A community energy management system for smart microgrids

Verba, Nandor; Nixon, Jonathan Daniel; Gaura, Elena; Dias, Leonardo Alves; Halford, Alison

DOI:
10.1016/j.epsr.2022.107959

Document Version
Publisher's PDF, also known as Version of record

Citation for published version (Harvard):
Verba, N, Nixon, JD, Gaura, E, Dias, LA & Halford, A 2022, 'A community energy management system for smart microgrids', *Electric Power Systems Research*, vol. 209, 107959. https://doi.org/10.1016/j.epsr.2022.107959

Electric Power Systems Research 209 (2022) 107959

Contents lists available at ScienceDirect

Electric Power Systems Research

journal homepage: www.elsevier.com/locate/epsr

A community energy management system for smart microgrids

Nandor Verba [a], Jonathan Daniel Nixon [b,*], Elena Gaura [a], Leonardo Alves Dias [c], Alison Halford [a]

[a] Centre for Computational Science and Mathematical Modelling, Coventry University, Coventry, United Kingdom
[b] Centre for Fluid and Complex Systems, Coventry University, Coventry, United Kingdom
[c] School of Computer Science, University of Birmingham, Birmingham, B15 2TT, United Kingdom

ARTICLE INFO

Keywords:
Smart microgrid
Community energy management system
Cyber-physical systems
Edge computing
Internet of things
Displaced communities

ABSTRACT

Community micro-grid energy projects are needed to drive de-carbonisation and increase equity of energy systems among displaced communities. However, micro-grid solutions are often inflexible and lack functionality to respond to displaced community energy needs and ensure the long-term sustainability of interventions. This paper explores the use of fog-computing retrofit architectures deployed on community micro-grid infrastructures to enable flexible demand management to improve service delivery and longevity. A micro-services solution is proposed that decouples components increasing resilience and testability while allowing hybrid edge-cloud deployments. The architecture is outlined and demonstrated for a micro-grid providing energy to two nurseries and a playground in Kigeme refugee camp, Rwanda. To enact the community priorities within the demand management system, modified Genetic Algorithm (GA) methods are outlined and tested for different use-case scenarios. The performance of the modified GA methods are then compared with a pre-existing battery protect controller and an alternative deterministic (space-shared) energy manager model. A modified search space GA method was required for GA to outperform both the existing battery controller and proposed deterministic method in terms of achieving the highest utility function in almost every use-case. The results further showed how simple community priorities can be set and used to enact control on the system in 24h timeframes that are in line with the local decision-making context.

1. Introduction

An estimated 89% of camp-based refugees have no access or limited access to electricity [1]. To improve energy access in refugee settlements, there is a growing emphasis on using photovoltaic-based micro-grids for humanitarian energy interventions [2]. There are also opportunities for community energy projects to be instrumental in driving de-carbonisation and increasing the equity of energy systems [3]. However, due to humanitarian short term funding cycles, resources and finances are typically limited in refugee camps and user needs in these infrastructure-less areas are subject to rapid change and are largely unknown (for example due to an influx of refugees, emergency needs and relocation schemes [4]). Conversely, refugees are residing in camps for protracted times, for example in Rwanda, the length of stay is 16 years or more [5]. Therefore, energy systems need to be flexible to allow for changes in control and facilitate how refugee communities can become active participants in managing a shared energy resource to ensure its sustainability [6].

Conventional load-shedding and economic dispatch models based controllers may not be suitable for resource-constrained decentralised community micro-grids. In a context where refugees often have to share energy resources, whilst -difficult to achieve in displaced settings, simple request, negotiation and priority-based approaches could improve micro-grid. The use of intelligent software and/or hardware interfaces would be needed for such user interactions, but this is an under-explored area of research for displaced communities. In contrast, adaptive load shedding to control frequencies and voltages in power networks is a widely explored area of research, which is now informing island micro-grid control [7]. Scheduling and load shedding in off-grid micro-grids has typically focused on increasing stability, renewable penetration and reducing energy costs [6]; however, there are other considerations for micro-grid control, particularly for displaced community systems, such as user dissatisfaction and longevity of batteries, which are often the cause of system failures in standalone solar energy systems [8]. Despite

* Corresponding author.
E-mail address: jonathan.nixon@coventry.ac.uk (J.D. Nixon).

https://doi.org/10.1016/j.epsr.2022.107959
Received 7 October 2021; Received in revised form 20 January 2022; Accepted 23 March 2022
Available online 9 April 2022

this, there is evidence [9] battery health is often not included for micro-grid system management control optimisation research.

Multi-objective energy dispatch models for micro-grids attempt to satisfy multiple competing utility functions. Bordin et al. [10] proposes a linear programming model to reduce energy costs and battery degradation and to investigate the sensitivity of battery degradation to make batteries a more financially viable solution compared to the use of diesel generators in Rwanda. Ju et al. [11] incorporate the energy cost associated with battery degradation into the scheduling of utilities in a micro-grid. Neither of these studies considers user requirements and satisfaction as a result of load shedding to decrease costs and increase component lifetimes. Mohammad et al. [12] explore managing user dissatisfaction and maintaining system stability by determining which loads to shed in an isolated multi micro-grid infrastructure so that shedding imposes the least cost to the system. They employ Genetic Algorithms (GA), Exponential Moving Averages (EMA) and General Algebraic Modeling Systems (GAMS) to solve the optimisation problem and compare the results with a conventional under-frequency load shedding (UFLS) scheme, which brings about non-optimal load shedding. EMA was shown to outperform GA both in terms of solution and convergence speed. Other authors [13] have also found that alternative modified algorithms can outperform conventional GA and PS techniques for multi-objective dispatch optimisation.

