quite well, and its curve is retracted toward the x-axis. The curve of the PSD-noLoss almost overlaps with that of PSC, which stands that our definition of the discrete power is qualified. Even with conversion losses, the PSD still shows a fair amount of improvement. Hereby, our seasonal planning approach is proven to be a success. We anticipate that the seasonal buffer would play a greater role in the scenarios with high penetration of PV panels. The curve suggests that the results tend to approach near autarky. With larger buffer capacities and a well-design hybrid storage system, the fulfill of near autarky is feasible.



(a) Full load duration curve *(b) Zoomed load duration curve*

Figure 3-26 Load duration curves of the power demand of the entire microgrid. (no battery, no control, and with continuous Profile steering, discrete Profile Steering)

4. SIMULATION AND RESULTS

4.1 Introduction

In the first chapter, we raised the main research question and four sub-questions. In the literature research, the potential interactions between users and sustainable technologies are studied. In this section, we continue on the outcome of the second chapter, and model and simulate the energy-saving scenario. The goal is to explore the potential impact of energy-saving behavior on the technical part, notably the sizing of the hybrid storage system. The simulation results of the energy-saving scenario will also be compared with that of the normal/ordinary behavior scenario. The second objective is to explore the extreme scenario when the total load approaches the sum of RE production. The extreme scenario is recognized as the six tiny houses plus a campus EV, which happens to be the first extension of the LIFE project. The third intention is to test the discrete mode (with loss). The simulations for these three targets will be analyzed in the next three sections in order. Among them, we also aim to explore the characteristics of our developed hybrid system. To be clear, the simulations of all objects except the extreme case are based on the model of three tiny houses. The research on all objects, except the discrete mode one, uses continuous mode to do simulations. The first reason is to investigate for the third intention. The second motivation is to explore the possibility of normal-behavior and energy-saving scenarios and compare the results.

As mentioned in section 3.2, we use ALPG to generate the profiles for simulation. We assume three distinct households, namely *Single worker*, *Single part-term*, and *Dual worker (full-time, part-time)*, would live in the tiny houses. For the scenario of six tiny houses, each type of household is assumed to be two. Our expectations and the ALPG profile results for the loads are given in Table 4-1. As estimated in Chapter 2, the energy-saving scenario would conserve around 10% energy, compared to normal-behavior. And we keep the campus EV consumption the same for comparison. Except for these in the modeling chapter, all other components take the default parameters while generating profiles by ALPG. To reach the expectation, a function of scaling the loads in ALPG is used. And the proposed heating demand is met by adjusting the (temperature) setting points of the thermostat. The energy-saving scenario is also generated by scaling the consumption to simulate the behavior change. The RE appliances are fixed, PV panels and a wind turbine, and their yearly production is 22.42 MWh and 6.88 MWh. We believe the different ratio of the total load to production (R_{LtP}) would influence the sizing of the storage system and thus, it is given in the table.

Table 4-1 Loads of average household supposition and three scenarios

Scenarios		Ordinary loads	Heating system	campus EV	Total loads	The ratio of total load to production (R_{LtP})
		[MWh]	[MWh]	[MWh]	[MWh]	%
Supposition	average household	1.5	2.5-3	2-2.5		
Three tiny houses	Normal-behaviour	4.46	8.08	1. 99	14.53	49.59
	Energy-saving	4.02	7.24	1.99	13.25	45.22
Six tiny houses	Normal-behaviour	9.3	15.76	2.2	27.26	93.04

The following settings are default ones for all simulation in this chapter unless particularly declared. The time interval of simulations is one year, and the start time is the 1st of January 2017. The same as stated in section 3.3.4.4.2, the short-term plan interval and horizon are 1 and 2 days, respectively; balancing mode is on for short-term battery but not for seasonal buffer. The initial SoC of short-term battery is 80% of its capacity and its power is half of its capacity in magnitude. We intend to do the simulation close to reality as much as possible and thus, the initial SoC of the seasonal buffer is always calibrated to equal the SoC at the end of the year.

Lastly, we introduce the nomenclature for the simulation scenarios in this chapter (see Table 4-2). The nomination consists of different elements (maximum four) that separated by dash '-'. Unless noted in remarks, the scenario is based on three tiny houses and normal behavior. An example is 'PSHBC-60-6000-NL', which stands for the scenario of continuous profile steering control with a hybrid storage system, 60 kWh short-term battery and 6000 kWh seasonal storage and no loss.

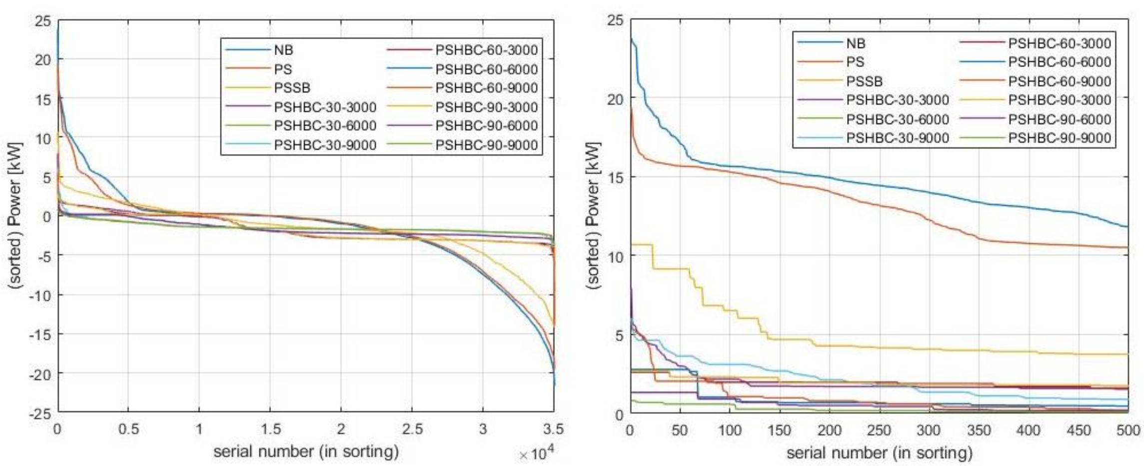
Table 4-2 The nomenclature of simulation scenarios

Place	First			Second	Third	Fourth
Options	Control algorithm	Storage system	Power mode	Short-term buffer capacity [kWh]	Seasonal buffer capacity [kWh]	Remarks
	NC (no control)	SB (short- term buffer)	C (continuous)			SBNC (short-term buffer not controlled)
	PS (Profile steering)	HB (hybrid buffer system)	D (discrete)			NL (no loss)
						ES (energy-saving)
						ST (six tiny houses)

4.2 Three tiny houses

4.2.1 Normal-behavior scenario

Firstly, we simulated the scenarios of NC, NB, and PSSB as control groups. The sorted profiles of the electricity exchange with the grid are given in Figure 4-1 and Table 4-3 shows the statistic results. The only PS can increase the DoA by over 10 percent points (for the loads of white goods and heating, as well as for the rest of results in this thesis). PS-SB can almost double the DoA compared to NB, although still a bit far from near autarky. The excess RE production, 'export' in the table, is far enough to support DoA. A decent-sized seasonal buffer is needed to shift the electricity supply from the grid to buffer.



(a) Full load duration curve *(b) Zoomed load duration curve*

Figure 4-1 Load duration curves of the power demand of the entire microgrid for all simulation in this section

We then explore the PSHBC with various combinations of short-term and seasonal storages. Figure 4-2 visualized the DoA results in 3D plots. It suggests that the increase of the seasonal buffer capacity is more effective than that of short-term battery. The improvement of DoA tends to decrease with the augment of seasonal buffer capacity. Double the seasonal buffer capacity from 3000 kWh can (almost) achieve near autarky (over 98%), while tripling can only enhance DoA by around 1 percent point.

Table 4-3 Statistic results of the simulations for three tiny houses with normal user behavior

Control algorithm	Short-term buffer SoC	Seasonal buffer SoC	Import	Export	DoA	Share in electricity supply [%]			Buffer exchange with grid [MWh]	
	[kWh]	[kWh]	[MWh]	[MWh]	[%]	Grid	RE	Buffer	Import	Export
NB			9.33	24.07	35.8	64.2	35.8			
PS			7.67	22.41	47.2	52.8	47.2			
PSSB	30		4.56	19.31	68.6	17	44.7	38.3	2.1	4.43
NC	60	6000	1.98	14.8	86.4	11.8	35.8	52.4	0.26	1.56
PSHBC	30	3000	1.58	16.16	89.2	6.3	39	54.7	0.66	8.32
		6000	0.43	14.78	97.1	2.2	39.2	58.6	0.11	7.62
		9000	0.32	13.34	97.8	2.1	39.4	58.5	0.02	7.04
	60	3000	1.42	16.03	90.2	5.4	38.9	55.8	0.63	8.23
		6000	0.27	14.62	98.2	1.3	38.8	59.9	0.08	7.57
		9000	0.14	13.15	99.1	0.6	38.7	60.7	0.04	7.04
	90	3000	1.42	16.02	90.2	5.4	38.8	55.7	0.63	8.24
		6000	0.25	14.59	98.3	1.2	38.7	60.1	0.08	7.57
		9000	0.04	13.04	99.8	0.2	38.7	61.1	0.01	6.95

As for short-term battery, although it plays a minor role in improving DoA at the R_{LtP} of 49.59%, it indeed reduces the amount of electricity imported from the grid to the buffer system (see Table 4-3). Note that the buffer would import electricity from the grid in off-peak time and supply the load in peak hours to flat out the overall profile. Here we believe the import to buffer from the grid is mainly contributed by the short-term battery since we use perfect prediction, as mentioned in section 3.3.4.1, to plan the seasonal SoC, plus the production is quite sufficient in this scenario. It is also noticeable that the DoA of PSHBC-60-3000 and PSHBC-90-3000 is the same, due to the rounding. The short-term buffer is necessary not only on the sense of tackling the fluctuation but also support the seasonal buffer to reach its target SoC of the long-term plan.