Due to the need for complex calculations to identify the various states and parameters of a single run, GA can be considered an expensive black-box model, with modifications usually required to make it fit for purpose [14], as computational time reduces the number of reasonable steps a method can take. Utilising a search-space reduction method to overcome this limitation, as proposed in [15], is one of the most commonly applied solutions for computationally complex and large search-space problems. Deterministic energy dispatch methods such as the one suggested in [16] also offer solutions for multi-objective, non-convex and non-differentiable optimization problems and offer simple and understandable solutions that produce the same results given the same set of initial parameters. These deterministic methods are commonly characterised by lower processing times and computational requirements when compared to metaheuristic methods but most often fail to find highly optimal solutions in large data-sets [17]. Whilst multi-objective and deterministic energy dispatch methods are a promising option for micro-grid control, they have not been developed and evaluated in the context of managing displaced community energy needs to improve system viability and longevity. Furthermore, the use of GA and Particle Swarm Optimisation (PSO) like algorithms for optimising multi-objective functions is common, but given the search-space of some problems that arise they, or the problems, need to be adapted to allow the methods to identify potential solutions that are better than deterministic solutions.

This paper explores the concepts of æsmartnessg and æflexibilityg in the context of off-grid, community renewable micro-grids for displaced contexts. We ask whether or not advances in cyber-physical systems can enable overlay of architectures for flexible demand management, enhanced sustainability and longevity and improved service delivery. Specifically, we set out to answer the following research questions:

- To what extent can cyber-physical design elements and control methods be overlaid onto existing smart microgrid structures to arrive at controllable community-led energy systems?
- How do deterministic, and metaheuristic methods compare in achieving conflicting multi-objective priorities such as battery health, efficiency, and user satisfaction?
- How can existing metaheuristic methods be improved for better convergence in environments with large search spaces.

To answer this papers research questions, we developed and evaluated an easy to retrofit energy management system overlay that allows for various demand control plug-ins and user-driven demand optimisation. The overlay is demonstrated for a solar-battery community micro-grid powering a playground and nurseries in a Rwandan refugee camp. The contribution of this paper is threefold: i) a proposed cyber-physical architecture designed to solve the issues related to rigid, single-purpose solutions, ii) novelties around search-space reduction for modified GA application to multi-objective energy dispatch optimisation problems that consider user needs and priorities, and iii) comparison of several deterministic and meta-heuristic energy dispatch methods. The practical innovation is offered as open-source software, vanilla modules for the various smart components.

The following methodology section outlines the research methods and models adopted; it follows a top-down approach where the architecture is first presented, after which each component is detailed. The next section details the case study for which the models are demonstrated and evaluated, detailing the specific site, local context and micro-grid system installation. The results and discussion section builds on the knowledge and requirements gained from co-design workshops to showcase how these affected the adapted architecture while also showcasing how the various components and overlays perform on historic and scaled energy usage scenarios.

2. Models

The alternative objectives considered in this research are maximising battery health and overall system efficiency (ratio of consumed energy to potential energy generation) and minimising user dissatisfaction in terms of access to prioritised energy supply for lights and sockets.

2.1. Battery model

The battery model proposed by [18] is simplified in Eq. 1 to only focus on estimating the available energy in the battery. For this model, S_t (St) represents the Stored Energy (energy available) at time t, $C_{t,t-1}$ (Ct) is the Charged Energy between time $t-1$ and t, $D_{t,t-1}$ (Dt)is the Discharged Energy between time $t-1$ and t, whilst C_e (Ce) is the Charged Efficiency.

It is essential to mention that the model does not consider the battery deterioration, battery fade, or any other factors proposed by [9] that can result in more accurate battery models. We use the battery model for short-term forecasting; thus, long-term degradation and capacity losses will have a minimal effect on our use case. The State of Charge ((*SoC*) (Soc)) is usually defined by the available energy stored in the battery divided by the storing energy capacity of the battery. Also, for real systems, the Soc is often provided by the underlying battery management systems.

$$\Delta S_t = S_{t-1} + C_e \Delta C_{t,t-1} - \Delta D_{t,t-1} \tag{1}$$

2.2. Energy demand forecasting

The approach for energy demand forecasting is based on the results of [19], where Long Short-term Memory (LSTM) forecasting has proven to be more accurate than Autoregressive Integrated Moving Average (ARIMA) and Seasonal Autoregressive Integrated Moving Average (SARIMA). Still, LSTM requires more resources to run, which might be an issue for the resource constrained Raspberry Pi systems and causes excess power consumption. Running multiple models has even higher requirements, especially when it comes to re-training the data. The SARIMA model was used for this system as it was a close second on accuracy to LSTM in the tests done by [19] and it is a more lightweight model, making it better suited for this application. This case would be similar to all microgrids where computational resource is scarce. With the lack of HVAC style, consumers achieve the desired accuracy to sustain the control methods.

A short-term hourly forecast of 24h is used to identify the energy demand in the upcoming period, which is in line with the seasonality of

the consumption and photo-voltaic data. A SARIMA model is used as it can identify seasonal and daily trends. The ARIMA model was introduced by [20] and is commonly denoted as $ARIMA(p,d,q)$ where p is the number of auto-regressive terms, d is the number of differences needed to make the data stationary, and q is the lagged forecast error. The SARIMA model expands on the ARIMA model by adding a seasonal component and is commonly denoted by $SARIMA(p,d,q)(P,D,Q)m$ where p, d and q have the same role as in the ARIMA model while m represents the number of data points in a season, and P,D and Q represent the seasonal equivalents of the p,d and q parameters.

The forecast model training step is performed using the data regarding the cleaned energy consumption paired with its hourly seasonality variable. For this purpose, the data is collected from the previous 14 days. The model can be used to generate a longer forecast with diminishing accuracy. If not enough data is available, a naive forecast representing a typical day is used instead.

2.3. PV generation forecasting

The energy generated by the photovoltaic (PV) panels is forecasted using a scaled ClearSky model. The authors in [21] show that there are significant errors when predicting cloudy days, even when using SARIMA and Clearsky model-based methods. The PV power output from panels is currently limited when the battery store is full; hence, the available power reported is significantly less than the actual amount. Based on the limiting behaviour of the charge controller and reduced processing capabilities of the system, a simple adapted ClearSky model is chosen for PV forecasting as it can offer the required accuracy without the need of implementing advanced models that will need to capture weather data and also be able to adapt to the limiting behaviour.

The SOLIS Clear Sky model from Ineichen [22] is used to gain the Global Horizontal Irradiance (GHI), Direct Normal Irradiance (DNI), and Diffuse Horizontal Irradiance (DHI) values for a particular area. The In-Plane Solar Irradiance I_T (IT) is calculated using GHI, DNI, DHI, solar zenith, solar azimuth, panel surface tilt and panel surface azimuth using the equation 2.16.7 from [23]. The potential available power is calculated using the equation 2, where E_P (EP) is the Potential Generated Power, A_{PV} (APV) is the area of the PV panels, R_{PV} (RPV) is the panel efficiency at standard test conditions (std) and k_t (kt) is the clearness index, which is the ratio of measured irradiance at ground level to estimated irradiance in clear sky conditions at ground level as proposed in [24].

$$E_A^{For} = A_{PV} R_{PV} I_T k_t \tag{2}$$

2.4. Method evaluation metrics

The SoC^{Mean} (SoCM)in Eq. 3 is a simple analogue to the battery health parameter that allows us to see how the mean battery Soc changed throughout the testing period. It is simply the mean value across all hours in the resulting system.

$$SoC^{Mean} = \sum_{t=0}^{N} SoC(t) \frac{1}{N} \tag{3}$$

The η_{System}^{Mean} (nSM) in Eq. 4 is the mean system efficiency across the testing period. It is represented by the total amount of energy generated by the PV arrays, E_A, divided by the total amount of potential energy that could have been generated, E_P. In the case of the scaled scenarios, the potential energy generated E_P is calculated as the clearsky model scaled to the system as proposed in Eq. 2. The energy generated by the array E_A is defined to follow the E_P value, except in cases where the battery is full, in which case its maximum value is equal to the power drawn from the system.

$$\eta_{System}^{Mean} = \frac{\sum_{t=0}^{N} E_A}{\sum_{t=0}^{N} E_P} \tag{4}$$

The C_S (CS)in Eq. 5 represents the capacity shortage in the system, or the amount of total device load that the system could satisfy. It is defined based on the actual energy consumed divided by the total amount of energy use estimated. The consumed energy is based on the energy consumption of the i-th device $E_D^{i,t}$ multiplied by it's available quota $Quota_D^{i,t}$ (QiD). Meanwhile, E_D^t (EtD) is the total estimated energy consumption without imposed quotas. Therefore, the higher the quota, the lower is the capacity shortage, as a high quote provides the desired energy consumption.

$$C_S = 1 - \sum_{t=0}^{N} \frac{\sum_{i=1}^{A} E_D^{i,t} Quota_D^{i,t}}{E_D^t} \frac{1}{N} \tag{5}$$

The $Quota_a^{Mean}$ (QM) in Eq. 6 represents the relative amount of Device Load requirements satisfied by the system adjusted for the individual priorities. For the 7 devices, that represent the existing use-case, the maximum value for QM is 0.37 in cases where all the device energy demands are met.

$$Quota_a^{Mean} = \sum_{t=0}^{N} \frac{\sum_{i=1}^{A} \left(Quota_D^{i,t}\right) \frac{1}{Priority_i}}{A} \frac{1}{N} \tag{6}$$

2.5. Battery health utility function

The battery health and longevity of a battery bank can be increased by minimising the Depth of Discharge (DoD) and reducing the amount of draw throughout the runtime of the system, and this is a common variable to consider when using empirical methods [25], where charge cycles are not available or are not practical for the time-frame. The simplified battery health factor can be seen in Eq. 7 where the Battery Factor $F_B(t)$ (FB) for time t is normalised to the range [0,1] using a maximum Soc, SoC_{max}, and minimum Soc, SoC_{min}, value.

$$F_B(t) = \frac{SoC(t) - SoC_{min}}{SoC_{max} - SoC_{min}} \tag{7}$$

2.6. System efficiency utility function

The system efficiency factor, $F_E(t)$, at time t, is calculated as in Eq. 8. The system efficiency considers the efficiency of the energy use η_{system} (nS) based on the total load and the useful load as measured at the socket and light meters. The net energy from the PV array, E_A, considers the measured energy captured from the PV panels as read from the systems, while E_P considers the potential energy available from the PV panels calculated using Eq. 2, with a performance ratio PR_{PV} (PRPV) of 1.0. The two parameters are already normalised, allowing $F_E(t)$ to be in the range [0,1].