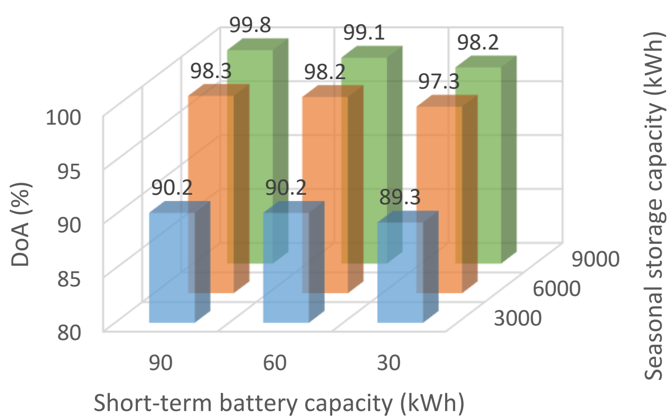


Figure 4-2 The 3D plot of the DoA of all PSHBC scenarios

Figure 4-3 is an example of profile results of a near-autarkic scenario, PSHBC-60-6000. The buffer system just supplied winter demand and smoothed the export of surplus RE production as well, resulting in a relatively smooth S-shape curve of the seasonal buffer SoC over the full year. This kind of S-shape curve is also regarded as a sign of near-autarkic scenario with proper sizing of the buffer system. The storage of the season buffer just touches down the ground and meanwhile keeps the import nearly zero around. Besides, the top of the S-shape curve (almost) reach the capacity of the buffer, which signifies that the seasonal buffer is fully exploited.

Figure 4-3 The comparison between the simulation profiles of NC-60-6000 (upper) and PSHBC-60-6000 (lower) (an example of near-autarkic scenario)

Lastly, we compared the results of NC-60-6000 and PSHBC-60-6000 (see Figure 4-3 and Table 4-3). Despite that the seasonal buffer SoC curve of no control case shows it supplies electricity in winter and charges in summer, the fluctuation and import part of power exchange with the gird is significant in contrast to that of PS case. The implement of PS prompts around 12 percent points enhancement of DoA for this case as well for energy-saving scenarios (see Table 4-4).

4.2.2 Energy-saving scenario

The same as the normal-behavior scenario, we start the simulation with the PSHBC-30-3000-ES. However, the maximum (planned) SoC does not reach its capacity 3MWh for seasonal buffer (see Figure 4-4). That is abnormal as the seasonal storage could have import more electricity in the summer to prevent import in the winter. The default long-term plan interval and horizon are six months and one year, respectively. First-time seasonal planning (at 1st January) would only consider to balance the demand and supply in 2017. The second time long-term planning (on 1st July) would instead consider both the second half-year of 2017 and the first half-year of 2018. Hence, We then speculate that the second-time long-term planning is too late to import enough energy for the consumption of succeeding 12 months. Hence, part of the energy stored in the seasonal buffer would remain for the first 6 months of 2018.

Figure 4-4 The simulation profiles of PSHBC-30-3000

Therefore, we simulated the PSHBC-30-3000-ES again, but with the long-term plan interval of 3 months. Besides, we plotted the long-term planning of seasonal SoC for both PSHBC-30-3000-ES and PSHBC-30-3000-3m-ES in Figure 4-5. The maximum seasonal buffer SoC is enhanced with a smaller long-term plan interval. The static results are in Table 4-4. The DoA also increased around 1 percent point with a long-term plan interval halved. Therefore, we use 3 months as the seasonal plan interval for the rest simulations in this section. For the simulations in the normal user behavior scenario, this phenomenon would not happen, as the loads in the second half-year are higher than that of the energy-saving scenario, which would influence the plan.

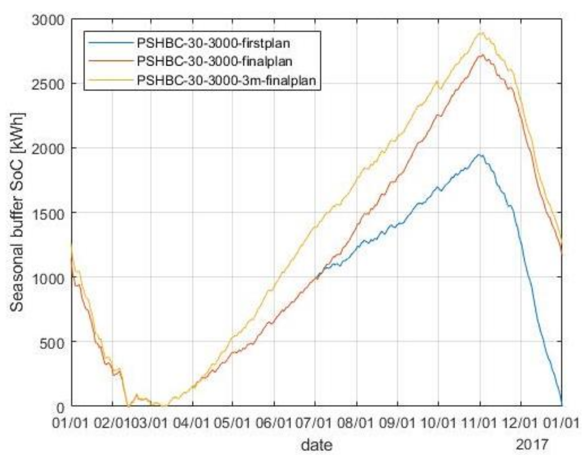


Figure 4-5 The long-term planning of seasonal storage SoC with plan interval (6 months vs. 3months) for PSHBC-30-3000

We increased the seasonal buffer capacity to 4 MWh and tried to achieve near autarky. The simulation profile result shows that the maximum seasonal buffer SoC is still the same as that of PSHBC-30-3000-3m-ES (see Figure 4-6) as well as the DoA (see Table 4-4). The reason is that the prediction is assumed to be perfect and PS would only make a plan that just satisfies the plan. In other words, in the plan, the buffer would not import extra energy as for demand, despite the buffer is still not full, likewise for export. We name it the feature of 'just'. Besides, the SoC curve also conforms with the feature of S-shape we described in section 4.2.1. Therefore 3 MWh is reckoned to be ample for seasonal buffer helping DoA to reach its limit in this case.

One method to mitigate the problem is to adopt a higher capacity of short-term battery. However, the escalation of short-term battery capacity is not very efficient after 60 kWh (see Table 4-4). The DoA only increases by 0.6 percent point when the capacity raised from 60 kWh to 90 kWh. That indicates the energy-saving scenario can not be near-autarkic with a decent size of the hybrid system under the current configuration.

Table 4-4 Statistic results of the simulations for three tiny houses with energy-saving user behavior

Control algorithm	Seasonal plan interval	Short battery SoC	Seasonal buffer SoC	Import	Export	DoA	Share in electricity supply [%]			Buffer exchange with grid [MWh]	
	[Months]	[kWh]	[kWh]	[MWh]	[MWh]	[%]	Grid	RE	Buffer	Import	Export
NB-ES				6.33	22.34	52.3	47.7	52.3			
NC-ES				2.78	17.6	80.9	17.6	35.8	46.6	0.22	1.37
PSHBC-ES	6	30	3000	1.07	17.04	91.9	4.8	41.4	53.8	0.44	8.76
	3		3000	0.95	16.95	92.8	4.2	41.4	54.3	0.39	8.73
			4000	0.94	16.92	92.9	4.3	41.4	54.3	0.37	8.7
		60	3000	0.77	16.76	94.2	3.5	41.4	55.1	0.3	8.58
		90	3000	0.69	16.66	94.8	3.2	41.4	55.5	0.27	8.5

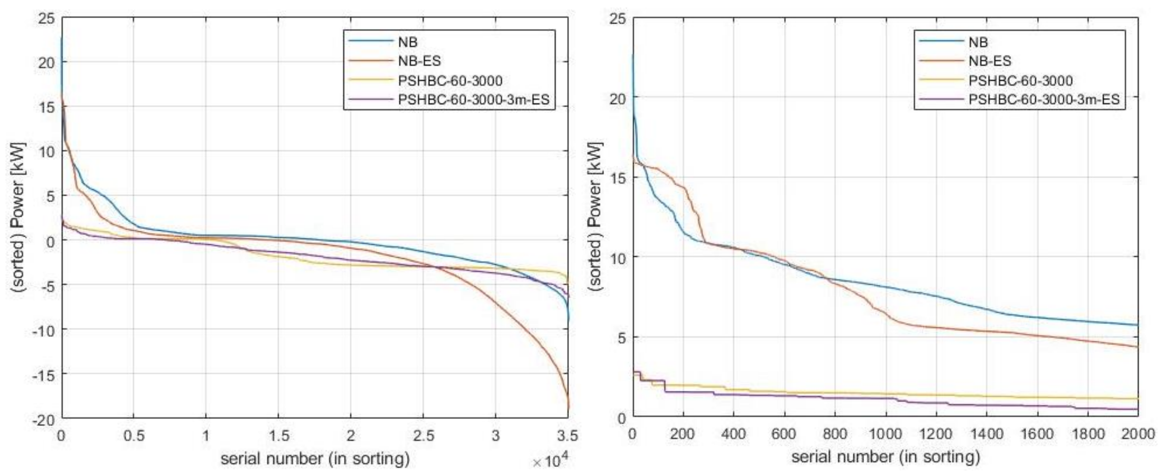
While still a small amount of energy needs to be imported from the grid, it is probably because of the way that the buffer control algorithm is configured (see section 3.3.4.4.2), which causes the SoC can not match with its (seasonal) plan exactly but follow the trend instead. This deviation would either constrain the export of the buffer or impel the buffers to import electricity from the grid in order to satisfy the plan. Therefore, The grid would supply a small amount of energy to the loads as well as to the buffer. Besides, the deviation is deemed to be mainly contributed by the intentionally induced prediction error through differing the plan interval and horizon (see section 3.3.4.4.2).