$$F_E(t) = \eta_{System}(t) = \frac{E_A(t)}{E_P(t)} \tag{8}$$

2.7. Dissatisfaction factor utility function

A dissatisfaction degree is introduced to formalise the experience of the users. When a control algorithm decides to curtail a consumer, this affects the system's users. The importance of each consumer is taken into account through their priorities. The level of curtailment for each consumer is set by percentage Quota $Quota_D^{i,t}$, where 1.0 means that all the energy requirements are met and 0 means that there is no energy available for the device.

This dissatisfaction degree $F_D(t)$ (FD) is shown in Eq. 9, where $Quota_D^{i,t}$ is the amount of energy available to be used for device group i, and $Priority_i$ is the priority of that appliance. This is divided by the total

number of appliances to get the values in the region [0,1].

$$F_D(t) = \frac{\sum_{i=1}^{A} Quota_D^{i,t} \frac{1}{Priority_i}}{A} \tag{9}$$

2.8. Utility function variations

We introduce three utility functions to balance and combine the previously described objectives. Utility Function 1 (U1) focuses on maximising battery health to provide insight into how a model optimises a solution for a single objective. Meanwhile, Utility Function 2 (U2) attempts to balance battery health with users access to energy, and Utility Function 3 (U3) combines user satisfaction with system efficiency.

1. **U1 - Battery Health Consideration** - is defined in Eq. 10. As it has no incentive for satisfying any user requirements, the solution will tend to turn off devices to prevent battery draws and, therefore, increase the battery health. The battery health weight is $w_B = 1.0$.

$$\max f(x) = \sum_{t \in T} w_B F_B(t) \tag{10}$$

2. **U2 - User Satisfaction combined with Battery Health Considerations** - is defined in Eq. 11. It focuses on improving user satisfaction, i.e., the number of user requirements that can be satisfied, while also improving battery health. Hence, resulting in a conservative approach to allowing the use of the system. For this purpose, we define the weights for the battery health and dissatisfaction factor as $w_B = 0.5$ and $w_D = 0.5$, respectively. Thus, normalising the two utilities in the range of [0,1].

$$\max f(x) = \sum_{t \in T} w_B F_B(t) + w_D F_D(t) \tag{11}$$

3. **U3- User Satisfaction combined with System Efficiency** - is defined in Eq. 12. It focuses on improving the user satisfaction and improving the sustainability of the system. This should result in an approach that attempts to use as much energy as possible while also ensuring that at the end of a 24h period, the amount of energy available is the same as at the start of it. The weight for the sustainability utility is $w_E = 0.5$, and the weight for the dissatisfaction factor is $w_D = 0.5$, which puts the two utilities in the same range of [0,1].

$$\max f(x) = \sum_{t \in T} w_E F_E(t) + w_D F_D(t) \tag{12}$$

3. Energy control methods

The energy management controller component is designed so that a variety of control methods can be adapted to suit the specifics of a local system. We propose a Space Shared Energy Manager and several variations of a Genetic Algorithm (GA) based optimiser. We compare these methods with each other and the existing basic battery protection controller.

3.1. Battery protection controller

The Battery Protection Controller (PC) is a standard feature on charge controllers. When the SoC (based on battery voltage) reaches a set point value, every device on the system is turned off to protect the battery. Once SoC values go above a set point, the devices are turned back on.

3.2. Space shared energy manager

The Space Shared Energy Manager (SSEM) Algorithm is based on Space Shared Resource allocation of the CPU and VM resources [26]. At every iteration, the algorithm uses the forecasted and measured data for its decision-making process, as can be seen in Algorithm 1.

As can be observed, the algorithm applies quotas on devices based on their priority level if the system's total energy consumption is higher than the available energy. The overall concept for the method is to first split the evaluated period into sets, each set defined by the change in the balance between energy demand and supply. Therefore, the Algorithm. 1 defines the quota for each device at every h-th hour, given a set of tolerances; thus, controlling the amount of energy the device can use. The tolerance limits are set based on typical accuracies for SARIMAX on 24h forecast windows from [19] and our tests on the underlying data suggest that a 15% bound is a typical safe region. Significant errors in forecasting can cause unwanted behaviour in devices being turned off too late or being turned off in vain.

Each device group can use all the energy resources available for its set. However, given the limited energy available, the devices within a group are active according to their priority (e.g. a device might only be able to use 60% of its desired energy until 6 a.m., but after the sun comes up, it can go up to 100% and then back down after the sun sets). Hence, each set will have a different number of active devices, and there might be cases where only the highest priority devices are active within some sets.