Figure 4-6 The simulation profiles of PSHBC-30-4000-3m

Raising the initial SoC of the seasonal buffer is not viable for improving DoA as well. Figure 4-7 shows the simulation profiles of PSHBC-30-3000 but with initial SoC set to 2.4 MWh, in contrast to the default one of 1.26 MWh (see Figure 4-4). It can be seen that the seasonal buffer SoC does not change at the end of the year. Instead, the minimum and maximum SoC is enhanced, which attributes to the long-term planning feature of ‘just’ as mentioned above. Therefore, we stick to adjusting the initial SoC equal to end SoC in the simulations as stated in section 4.1.

Figure 4-7 The simulation profiles of PSHBC-30-3000 with initial SoC set to 2.4 MWh

Finally, we compare the simulation results of normal and energy-saving user behavior scenarios (see Figure 4-8 and Table 4-4). The DoA of NB-ES is 16.5% higher than that of NB (35.5%). With a relatively small seasonal buffer, 30 MWh, the energy-saving scenario can already have a comparatively high DoA. For example, the DoA of PSHBC-60-3000 is only 90.2%, while the DoA of PSHBC-60-3000-3m-ES is 94.2%. The data also imply that the energy-saving behavior, together with smart appliances, can lead up to 4 percent points enhancement of DoA, compared to that of normal behavior one. Notwithstanding, the DoA ceiling of the energy-saving scenario is around 95% as well, limited by the current configuration of the hybrid storage system.



(a) Full load duration curve *(b) Zoomed load duration curve*

Figure 4-8 Load duration curves of the power demand of the entire microgrid: normal behavior scenario vs. energy-saving scenario

4.3 Six tiny houses (extreme scenario)

In this section, we seek the possibility of near autarky in extreme cases, which means the loads approach to production or even equal. The envisioned second phase of the LIFE contains six tiny houses, which results in the total loads is 27.26 MWh against the yearly production of 29.3 MWh. We deem this to be the ideal case to see how well the system can cope. The simulation starts with PSHBC-90-9000-ST, the same buffer system setting as the maximum buffer capacities of three tiny houses. The DoA of PSHBC-90-9000-ST is 90.4%, which implies the potential feasibility of near autarky.

Table 4-5 Statistic results of the simulations for six tiny houses scenario

Control algorithm	Short-term battery SoC	Seasonal buffer SoC	Import	Export	DoA	Share in electricity supply [%]			Buffer exchange with grid [MWh]	
	[kWh]	[kWh]	[MWh]	[MWh]	[%]	Grid	RE	Buffer	Import	Export
PSHBC-ST	90	9000	2.64	4.78	90.4	6.2	33.3	60.5	0.95	2.49
	210	12000	1.45	2.21	94.7	3.6	33.4	63	0.46	1.19
	270	18000	1.48	1.92	94.6	3.6	33.6	62.8	0.5	1.18
PSHBC-ST -SBNC	20	12000	0.26	2.03	99.1	0.8	32.8	66.4	0.03	1.16
	30		0.22	1.99	99.2	0.7	32.8	66.5	0.03	1.12

We continue with PSHBC-210-12000-ST and PSHBC-270-18000-ST to investigate the upper limit of DoA for this scenario. Figure 4-9 shows the simulation profiles of the two cases. The electricity exchange with the grid for both cases fluctuates around zero as expected. The curve of seasonal buffer SoC for PSHBC-270-18000-ST is still in good S shape, but elevated into the 'air', owing to almost equality of the total load and production. That suggests no matter how higher the seasonal buffer capacity (than 12000 kWh), the DoA would not vary.

Figure 4-9 The comparison between the profile results of PSHBC-210-12000-ST (upper) and PSHBC-270-18000-ST (lower)

The outcome shows that the DoAs of these two cases are more or less the same, close to 95% (see Table 4-5). In other words, the DoA ceiling for the extreme scenario is around 95%, which is equivalent to that of the energy-saving scenario of three tiny houses. Consequently, we hold that the explanation of the DoA ceiling for the extreme case is the same as that of the energy-saving scenario of three tiny houses (see section 4.2.2).

In order to confirm the conjecture, we force the short-term battery uncontrolled but still regulate the seasonal buffer. The simulate results of these PSHBC-ST-SBNC cases are given in Table 4-5. It suggests that 12000 kWh seasonal storage with a comparatively small short-term buffer, 20 kWh can facilitate the extreme scenario to become near-autarkic with the DoA of 99.1% (see Figure 4-10). The remaining import in the winter is attributed to the seasonal buffer configuration.

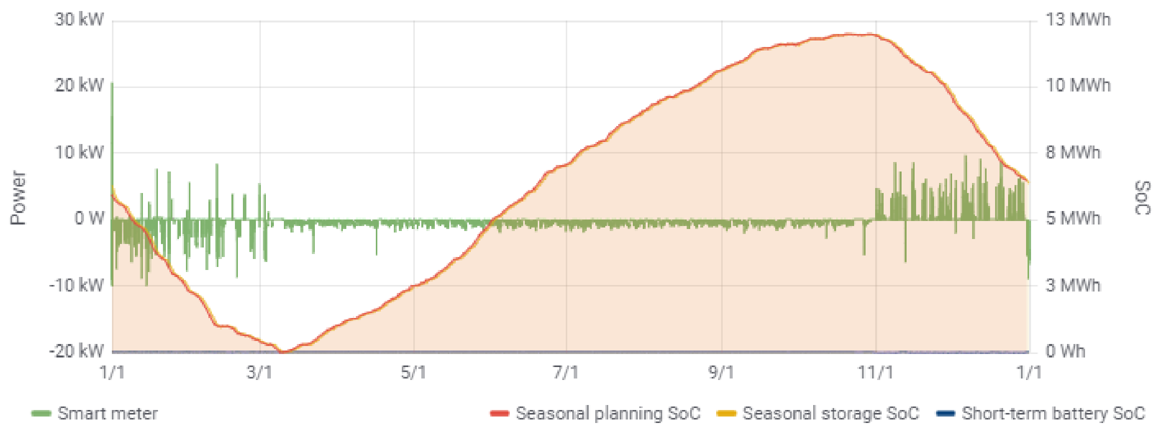


Figure 4-10 The simulation profiles of PSHBC-20-12000- ST-SBNC

We also exert the same trick on the normal-behavior and energy-saving scenarios of three tiny houses, and it indeed improves the DoA (see Table 4-6). Nonetheless, this kind of configuration requires the prediction to be very accurate. Otherwise, it would have an adverse effect on DoA.

Table 4-6 Statistic results of the simulations for the normal-behavior scenario of three tiny houses with the short-term battery not controlled

Control algorithm	Short-term battery SoC	Seasonal buffer SoC	Import	Export	DoA	Share in electricity supply [%]			Buffer exchange with grid [MWh]	
	[kWh]	[kWh]	[MWh]	[MWh]	[%]	Grid	RE	Buffer	Import	Export
PSHBC -SBNC	10	9000	0.1	13.1	99.3	0.7	38.3	61	0	7.1
	20		0.01	13.01	100	0	38.3	61.7	0	7.01
	30		0	13	100	0	38.3	61.7	0	7.01
	60	6000	0.12	14.45	99.1	0.6	38.4	61	0.04	7.56
PSHBC -SBNC-ES	90	3000	0.4	16.39	97	2.1	40.9	56.9	0.12	8.42

4.4 Discrete power mode with seasonal buffer loss

The precedent simulations do not account for the conversion losses with the seasonal buffer, which cannot be neglected for practical cases. In this section, the conversion efficiencies of the seasonal buffer are taken into account by the discrete power mode of DEMKit. The efficiency for discharging and charging is 45% and 65%, respectively, as mentioned in section 3.3.4.2. To be noticed, the discrete mode is only valid for the control algorithms so far. That means the no PSHBD-NB case for comparison. The subsequent simulations are based on the normal-behavior scenario of three tiny houses. For the reason that the losses are hard to estimate in advance, especially when using the discrete mode, and would vary along with the change of buffer system, we will hence only give one simulation as a reference.

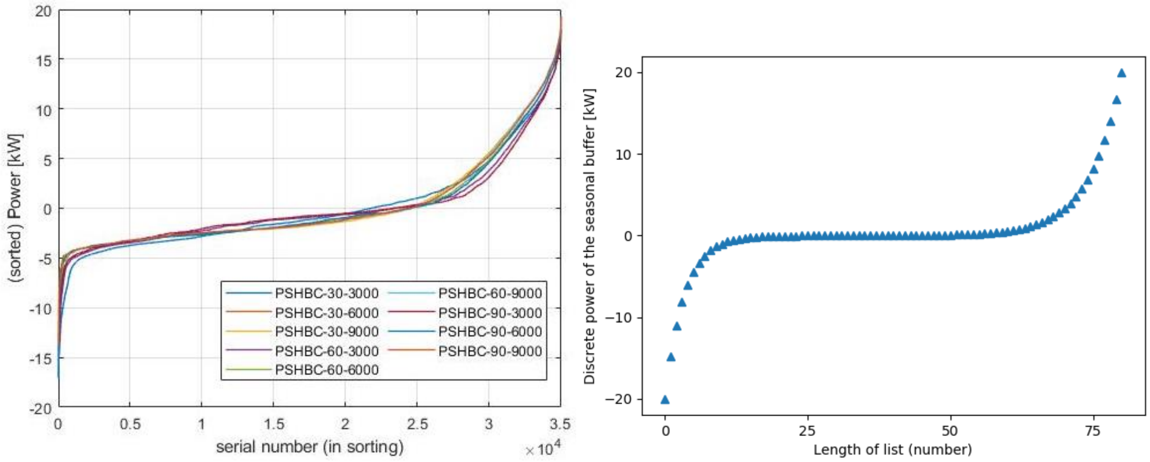


Figure 4-11 The sorted seasonal buffer power for all the cases of PSHBC in the three tiny houses normal behavior scenario vs. compile discrete power

We plotted the seasonal buffer power (with values being sorted) for all the cases of PSHBC in the three tiny houses normal behavior scenarios and attempted to compose a discrete power vector as similar as possible to allow the seasonal buffer operate like continuous mode as much as possible (see Figure 4-1). The log-scale method, introduced in the modeling chapter, is used to specify the discrete power. We first simulated the PSHBD-60-6000-NL to compare the discrete and continuous mode and also to provide the reference for losses-considered continuous mode for future work. The simulation results of the profiles resemble that in the modeling section (see section 3.3.4.4.3). The DoA of PSHBD-60-6000-NL, 94.4% drops a bit, in contrast with the DoA of PSHBC-60-6000, 98.2%, which is reasonable.

Table 4-7 Statistic results of the simulations for six tiny houses

Control algorithm	Short battery SoC	Seasonal buffer SoC	Import	Export	DoA	Share in electricity supply [%]			Buffer exchange with grid [MWh]		Loss
	[kWh]	[kWh]	[MWh]	[MWh]	[%]	Grid	RE	Buffer	Import	Export	[MWh]
PSHBD	60	9230	3.13	3.71	78.5	8.7	40.4	50.9	1.87	1.19	14.18
PSHBD-NL	60	6000	0.82	15.14	94.4	2.2	39.2	58.6	0.5	8.01	

We scaled the seasonal buffer SoC from 6000 kWh to 9230 kWh based on charging efficiency 65%, trying to accomplish near autarky. The simulation profile of seasonal buffer SoC shows a good S-shape curve (see Figure 4-11), which represents the optimal DoA of this scenario, as argued in section 4.2.1. The DoA results in 78.5%, dropping a lot compared to that of PSHBD-60-6000-NL (see Table 4-5). The overall loss is enormous around 14.18 MWh, around 50% of RE production, which is yet not unexpected with a low round-trip efficiency of the seasonal buffer.

The significant losses turn this scenario into an extreme case. The sum of the total load and the loss is 28.71 almost equivalent to the total production of 29.3 MWh. The relatively straight profile of the electricity exchange with the grid further identifies the extremity of this case. Aside from the current configuration of the hybrid storage system (see section 4.2.2), the discrete power further worsen the degree of autarky. However, based on the results of the extreme scenario (see section 4.3), we believe if this scenario is simulated by continuous mode with losses involved, the DoA may approach 95% and with the prediction error tackled, 'soft-islanding' perhaps be possible.

Figure 4-12 The simulation profiles of PSHBD-60-9230

The magnificent losses can be potentially reduced. As shown in Figure 3-22, the realized seasonal buffer power is extremely violent. The reason is that the seasonal storage would tackle the fluctuations in advance to short-term buffer under the current configuration (see section 3.3.4.4.3). A conceivable solution is to force the planning of short-term battery being simulated first, instead of the seasonal storage, but meanwhile still keeps the sequence of realization unchanged.

5. CONCLUSIONS AND RECOMMENDATIONS

5.1 Conclusions

In this thesis, we successfully built a model of the LIFE project and explored the possibility of near autarky in multiple scenarios. We conclude the study by answering the research questions proposed in Chapter 1.

1) *What are other state-of-the-art sustainable technologies that can be included to improve system DoA?*

   Aside from the renewable technologies that are already proposed and the infrared heating system in the tiny houses, HEMS and smart appliances would make a decent contribution to increasing the DoA of the system. Popular smart appliances include smart plugs, smart lighting systems, smart thermostats. These intelligent products can free people from trivial matters and at the same time save energy. Besides, the Profile Steering control algorithm, storage system, and their combination can largely increase the degree of autarky of the system. In addition to these physical solutions, energy conservation strategies are effective at promoting user behavior change towards sustainability.

2) *How would user behavior change with these technologies being applied, and how would these changes influence power balance in the microgrid?*

   The energy conservation strategies typically target at energy-saving and load-shifting. However, the problem is that the behavior change would rebound over time. In terms of smart appliances, despite most users would keep them on to save energy, few users would disable them, owing to the problems of aesthetics inharmony, inconvenience, complexity, comfort level changing. More importantly, users tend to consume more energy after adopting RE technologies, which is known as the rebound effect.

   The literature study suggests that the energy-saving efficacy of smart appliances are: 39.5% for the smart lighting system on average, approximate 14.07% for Heating, ventilation, and air conditioning (HVAC), and 16.66% for other products. The application of energy conservation strategies is around 10% on average. Taking into the rebound effect by about 18%. We conservatively estimate that with these technologies being applied, around 10% of the consumption would be saved for a household for the LIFE, which results in a 16.5 percent point increase of DoA. With PS and hybrid storage system applied, the energy-saving scenario can have up to 4 percent points DoA lead, compared to that of normal-behavior scenario.

3) *What is needed from the technical part to support people in making these social changes?*

   As to energy conservation strategies, a customizable and functional display or an online portal is vital as the feedback agency. The accessibility and robustness of smart appliances and their services are crucial for user adoption and behavior change. Simplicity or the degree of automation of the smart products play a role in determining the continuation of developed sustainable behaviors.

4) *How can the optimal size of these technologies be determined in the light of system integration?*

   For this question, we only focus on the sizing of the hybrid storage system, since the size of other components is already determined. The optimal seasonal buffer size ought to be determined first, as the amount of energy that needs to be imported from the grid largely depends on it. The penetration of different RE technologies that determines the demand and supply profile is one of the main factors that influence the seasonal battery sizing. The other factor is the ratio of total loads to production. For example, the energy-saving scenarios only need 3 MWh long-term storage to reach its maximum DoA, whereas the normal-behavior scenario requires at least 6 MWh seasonal buffer to achieve near autarky.

The capacity of the seasonal buffer should be evaluated by the yearly curve of the seasonal buffer SoC to see if it is in S-shape or not. The import in winter should also be taken into consideration. As long as the raise of the capacity can not lead to the reduction of winter imports, the capacity might be the ideal one. Once an optimal seasonal buffer is determined, the sizing of the short-term battery should be decided based on prediction error, as described in Chapter 4.

5) *What is needed from the technical part to create an autarkic field lab?*

The first indispensable component is the Profile Steering algorithm, which can lead to over 10 percent points raise of DoA alone and around 12 percent points with a hybrid buffer system. Plus the hybrid buffer system, the DoA of the microgrid can be uplifted to a new level. While the seasonal buffer loss being ignored, a 60 kWh short-term battery and 6000 kWh can turn the normal-behavior scenario to near-autarkic. However, when the loss is counted, with the optimal seasonal buffer capacity 9230 kWh and a 60 kWh short-term battery, the normal-behavior scenario can only achieve 78.5% DoA with the present hybrid storage configuration, largely because the losses are massive, around 50% of total production, so as to turn the scenario to an extreme case.

In addition, recommendations on improving the current storage model are, integrating loss into the continuous power mode of DEMKit and tackling prediction errors (95% ceiling problem). A possibility exists that if these issues were solved, the normal-behavior scenario of three tiny houses may achieve ‘soft-islanding’ with a larger long-term buffer.

5.2 Recommendations

In the literature study, we mainly investigate the potential interactions between consumers and sustainable technologies, especially the figures of the induced savings and increase of consumption. Nonetheless, the underlying theories are found to be fruitful and are expected to provide more insights for a better design of the project.

In this thesis, the campus EV is assumed to be used once a day. A more realistic EV consumption profile that allows multiple outings a day is expected. The outings are to supposed to start and end randomly, but the time overlap among events needs to be avoided. Moreover, the UIHS does not have a buffer and consequently the PS can not be applied to it. Taking advantage of the house structure and furnishes etc. that has the thermal capacity and modeling them as a buffer is a feasible way to further improve DoA.

As discussed in the Conclusions, the current configuration of the hybrid buffer system has some limits and shortcomings. The first one is the way to take into account seasonal buffer loss in DEMKit. The discrete mode has poor performance when losses being involved and we speculate the continuous mode is more friendly with loss. Hereby, integrate the loss into the continuous power mode algorithm is highly recommended.

On the other hand, the prediction error needs to be tackled; otherwise, the DoA ceiling might appear (see section 4.2.2). A potential solution is to force short-term battery planning before seasonal one, and thus, the seasonal buffer is more likely to realize its plan as the short-term battery will deal with the fluctuations first. This method might also reduce the losses of the seasonal buffer by lowering the charging and discharging energy. A more straightforward way to lower losses is to adopt a more efficient long-term buffer. A possible more effective solution is, from the perspective of the 'just' feature of PS (see section 4.2.2), introduce more flexibility for the buffer. The third option is using two separate short-term batteries with one controlled and the other one not.

Furthermore, no control seasonal buffer operation mode is proposed in section 3.3.4.3. In practice, in order to minimize the on-off cycling and increase the lifetime of the system, more complex system control is regularly used [80, 81]. Little et al. [78] investigated the effect of the hysteresis approach on reducing the cycling of the electrolyzer. Figure 5-1 illustrates how hysteresis works: take electrolyzer for example, electrolyzer starts once the short-term battery SoC exceeds the upper setpoint; instead of at the same standard, electrolyzer shuts down at a lower setpoint, likewise for the fuel cell. The difference between these two setpoints refers to hysteresis. Little found that increasing the hysteresis from 5% to 20% can lead to 500 times reduction of electrolyzer on-off cycles.