Regarding the algorithm inputs, E_A represents the generated energy in the system, that is, the total energy that will be available in the h-th hour, whilst E_A^h is the amount of energy forecasted to be generated in hour h ahead of the current time. The total available energy is calculated based on Eq. 13, where E_{Avail}^h (EtA) is the total energy that will be available in the next h hours (based on the energy available in the battery and the generation forecast), SoC_{Bat} is the latest state of charge of the battery in the set and SoC_{Bat}^{Min} is the lowest allowable state of charge of the battery, and E_{Bat}^{Cap} is the total Energy Capacity of the Battery. The total energy consumed in hour h, E_D^h, is shown in Eq. 14, where k is the total number of connected devices and the system loads, and $E_D^{i,h}$ is the predicted energy consumption of device i in time h.

$$E_{Avail}^h = E_A^h + \left(SoC_{Bat} - SoC_{Bat}^{Min}\right) E_{Bat}^{Cap} \tag{13}$$

$$E_D = \sum_{i=0}^{k} E_D^i = \sum_{i=0}^{k} \sum_{h=0}^{d} E_D^{i,h} \tag{14}$$

The $split\_to\_sets(E_A^h, E_D^{ij})$ function splits the existing time-domain into slots by identifying when the generation and consumption balance switches signs so that regions where the PV panels are charging the batteries are not constrained by slots where they are being depleted and the consumption needs to be constrained.

The $apply\_scenario(SoC_{Bat}, E_A^h, E_D^h)$ function takes the new consumption values after the new quotas were applied to the device in question for the time set and calculates the batteries SoC and resulting Energy Availability for the time period in question.

Example: Suppose the total energy available, E_{Avail}^{set}, for a period is 7kW; however, the system and devices will use a total of 7.4kW, in which the lowest priority group of devices uses 1.4kW. Therefore, the lowest priority group receives a quota that limits it to a maximum consumption of 1kW until the next period with a higher amount of energy available. Also, the lowest priority group devices can be turned off if their quota reaches the limit, allowing higher priority devices to fulfil their energy needs. This *group limit* is then distributed equally to each device in the group. For Lights, the group Wh limits need to be translated to minutes of light. This can be done by taking the average *on* value for each light and dividing it with the Wh available. These calculations consider inaccuracies in measurement and forecasting by adding 15%

Input: $E_A^h, E_D^h, E_D^{i,h}, E_{Avail}^h, SoC_{Bat}^h, Priority\_list[d1...d2]$;
Output: $Quota_D^{i,h}$;
Set: $tolerance \leftarrow 1.1$; $SoC_{Min} \leftarrow 40.0$;
foreach $device \in ordered(priority\_list)$ **do**
  $sets \leftarrow split\_to\_sets(E_A^h, E_D^{device,h})$;
  foreach $set \in sets$ **do**
    if $E_{Avail}^{set} - E_{Cons}^{set} \leq 0.0$ **then**
      $Quota_{Dev}^{i,set} \leftarrow 0.0$;
    else if $E_{Avail}^{set} - E_D^{set} \geq E_D^{i,set}$ **then**
      $Quota_{Dev}^{i,set} \leftarrow 1.0$;
    else
      $Quota_{Dev}^{i,set} \leftarrow \frac{E_{Avail}^{set} - E_D^{set}}{E_D^{i,set}}$;
    end
    $E_D^h \leftarrow E_D^h + E_D^{i,set} * Quota_D^{i,set}$;
    $SoC_{Bat}^h, E_{Avail}^h \leftarrow apply\_scenario(SoC_{Bat}, E_A^h, E_D^h)$;
  end
end

Algorithm 1. Space Shared Energy Manager

over-estimation on each device.

3.3. GA based optimisation algorithms

Genetic Algorithms (GA) are a common form of metaheuristic optimisation algorithms that can be used for any instance where the optimisation problem can be translated into a set of chromosomes and generations. These algorithms, however, need to be tailored to the application so that they have better performance and dont get stuck in local minima points. For this purpose, we modify how a gene is defined by tuning the method's hyper-parameters and the implementation of its mutation, crossing and selection methods.

In this paper, we propose several variations of GA. The simplest of these, called GA-V1, translates hourly quotas for devices to genes and showcases GA with a direct transformation of the quota system into genes. Considering the search-space implications of doing this, we propose a second variation, called GA-V2, that improves the precision of quotas and some hyper-parameter tuning together with mutation, generation and crossing methods selection. The final variation, GA-V3, uses a set of search-space reduction techniques that identify sets in the data and further reduce quotas' precision, significantly improving the convergence of the method.

4. Case study

The community energy management system proposed and demonstrated in this research relies on community input and information from an energy system controller (e.g. consumption and generation data) to make decisions on setting system constraints (e.g. setting socket/light quotas to limit how much and when energy is available to the users). The energy system controller makes these decisions based on a communitys preferences and other system constraints (see Fig. 1). We start by outlining the architecture for a general community energy management system controller and then propose alternative models to optimise the controller to balance specific conflicting objectives. The model is then demonstrated for a case study micro-grid system to evaluate the