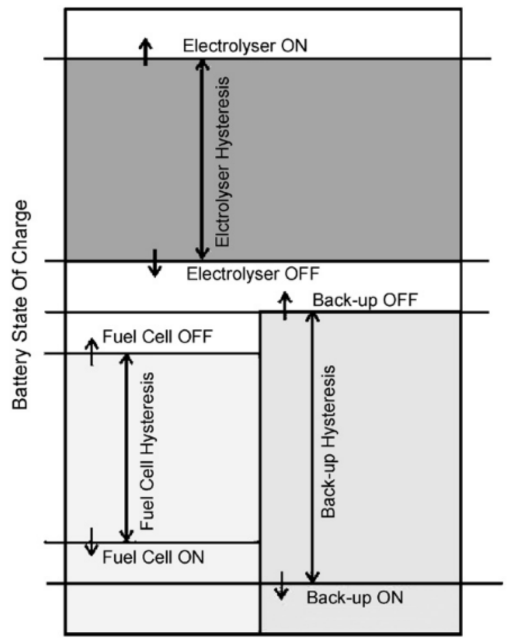


Figure 5-1 Schematic diagram of hysteresis for seasonal storage that depend on the short-term battery SoC [78]

REFERENCES

1. *Sustainable Development Goals*. Ministry of Foreign Affairs of the Kingdom of the Netherlands.
2. B. Homan, G. Hoogsteen, S. Nebiolo, et al., *Maximizing the degree of autarky of a 16 house neighbourhood by locally produced energy and smart control.* Sustainable Energy, Grids and Networks, 2019. 20: p. 100270.
3. T. Hargreaves, M. Nye, and J. Burgess, *Making energy visible: A qualitative field study of how householders interact with feedback from smart energy monitors.* Energy Policy, 2010. 38(10): p. 6111-6119.
4. G. Hoogsteen, *Demkit: a flexible smart grid simulation and demonstration platform written in python.* 2017.
5. G. Hoogsteen, A. Molderink, J.L. Hurink, et al. *Generation of flexible domestic load profiles to evaluate demand side management approaches*. in *2016 IEEE International Energy Conference (ENERGYCON)*. 2016. IEEE.
6. W. Abrahamse, L. Steg, C. Vlek, et al., *A review of intervention studies aimed at household energy conservation.* Journal of Environmental Psychology, 2005. 25(3): p. 273-291.
7. M.S. Pallak and W. Cummings, *Commitment and voluntary energy conservation.* 1976. 2(1): p. 27-30.
8. L.J. Becker, *Joint effect of feedback and goal setting on performance: A field study of residential energy conservation.* Journal of applied psychology, 1978. 63(4): p. 428.
9. R.A. Winett, I.N. Leckliter, D.E. Chinn, et al., *Effects of television modeling on residential energy conservation.* 1985. 18(1): p. 33-44.
10. R.A. Winett, S.Q. Love, and C. Kidd, *The effectiveness of an energy specialist and extension agents in promoting summer energy conservation by home visits.* Journal of Environmental Systems, 1982. 12(1).
11. R.A. Winett, J.H. Kagel, R.C. Battalio, et al., *Effects of monetary rebates, feedback, and information on residential electricity conservation.* 1978. 63(1): p. 73.
12. L. McClelland and S.W. Cook, *Promoting Energy Conservation in Master‐Metered Apartments through Group Financial Incentives1.* Journal of Applied Social Psychology, 1980. 10(1): p. 20-31.
13. R.E. Pitts and J.L. Wittenbach, *Tax credits as a means of influencing consumer behavior.* Journal of Consumer Research, 1981. 8(3): p. 335-338.
14. M.J.J. Handgraaf, M.A. Van Lidth de Jeude, and K.C. Appelt, *Public praise vs. private pay: Effects of rewards on energy conservation in the workplace.* Ecological Economics, 2013. 86: p. 86-92.
15. P.W. Schultz, J.M. Nolan, R.B. Cialdini, et al., *The constructive, destructive, and reconstructive power of social norms.* 2007. 18(5): p. 429-434.
16. S. Darby, *The effectiveness of feedback on energy consumption*. 2006, Environmental change institute: University of Oxford. p. 26.
17. R.G. Bittle, R.M. Valesano, and G.M. Thaler, *The effects of daily feedback on residential electricity usage as a function of usage level and type of feedback information.* Journal of environmental systems, 1979. 9(3).
18. J.H. Van Houwelingen, Van Raaij, W. Fred, *The effect of goal-setting and daily electronic feedback on in-home energy use.* 1989. 16(1): p. 98-105.
19. R.J. Sexton, N.B. Johnson, and A. Konakayama, *Consumer response to continuous-display electricity-use monitors in a time-of-use pricing experiment.* Journal of Consumer Research, 1987. 14(1): p. 55-62.
20. O.I. Asensio and M.A. Delmas, *The dynamics of behavior change: Evidence from energy conservation.* Journal of Economic Behavior & Organization, 2016. 126: p. 196-212.
21. P. Petkov, F. Köbler, M. Foth, et al. *Motivating domestic energy conservation through comparative, community-based feedback in mobile and social media*. in *Proceedings of the 5th International Conference on Communities and Technologies*. 2011. ACM.
22. T. Ploug, P. Hasle, and H. Oinas-Kukkonen, *Persuasive Technology, 5th International Conference, PERSUASIVE 2010, Copenhagen, Denmark, June 7-10, 2010. Proceedings.* 2010.
23. M. Martinez and L.F.J.S.C.E. Stein, *California Information Display Pilot Technology Assessment.* 2004.

24. L.J. Becker, C.J.P. Seligman, and S.P. Bulletin, *Reducing air conditioning waste by signalling it is cool outside.* 1978. 4(3): p. 412-415.
25. I. Ajzen, *The theory of planned behavior.* Organizational Behavior and Human Decision Processes, 1991. 50(2): p. 179-211.
26. P. de Vries, H. Aarts, and C.J.H. Midden, *Changing Simple Energy-Related Consumer Behaviors: How the Enactment of Intentions Is Thwarted by Acting and Non-Acting Habits.* Environment and Behavior, 2011. 43(5): p. 612-633.
27. L.T. McCalley, P.W. de Vries, and C.J.H. Midden, *Consumer Response to Product-Integrated Energy Feedback: Behavior, Goal Level Shifts, and Energy Conservation.* Environment and Behavior, 2010. 43(4): p. 525-545.
28. I. Wittenberg and E. Matthies, *How do PV households use their PV system and how is this related to their energy use?* Renewable Energy, 2018. 122: p. 291-300.
29. H. Hondo and K. Baba, *Socio-psychological impacts of the introduction of energy technologies: Change in environmental behavior of households with photovoltaic systems.* Applied Energy, 2010. 87(1): p. 229-235.
30. Y. Qiu, M.E. Kahn, and B. Xing, *Quantifying the rebound effects of residential solar panel adoption.* Journal of Environmental Economics and Management, 2019. 96: p. 310-341.
31. G. Deng and P. Newton, *Assessing the impact of solar PV on domestic electricity consumption: Exploring the prospect of rebound effects.* Energy Policy, 2017. 110: p. 313-324.
32. L. Havas, J. Ballweg, C. Penna, et al., *Power to change: Analysis of household participation in a renewable energy and energy efficiency programme in Central Australia.* Energy Policy, 2015. 87: p. 325-333.
33. G. Deng and P.J.E.P. Newton, *Assessing the impact of solar PV on domestic electricity consumption: Exploring the prospect of rebound effects.* 2017. 110: p. 313-324.
34. J. Keirstead, *Household behavioural responses to photovoltaic-system monitoring devices.* 2005.
35. R. Ford, M. Pritoni, A. Sanguinetti, et al., *Categories and functionality of smart home technology for energy management.* Building and Environment, 2017. 123: p. 543-554.
36. D. Lee and C.-C. Cheng, *Energy savings by energy management systems: A review.* Renewable and Sustainable Energy Reviews, 2016. 56: p. 760-777.
37. M. Alaa, A.A. Zaidan, B.B. Zaidan, et al., *A review of smart home applications based on Internet of Things.* Journal of Network and Computer Applications, 2017. 97: p. 48-65.
38. B. Von Neida, D. Maniccia, and A. Tweed, *An Analysis of the Energy and Cost Savings Potential of Occupancy Sensors for Commercial Lighting Systems*. Journal of the Illuminating Engineering Society, 2001. 30(2): p. 111-125.
39. I. Chew, D. Karunatilaka, C.P. Tan, et al., *Smart lighting: The way forward? Reviewing the past to shape the future.* Energy and Buildings, 2017. 149: p. 180-191.
40. M. Eilers, J. Reed, and T. Works, *Behavioral aspects of lighting and occupancy sensors in private offices: a case study of a university office building.* ACEEE summer study on energy efficiency in buildings, 1996.
41. L. Heschong, O. Howlett, J. McHugh, et al., *Sidelighting photocontrols field study.* 2005.
42. J.D. Jennings, F.M. Rubinstein, D. DiBartolomeo, et al., *Comparison of Control Options in Private Offices in an Advanced Lighting Controls Testbed.* Journal of the Illuminating Engineering Society, 2000. 29(2): p. 39-60.
43. M.A. ul Haq, M.Y. Hassan, H. Abdullah, et al., *A review on lighting control technologies in commercial buildings, their performance and affecting factors.* 2014. 33: p. 268-279.
44. A. Yasukouchi and K. Ishibashi, *Non-visual effects of the color temperature of fluorescent lamps on physiological aspects in humans.* Journal of physiological anthropology applied human science, 2005. 24(1): p. 41-43.
45. B.-C. Park, J.-H. Chang, Y.-S. Kim, et al., *A study on the subjective response for corrected colour temperature conditions in a specific space.* 2010. 19(6): p. 623-637.
46. C. Aarish and C. Jones, *Smart Thermostats and the Triple Bottom Line: People, Planet, and Profits.* Proceedings of the ACEEE Summer Study on Energy Efficiency in Buildings, Pacific Grove, CA, USA, 2016: p. 21-26.
47. N. Lieb, S. Dimetrosky, and D. Rubado, *Thriller in Asilomar: Battle of the Smart Thermostats.* the ACEEE Summer Study on Energy Efficiency in Buildings, Pacific Grove, CA., 2018. 8.
48. EcoCabins. *Tiny Houses*. Available from: https://www.eco-cabins.com/en/tiny-houses/.

49. M.E.T. Gerards, H.A. Toersche, G. Hoogsteen, et al. *Demand side management using profile steering*. in *2015 IEEE Eindhoven PowerTech*. 2015.
50. G. Hoogsteen, A. Molderink, J.L. Hurink, et al. *Asynchronous event driven distributed energy management using profile steering*. in *2017 IEEE Manchester PowerTech*. 2017.
51. G. Hoogsteen, *A cyber-physical systems perspective on decentralized energy management*. 2017: University of Twente.
52. Eco-term. *Infrared heating film*. Available from: https://irshop.eu/eng/?product=infrared-heating-film-heat-decor-hd-310-80wlm.
53. *80 cm 2 M / 3 M / 4 M 220 V Far Infrared Floor Heating Vinyl Film Crystal Carbon Fiber Floor Heating Film*. Available from: https://www.mijn-artikel.nl/Webwinkel-Product-358096230/80cm-2M-3M-4M-220V-Far-Infrared-Floor-Heating-Film-Crystal-Carbon-Fiber-Underfloor-Heating-Film.html.
54. A. Kavga, T. Panidis, V. Bontozoglou, et al., *Infrared Heating of Greenhouses Revisited: An Experimental and Modeling Study.* Transactions of the ASABE, 2009. 52.
55. P. Caouette, L. Gauthier, B. Casault, et al., *Temperature and radiation distribution in an infrared-heated greenhouse with a polyethylene cover.* 1990. 32: p. 111-116.
56. H. Plus, *Infrared Underfloor Heating*.
57. H. Decor, *Underfloor heating system - Heat Decor heating film.*
58. *Underfloor heating – Arkon Heating*. Available from: http://www.arkonheating.com/infrared-heating-film/.
59. Tremofol, *The technologically advanced infrared heating film.*
60. Calorique. *Radiant Heating Film Calorique - installation and operating instructions*. Available from: https://calorique.info/images/MontageAnleitung/Manual_Calorique_U.pdf.
61. A.M. Flickstein, *Infrared Thermal System For Whole-Body Regenerative Radiant Therapy*
62. N. M., *Thermal camera testing, infrared vs. hot air fan heater*.
63. R.P.v. Leeuwen, *Towards 100% renewable energy supply for urban areas and the role of smart control*. 2016, University of Twente: Enschede. p. 59.
64. O. Glass. *Technical Data - Triple Glazing*. Available from: http://www.olympicglass.co.uk/ewcommon/tools/download.ashx?docId=670.
65. *Climatology - Hourly weather data in the Netherlands*. Available from: http://projects.knmi.nl/klimatologie/uurgegevens/.
66. *THE AMPERAPORT*. Available from: https://amperapark.com/products.
67. *Twente airport explanation - applicable legal planning regulations*. Available from: https://www.ruimtelijkeplannen.enschede.nl/NL.IMRO.0153.BP00001-0002/t_NL.IMRO.0153.BP00001-0002.html.
68. C. Eisenhut, F. Krug, C. Schram, et al., *Wind-Turbine Model for System Simulations Near Cut-In Wind Speed.* IEEE Transactions on Energy Conversion, 2007. 22(2): p. 414-420.
69. IRENA. *Map: DTU Global Wind Atlas 1km resolution*. Available from: https://irena.masdar.ac.ae/gallery/#tool/10%E2%80%8B.
70. I. Troen and E.L. Petersen, *European wind atlas*. 1989: Risø National Laboratory.
71. KNMI. *Potential wind speed of the Netherlands - Download*. Available from: http://projects.knmi.nl/klimatologie/onderzoeksgegevens/potentiele_wind/index.cgi?language=eng.
72. T.v.d. Klauw, *Decentralized energy management with profile steering : resource allocation problems in energy management*. 2016, University of Twente: Enschede.
73. A. Maleki and A. Askarzadeh, *Artificial bee swarm optimization for optimum sizing of a stand-alone PV/WT/FC hybrid system considering LPSP concept.* Solar Energy, 2014. 107: p. 227-235.
74. C.-H. Li, X.-J. Zhu, G.-Y. Cao, et al., *Dynamic modeling and sizing optimization of stand-alone photovoltaic power systems using hybrid energy storage technology.* Renewable Energy, 2009. 34(3): p. 815-826.
75. A. Dicks and D.A.J. Rand, *Fuel cell systems explained*. 2018: Wiley Online Library.
76. K. Agbossou, M.L. Doumbia, and A. Anouar. *Optimal hydrogen production in a stand-alone renewable energy system*. in *Fourtieth IAS Annual Meeting. Conference Record of the 2005 Industry Applications Conference, 2005.* 2005.
77. D. Stolten and B. Emonts, *Hydrogen Science and Engineering, 2 Volume Set: Materials, Processes, Systems, and Technology*. Vol. 1. 2016: John Wiley & Sons.

78. M. Little, M. Thomson, and D. Infield, *Electrical integration of renewable energy into stand-alone power supplies incorporating hydrogen storage.* International Journal of Hydrogen Energy, 2007. 32(10): p. 1582-1588.
79. L. Valverde, D. Ali, M. Abdel-Wahab, et al. *A technical evaluation of Wind-Hydrogen (WH) demonstration projects in Europe*. in *4th International Conference on Power Engineering, Energy and Electrical Drives*. 2013.
80. F.J. Vivas, A. De las Heras, F. Segura, et al., *A review of energy management strategies for renewable hybrid energy systems with hydrogen backup.* Renewable and Sustainable Energy Reviews, 2018. 82: p. 126-155.
81. E.L.V. Eriksson and E.M. Gray, *Optimization and integration of hybrid renewable energy hydrogen fuel cell energy systems – A critical review.* Applied Energy, 2017. 202: p. 348-364.