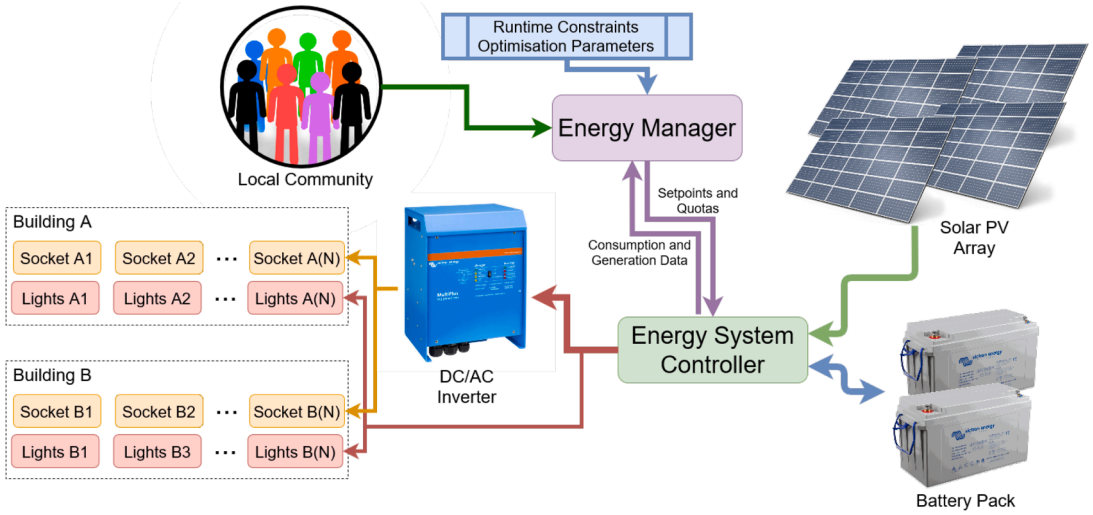


Fig. 1. Overview diagram of the cyber-physical system and its connections with the community, cyber-physical components and buildings represented.

controller and the performance of the alternative optimisation models.

The various controllers are demonstrated using real data from a micro-grid system deployed in the Kigeme Refugee camp. The system's site and set-up are outlined in this section, along with specific model input parameters for evaluating and demonstrating the proposed energy management system and optimisation algorithms. Alternative system scales are also introduced to test the controller under different potential demand and utilisation scenarios. In the scaled scenarios, the size of the PV panels is scaled as altering their size is a more realistic scenario than altering the energy consumption of the connected devices.

4.1. Site location and refugee context

Rwanda hosts nearly 164,000 refugees, the majority from the Democratic Republic of Congo and Burundi, who live in one of six refugee camps: Gihembe, Kigeme, Kiziba, Mahama, Mugombwa, and Nyabiheke [27]. In the governmental drive for better energy access provision for all, off-grid systems have the potential to deliver affordable, sustainable and safe energy for camp-based refugees. In 2021, Kigeme refugee camp in the Nyamagabe District in Southern Province, Rwanda hosts just under 18,000 refugees [28]. Although there are some communal solar energy interventions (streetlights, for example), the camp has limited connection to the main electricity grid [29].

4.2. Local system

The Microgrid deployed in Kigeme, Rwanda is based on a Smart Metering system that is outlined in Fig. 2. This system relies on a set of AC and DC smart switches, which were installed by a local supplier, that sit on top of a Victron Controller, inverter and 21.2kW battery set-up. A GSM connection with the installer's central Server ensures that quotas can be updated externally and that the load and generation profiles of the system can be monitored. A Local Telnet interface can be used to attach extensions. The connected smart switches operate on LoRaWAN and can be controlled through the local interface. The communication lines are denoted by blue connections while the power-lines are denoted by yellow connections. The daily quota system can be used to impose power and time limits on sockets and lights.

Based on the community engagement workshops (see 4.3), for the purpose of our testing the quota priorities were set as: ['Streetlights','-Playground Lights','Nursery1 Lights', 'Nursery2 Lights','Nursery1 Sockets', 'Playground Sockets'], where the the Streetlights are the most important and the Playground sockets are the least important. On a deployed system, these priorities would be changing with the current user needs (prioritising buildings for events or sockets for businesses).

The interface is designed to be deployed on a Raspberry Pi 3 Model B to ensure low power consumption while enabling the use of networking, hosting and computation. The use of SARIMAX as the forecasting method ensures that the computational load is appropriate for the device. If more computationally constrained systems are used such as Raspberry Pi Zero, the GA based optimisation can be replaced by the SSEM method. The final system was tested with simulation data to ensure the stability of the deployment and loads.

4.3. Community co-design and problem formulation

Co-designing with communities facilitates the deployment of a sustainable and contextually appropriate energy system that helps transition from needs-based energy solutions to energy interventions that build community capacity. During the planning stages of the micro-grid, a co-design workshop was held to engage potential micro-grid system users, community and camp leaders, and other vested stakeholders in Kigeme camp, Rwanda, in November 2019. The inclusion of end-users in the design process encouraged thinking around provision for the day-to-day maintenance post-installation, increased the utilisation of the systems and improved understanding of the benefits of renewable energy. These recommendations will aid not only humanitarian decision-makers when planning energy services, but are transferable into other energy-poor contexts.

Underpinning the scheduling method was the delivery of shared energy resources in an equitable and just framework that respected and responded to community needs, resources, and future aspirations.

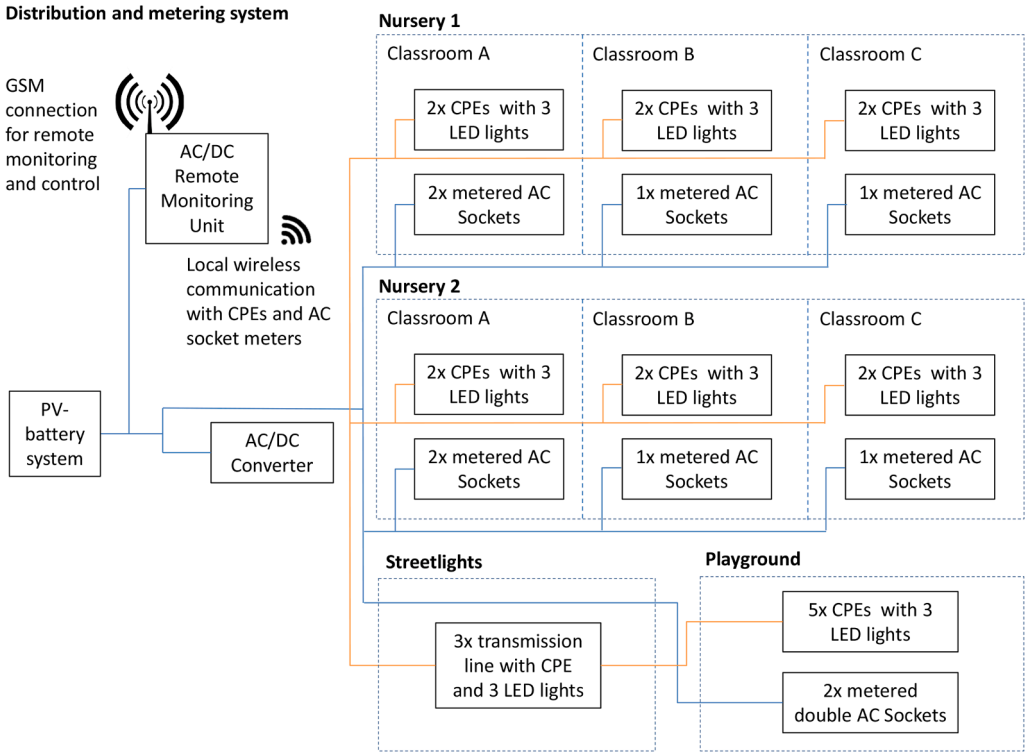


Fig. 2. Distributed metering system.

Although device scheduling, as presented in [30], is most commonly associated with Smart Devices and IoT trends and refuges do use mobile phones extensively, this was deemed unsuitable as smart devices and smartphones are not prevalent. There was also concern around the inability to control energy inclusively. Instead, the design of the interfaces, control methods and governance structure were guided by the core principles highlighted in [31].

4.4. Adapted architecture

The high-level view of the micro-grid system and its added components can be seen in Fig. 3. The controller is a two layered system similar to the approach used in [32]. The first layer ensures the safe functioning of the energy system, and is part of the original design. The second layer comprises of a smart overlay that enables usage patterns to be specified by the energy manager and translates these into energy usage quotas for the first layer controller based on a set algorithm.

The deployment follows a micro-service architecture that uses concepts from Extract Transfer Load (ETL) [33] to loosely couple digital components with the aim of maximising interchangeability among systems while keeping the architecture simple and reducing overheads. This approach means that a large system can be broken down into separate components that share information with each other through a database. These components run independently and are deployed as docker services where appropriate to increase reliability and control over them. Applying fog-computing concepts to the design of the architecture means that the components can be deployed in environments with high latencies, local processing requirements and power-consumption constraints as well as in centralised deployments that can offer the processing power that may be required by more complex systems that rely on deep-learning models for forecasting and decision making. Both these systems can use this architecture with just configuration and deployment modifications.

The microservices architecture and separation of concerns between services allows the digital components to function independently. The critical components are those that communicate with the physical components and that make decisions. The control algorithm can make decisions using outdated data and forecasting which ensures the typical delays dont cause issues. The set point deployer can enact control without needing a stable connection to the controller. Without the deployer no changes can be enacted on the system as that is the key component that communicates with the physical system. If this component fails the system can revert to the built-in Battery Protection Control method.

The front end REST API provides a data-source and sink for the community interface to send user priority preferences that it saves to the database. These priorities are in line with the user requirement on governance and the type of control they want on the system. The local deployment of forecasting and control allows the connectivity issues to be resolved, while the updates to the interface's language and icons make the system easier to use in the local context. The iterations through the versions and feedback from stakeholders and partners ensures that the design and the underlying governance structure was co-developed and meets the needs of the community.

The interface is based on the requirements of the community and can be seen in Fig. 4. It provides access to the various information and controls available in the system while providing information in multiple languages. Through an iterative process, the interface was made more understandable with clearer descriptions, information, and more understandable labelling being added. Updating the priorities on the