Appendix A DoA analysis in python code

```python
# DoA analysis code for no buffer scenarios (Python code)

# Parameters setup
workSpacePath = 'C:/Users/D/demkitsim/Dcode/Data/Tinyhouse/DoA/'

schemes = [10,11]

# Define source file path
cnt = 0
sourceFilePath = {}

for scheme in schemes:
   sourceFilePath[cnt] = workSpacePath + str(scheme) + '.csv'
   cnt +=1

# Define function parameters
sBeginL = 1
sEndL = -1

smField = 1
pvField = 2
wtField = 3

cntLoop = 0
currentLine = 0
cnt = 0

DoAtotal = 0
DoAcnt = 0

while cntLoop <= (len(schemes)-1):

   # Reset parameters each loop
   currentLine = 0
   cnt = 0
   Eretoloadtotal = 0 #renewable energy to load
   Eloadtotal = 0
   Epvtotal = 0
   Ewttotal = 0

   Eimporttotal = 0
   Eexporttotal = 0
   Eimport = 0
   Eexport = 0

   DoAtotal = 0
   DoAcnt = 0

   # Read data from file
   sfp = sourceFilePath[cntLoop]
   with open(sfp, 'r') as sf:

      for line in sf:
         # stop when reach desired end line
         if sEndL != -1:
            if currentLine >= sEndL:
               exit()

         # put each line values into a list into fields
         fields = line.split(",")
```

```
         if currentLine >= sBeginL:
            # Import data
            Epv = fields[pvField]
            Epv = float(Epv)
            Epvtotal += Epv

            Ewt = fields[wtField]
            Ewt = float(Ewt)
            Ewttotal += Ewt

            Esm = fields[smField]
            Esm = float(Esm)

            # Calculating data
            # Eimport and Eexport
            Eimport = 0
            Eexport = 0
            if Esm > 0:
               Eimport = Esm
               Eimporttotal += Esm
            else:
               Eexport = Esm
               Eexporttotal += Esm

            # Eload
            Eload = Esm - Epv - Ewt
            Eloadtotal += Eload

            # Epv and Ewt supply to load, dont know each contribution
            Eretoload = min(Eload, -(Epv+Ewt))
            Eretoloadtotal += Eretoload

            # Update cnt number each for-loop
            cnt += 1
            currentLine += 1

         else:
            currentLine += 1

      print("Scheme:" + str(schemes[cntLoop]))

   #DoA
   DoA = (Eloadtotal-Eimporttotal) / Eloadtotal
   print('DoA:' + str(DoA) + '\n')

   # Results for table
   print(str(round(Eimporttotal / 4 / 1e6,2)))      #unit kWh, one decimel
   print(str(round(-Eexporttotal/ 4 / 1e6,2)))
   print(str(round(DoA * 100,1)))

   # Share in electricity supply
   print(str(round(Eimporttotal /Eloadtotal*100,1)))
   print(str(round(Eretoloadtotal/Eloadtotal*100,1)))

   cntLoop += 1

exit()
```

```python
# DoA analysis code for buffer-applied scenarios (Python code)

# Parameters setup
workSpacePath = 'C:/Users/D/demkitsim/Dcode/Data/Tinyhouse/DoA/'

schemes = ['13_30_3000', '13_30_6000']

# Define file path
cnt = 0
sourceFilePath = {}
discrete = True
efficiency = [0.45,0.65]        #efficiency for dischargin(negative) and
charging (postive)

for scheme in schemes:
   sourceFilePath[cnt] = workSpacePath + str(scheme) + '.csv'
   cnt +=1

# Define function parameters
sBeginL = 1
sEndL = -1

smField = 1
pvField = 2
wtField = 3
shortBfield = 6
seasonalBfield = 8   #None

cntLoop = 0
currentLine = 0
cnt = 0

DoAtotal = 0
DoAcnt = 0

while cntLoop <= (len(schemes)-1):

   # Reset parameters each loop
   currentLine = 0
   cnt = 0

   loss = 0

   Eloadtotal = 0
   Epvtotal = 0
   Ewttotal = 0
   Eretoloadtotal = 0 #renewable energy to load

   # buffer
   Eretobuffertotal = 0
   EshortBtoloadtotal = 0
   Egridtobuffer = 0
   Egridtobuffertotal = 0
   Ebuffertogrid = 0
   Ebuffertogridtotal = 0

   Egridtoload = 0
   Egridtoloadtotal = 0

   Eimporttotal = 0
   Eexporttotal = 0
   Eimport = 0
   Eexport = 0
```

```
    DoAtotal = 0
    DoAcnt = 0

    # Read data from source file
    sfp = sourceFilePath[cntLoop]
    with open(sfp, 'r') as sf:

        for line in sf:
            # stop when reach desired end line
            if sEndL != -1:
                if currentLine >= sEndL:
                    exit()

            # put each line values into a list into fields
            fields = line.split(",")

            if currentLine >= sBeginL:
                # Import data
                Epv = fields[pvField]
                Epv = float(Epv)
                Epvtotal += Epv

                Ewt = fields[wtField]
                Ewt = float(Ewt)
                Ewttotal += Ewt

                Ere = Epv + Ewt

                Esm = fields[smField]
                Esm = float(Esm)

                EshortB = fields[shortBfield]
                EshortB = float(EshortB)
                if seasonalBfield == None:
                    EseasonalB = 0.0
                else:
                    EseasonalB = fields[seasonalBfield]
                    EseasonalB = float(EseasonalB)

                    # The loss of seasonalB
                    if discrete == True:
                        if EseasonalB > 0:
                            # charging
                            loss += EseasonalB * (1-efficiency[-1])
                        else:
                            loss += -1 * EseasonalB / efficiency[47]*(1-
efficiency[47])

                Ebuffer = EshortB + EseasonalB

                # Calculating data
                # Eimport and Eexport
                Eimport = 0
                Eexport = 0
                if Esm > 0:
                    Eimport = Esm
                    Eimporttotal += Esm
                else:
                    Eexport = Esm
                    Eexporttotal += Esm

                # Eload
```

```
            Eload = Esm - Epv - Ewt - Ebuffer
            Eloadtotal += Eload

            # The relationship among load, buffer and re
            Eretobuffer = 0
            EshortBtoload = 0

            Eretoload = min(Eload, -Ere)
            Eretoloadtotal += Eretoload

            # EshortB: positive-charging, negative-discharing
            if Ebuffer > 0:
               Eretobuffer = min(Ebuffer,(-Ere - Eretoload))
               Egridtobuffer = max(0,Ebuffer - Eretobuffer)
               Ebuffertogrid = 0
               EshortBtoload = 0
               Egridtoload = Eimport - Egridtobuffer
            else:
               EshortBtoload = min(-Ebuffer,(Eload-Eretoload))
               Ebuffertogrid = max(0, (-Ebuffer - EshortBtoload))
               Egridtobuffer = 0
               Eretobuffer = 0
               Egridtoload = Eimport

            check = Eload - (Eretoload+EshortBtoload+Egridtoload)
            Egridtoloadtotal += Egridtoload
            Eretobuffertotal += Eretobuffer
            EshortBtoloadtotal += EshortBtoload
            Egridtobuffertotal += Egridtobuffer
            Ebuffertogridtotal += Ebuffertogrid

            # Update cnt number each for-loop
            cnt += 1
            currentLine += 1

         else:
            currentLine += 1

      print("Scheme:" + str(schemes[cntLoop]))

   #DoA
   DoA2 = (Eloadtotal-Eimporttotal) / Eloadtotal
   print('DoA2:' + str(DoA2) + '\n')

   # Results for table
   print(str(round(Eimporttotal / 4 / 1e6,2)))      #unit kWh, one decimel
   print(str(round(-Eexporttotal/ 4 / 1e6,2)))
   print(str(round(DoA2 * 100,1)))

   print(str(round((Eimporttotal-Egridtobuffertotal)/Eloadtotal*100,1)))
   print(str(round(Eretoloadtotal/Eloadtotal*100,1)))
   print(str(round(EshortBtoloadtotal/Eloadtotal*100,1)))

   print(str(round(Egridtobuffertotal / 4 / 1e6,2)))
   print(str(round(Ebuffertogridtotal / 4 / 1e6,2)))

   if discrete == True:
      print(str(round(loss / 4 / 1e6, 2)))

   cntLoop += 1

exit()
```

Appendix B The Modification records of ALPG and DEMKit

ALP	Folder	Working file	Operation	Values
1	workspace		1.copy example folder 2.change name	 tinyhouse
2	alpg\configs	example.py	1.rename example.py to 2.change numDays to 3.update weather_irradiation to 4.update parameters ConsumptionStoveVentilation consumptionFactor 5. location.longitude = location.latitude = location.elevation= 6.add penetrationInfraredHeater	1.tinyhouseAlpgConfig.py 2.365 * 2 3.KNMI_irradiation_2016_2019_hourly 4. 0 0.5 5. 6.853525 52.241589 28 6. 100
3	alpg	neighbourhood.py	1. line 59: change < to 2. line 74: add InfraredHeater option	> 2. while I < ...
4	alpg	profilegenerator.py	1. line 97: add InfraredHeater opt	if ... +penetration ....
5	alpg	heatdemand.py	... change ventilation parameters 1. line 207-210:	MaxAirflow=126 IdleAirflow=15 CookingAirFlow=75
6	alpg	households.py	... create parkinglot and campusEV 1.copy and rename to 2.change parameters & algorithms	parkinglot.py
7	alpg	devices.py	continue...add campusEV ... 1. line 565-599: add	1. class DeviceCampusElectricalVehicle
8	alpg/configs	tinyhouseAlpgConfig.py	continue ... configure campusEV ... 1.line191-198: add	parkinglotList = [] import parkinglot parkinglotList.append ...
9	alpg	neighbourhood.py	continue ... configure campusEV some parameters ... 1.line113-117: add	1. ...BufferCapacity = ... Consumption = ...
10	alpg	writer.py	continue... add campusEV in writer ... 1.line 89-93: create empty files beforehead 2.line207-210: write parkinglot 3.line213-230: define write campusEV	1.createFile('CampusEV_Starttimes.txt') ... 2.def writeParkinglots ... 3.def writeDeviceCampusEV ...
11	alpg	profilegenerator.py	continue...configure parkinglot ... 1.line 115-140: configure parkinglot	

DEN	Folder	Working file	Operation	Values
1	grafana	datasource	add new datasource tinyhouse,	tinyhouse

2	workspace\t	settings_demostreet.py	1.change file name 2.line35 change database name 3. to change intervals 4. numOfHouses 5. change alpgFolder 6. to change useCtrl = 7. to change clearDB 8. change startTime = 9. change timeOffset = 10. line53 add parkinglot option	1.settings_tinyhouse.py 2.'tinyhouse' 3. 24*365 4, 3 5.'alpg/output/tinyhouseo/' 8. 2017.1.1 9. 2016.1.1 10. numOfParkinglot = 1
3	workspace\t	global.py	1. change weather.weatherFile= 2. change sun.irradianceFile =	KNMI_temperature_2016_2019_hourl y 'KNMI_irradiation_2016_2019_hourly. csv'
4	demkit\conf	usrconf.py	update influxDB from "dem" to	"tinyhouse"
5	workspace\t	demostreet.py	1.change file name to 2.line 23, 27: change "demostreet" to	tinyhouse.py "tinyhouse"
6	workspace\t	streetcontorl.py	Create street smart meter: 1.line 22-25: add street-level meters 2.line 65-70, 85-90, 116-121: link smStreet to each controller	smStreet pvStreetMeter gmStreetc hmStreet
7	workspace\t	3.baseload 4.pv.py 5.timeshifter 6.electricveh 7.thermal.py	smStreet continue: add smStreet 3.line 34,48: 4.line 53,67 (add pvStreetMeter too) 5.line 40,48,101,112 6.line 59.70 7.line 90,149	try: smStreet.addDevice(~) except:
8	workspace\t	centralPV.py	create centralPV.py file... 1.update parameters and rootctrl 2.introduce pvSystemEfficiency	2.pvSystemEfficiency=0.811
9	workspace\t	tinyhouse.py	centralPV continue... 3. add it to tinyhouse.py/line 50 4. comment out/line 39	3.composer.add("street/centralPV.py") 4. ...("house/pv.py")
10	workspace\t	centralBattery.py	1.create centralBattery.py file 2.coding (inherit from battery & CHP)	
11	workspace\t	tinyhouse.py	centralBattery continue... 3. add it to tinyhouse.py/line 44 4. comment out/line 42 "house/battery.py" line	3.composer.add("street/centralBatter y.py") 4.composer.add("house/battery.py")
12	demkit\com	zoneDev1R1C.py	Modify...solar/window heatgain 1.line294	

13	workspace\t	thermal.py	create heating system … 1.line 94-99: add InfraredHeater option 2. line 127,133,146,157 add controller … parameter update … 3.line 66 temperatureDeadband 4.line 55-57 add windows 5.line 41 introduce Setpointdrop 6. line 49, 66, 68 append	1. elif heat_specs[idx][0] == "INFRAREDHEATER": 3. [-0.2, 0.0, 0.5, 0.6] 4. addWindow(1.6, 13, 90, 0.7) … N,S,E 5. setPointDrop = -4 6. + setPointDrop
14	demkit\com	heatPumpDev.py	continue …. create infrared heater 1.copy and rename to 2.line 27 HeatPumpDev 3. change commdities producingPowers cop producingTemperatures	1. infraredHeaterDev.py 2. InfraredHeaterDev 3. ['ELECTRICITY', 'HEAT'] [0,5120] {'ELECTRICITY': 1.0} [0.0, 35.0]
15	demkit\com	thermostat.py	continue… modify heatdemand for infrared heating 1.line 127,130,133 bugs 2.line 136 bugs continue … introduce newMode … 3.line 42-43, introduce parameter 4.line141-152, change algorithm	1. change self.devData['maxHeat'] to self.dev.maxHeat 2. change temperatureDeadband[1] to … [0] 3. deadBandMode & flagOperating
16	workspace\t	HeatingSettings.txt	continue … change ALPG settings … 1. change line 1: 0:CONVENTIONAL	1. 0:INFRAREDHEATER
17	demkit\com	heatPumpDev.py	continue …. create E-boiler … 1.copy and rename to 2.line 27 GasBoilerDev 3. change producingPowers cop producingTemperatures	1. eBoilerDev.py 2. EBoilerDev 3. [0, 5000] {'ELECTRICITY': 1.0} [0.0, 60.0]
18	workspace\t	components.py	continue … import new added devices when composing … 1. add line 45:	1. from dev.thermal.InfraredHeaterDev import InfraredHeaterDev 2. from dev.thermal.eBoilerDev import EBoilerDev
19	workspace\t	connectedhouse.py	continue … import new added devices when composing … 1. add line 45:	1. hm = MeterDev("HeatMeter-House-"+str(houseNum), sim, commodities=["HEAT"])

20	demkit\com	thermalBufConvDev.py	continue … allow SoC … 1. line 46: introduce chargingFlag 2. line 136-143:revalue heatProduction 3. Trigger flagBufCharging: 3.1 stop charge-buf overflow: 3.1.1 line 162: 3.1.2 line 163: revalue heatProduction 3.2 start charge-buf underflow" 3.2.1 line 187: 3.2.2 line 188: revalue heatProduction	1. self.flagBufCharging=False 2. if … elif … else … 3.1.1 flagBufCharging = False 3.1.2 self.heatProduction = 0 3.2.1 flagBufCharging = True 3.2.2 = producingPowers[-1]
21	demkit\com	heatSourceDev.py	continue … change heatDemand algorithm only for no control: 1. line 138: for zone, comment out	heatDemand = min(heatDemand, producingPowers[-1])
22	demkit\com	thermalBufConvDev.py	continue … introduce deficit supply … 1. line 47: add 2. line 124: value it 0, each timetick 3. line 191: revalue in buffer underrun 4. line 197-201: reset heat supply for zones	1. self.deficitSupply = 0 2. self.deficitSupply = 0 3. self.deficitSupply = … 4. if self.zoneDemand > 0.1 …
23	demkit\com	thermalBufConvDev.py	continue … update heatSupply for newZoneTemperature after heatProduction determined… 1.line 189-190: change label 2.line 161-165: update zone heatSupply	1. xxx.supply.xxx to xxx.demand.xxx 2. …zSet(zone, 'heatSupply', dict(heatSupplyZone))
24	demkit\com	heatSourceDev.py	continue… above … 1. line 162: comment out	1.self.zSet(zone, 'heatSupply', ….
25	demkit\com	solarPanelDev.py	create wind turbine(wt) … 1. copy and rename it to …	1. windTurbineDev.py
26	demkit\com	sunEnv.py	continue… create wind env … 1. copy and rename it to 2. line 28: import numpy package…	1. windEnv 2.import numpy as np
27	workspace\t	pv.py	continue… create wt … 1. copy and rename it to … 2. move it to folder	1. windTurbine.py … 2. street
28	workspace\tinyhouse\data\weather		continue… add data source … 1. add wind weather data files	1.potwind_twenthe_2017_2019_hourly.csv
29	workspace\t	global.py	continue…specify wind env device 1. line 28-30: add	1. windEnv & windSpeedFile
30	workspace\t	components.py	continue…import windEnv & wt 1.line 34:add 2.line 54: add	1. … Import WindTurbineDev 2. … import WindEnv

31	workspace\t	connectedhouse.py	create parkinglot(pl) … 1. copy and rename it to 2. move to new folder 3.define function	1.parkinglot 2. ...\tinyhouse\parkinglot 3. addParkingLot
32	workspace\t	centralPV.py	continue…create parkinglotPV… 1. copy and rename it to 2. .update parameters and ctrl 3. move it to folder	1. parkinglotPV.py 3. ...\tinyhouse\parkinglot
33	workspace\t	windTurbine.py	continue…create parkinglotwindturbine… 1. copy and rename it to 2. update parameters and ctrl 3. move it to folder	1.parkinglotwindturbine.py 3. ...\tinyhouse\parkinglot
34	workspace\t	street.py	continue … include parkinglot option … 1.line 26: add parkinglot option	1. ...addParkingLot…
35	workspace\t	tinyhouse.py	continue…add when compose… 1.line 46 add components for pl	1. ...parkinglot.py... ...parkinglotPV.py… ...parkinglotwindturbine... …campusEV.py…
36	workspace\t	electricvehicle.py	continue…create campusEV… 1.copy and rename it to 2.update parameters and ctrl 3.move it to folder	1. campusEV.py 3. ...\tinyhouse\parkinglot
37	workspace\t	alpgdata.py	continue… add campuseEV option… 1.line 39-43: speficify ALPG input	1. campusEV_starttimes = …
38	demkit\com	bufDev.py	Create seasonal buffer with NC … 1.copy and rename it to 2.line27: change class name to 3.line30: change self.devtype to 4.line67: add 5.line68: add 6.line214-216: add function	1. seasonalBufDev.py 2.SeasonalBufDev 3.SeasonalBuffer 4.self.battery = [] 5.self.hybridStroage = False 6.def addBattery(self, battery)
39	workspace\t	centralBattery.py	continue… create seasonalStorage … 1.copy and rename it to	1.seasonalStorage
40	workspace\t	components.py	continue…import class … 1.line 31 add 2.line 71 add	1. … import SeasonalBufDev 2. … import SeasonalBufCtrl
41	workspace\t	tinyhouse.py	continue… composing … 1.line 56 add	1. … add … seasonalStorage.py
42	demkit\com	bufCtrl.py	Create seasonal buffer PS algorithm … 1.copy and rename it to 2.line 73: overwrite startup function 3.line 121,130: define helper functions: 4.line 171: modify doInitialPlanning function to include trigger seasonalPlan 5.line 191: def doPlanningSeasonal	1. seasonalBufCtrl.py 3. createSignalS and aggregateProfiles

43	C:\Users\D\(	1.loadCtrl.py 2.tsCtrl.py	continue... create doSeasonalPrediction for controllers... 1.line 203-209 2.line 271-334: both ts and bts Ctrl	
44	C:\Users\D\(	1.loadDev.py 2.solarPanelDev.py 3.windTurbineDev.py	continue... modify readValue function.. 1. comment out readValues function in file 1,2,3 2. create new readValues function for file 1, line 146	
45	demkit\com	entity.py	continue ... create resample functions for profiles ... 1.line 127, def function	1.resampleProfile
46	demkit\com	lzoneDev1R1C.py	continue ... modify thermal doPrediction function... 1.line 324:	
47	demkit\com	lthermostat.py	continue... modify (functions) to include timeBase... 1.line 240: modify doUpperPrediction 2.line 271: modify doLowerPrediction 3.line 233,234: add timeBase in code	
48	demkit\com	1.environment\weather Env.py 2.dev/thermal/zoneDev 1R1C.py	continue...modify do...Prediction function (use timeBase) in zoneDev dpPrediction... 1.file1 line92: TemperaturePrediction 2.file2 line231: doGainPrediction 2.file2 line255: doVentilationPrediction	
49	demkit\com	simHost.py	continue ... reverse device list, simulate buf in the end ... 1.line 37: add	1.self.devices.reverse()
50	demkit\com	1.opt\optAlg.py 2.ctrl\seasonalBufCtrl.py 3.dev\seasonalBufDev.p y	Add inefficiency to discrete mode for buffer by Gerwin... 1.2.3. modify algorithm	
51	demkit\com	seasonalBufDev.py	continue ... introduce a new efficiency paramter different from Gerwin's discrete vector 1. line 52: add (efficiency for charging, storage and discharging) for next step	1. [1.0, 1.0, 1.0]
52	demkit\com	seasonalBufCtrl.py	continue... integrate inefficiency speficially for seasonal planning... 1. line 450-469: loss when calculating seasonal target SoC	

53	workspace\t	seasonalStorage.py	continue…specify paramters... 1. line 47: add 2. line 51-52: specify ratio to scale power into discete vector 3. line 55-56: specify efficiency for discharging, storage, charging	1. .discrete = True 2. ratio = np.logspace… powers = … * np.array(ratio) 3. .efficiency=[0.45,1.0,0.65] chargingEfficiency = …