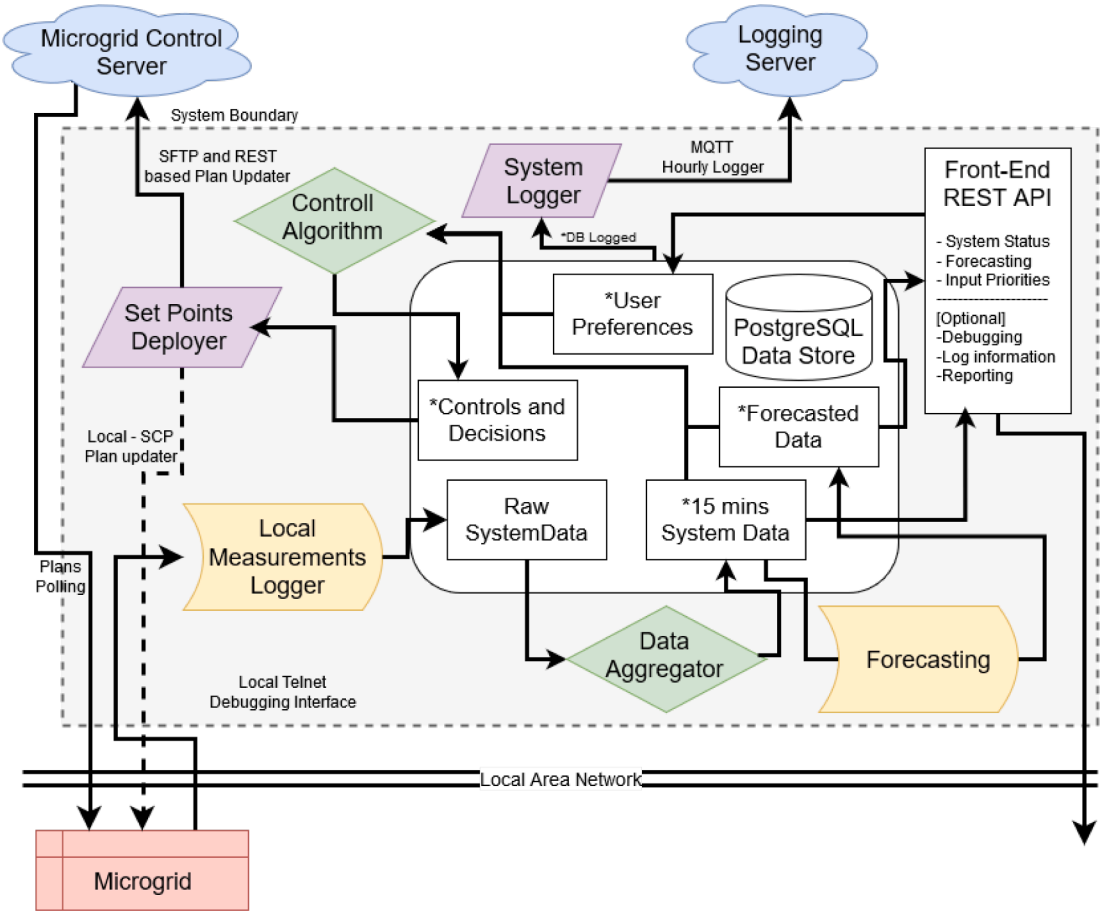


Fig. 3. High level architecture of the System.

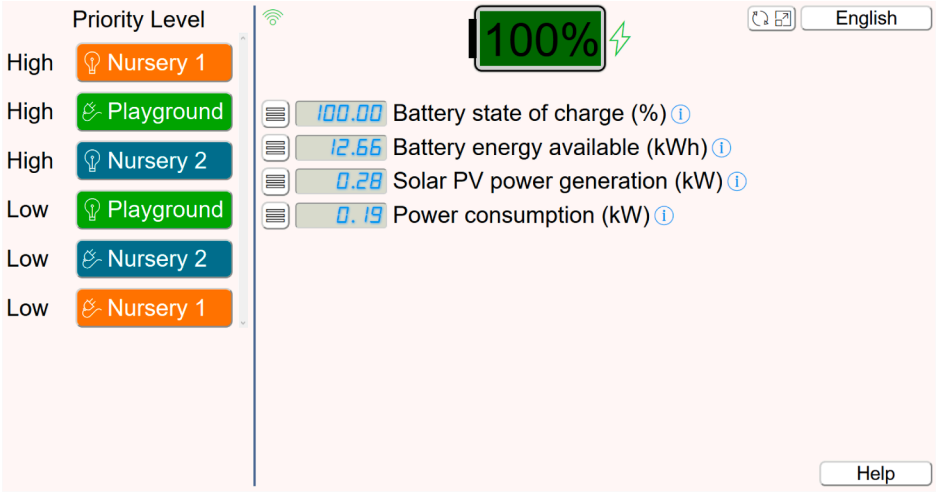


Fig. 4. Final version of the community interface, showing the communities selected priorities of light over socket power

interface triggers the controller to re-run and allows the users to dynamically vary devices by priority.

The tailored System Logger creates a snapshot of the systems every 15 minutes, saves it to the database and sends the log via MQTT to a Logging University server. The logging data contains a snapshot of the current system state at a given timestamp, together with message ID and transmission information, including sending the report to the central Coventry server, which allows failed messages to be re-sent and the system to retain logging even with extended outages. This component is crucial in enabling digital twin setups and fault detection systems.

The adapted architecture was designed to be deployed in two use-cases: a local deployment and a remote deployment. As there were difficulties gaining access to the camps due to restrictions, the local system was tested on a replicated telnet client that simulated the messages of the on-site base-station. The remote deployment was tested and evaluated by connecting to the remote data portal offered by the micro-grid providers. This deployment shows the versatility of the flexible architecture as all the functionality was retained. The downside of deploying the system remotely was the introduction of large latencies into the system.

The developed architecture can be found on GitHub [34], where the existing platform is implemented on top of the presented use-case. Various deployment options are supported with Docker and Makefile based installs available, to showcase the functionality of the interface. In situations where there is no system to connect to, an example deployment is also available.

The latencies for data upload to the portal were monitored for a 196 hour period in March, 2021 for which the results can be seen in Fig. 5. These are synchronisation latencies and will be the same for the upload of quotas to the online system.

The latencies in remote deployment show that unstable network connections in remote areas require fog computing-based approaches where the central system is deployed as close to the physical system as possible. These added latencies also reduce the type of control that can be enacted as decision-making processes need to consider that even the most well-behaved devices have a mean averaged delay of 53.7 minutes,

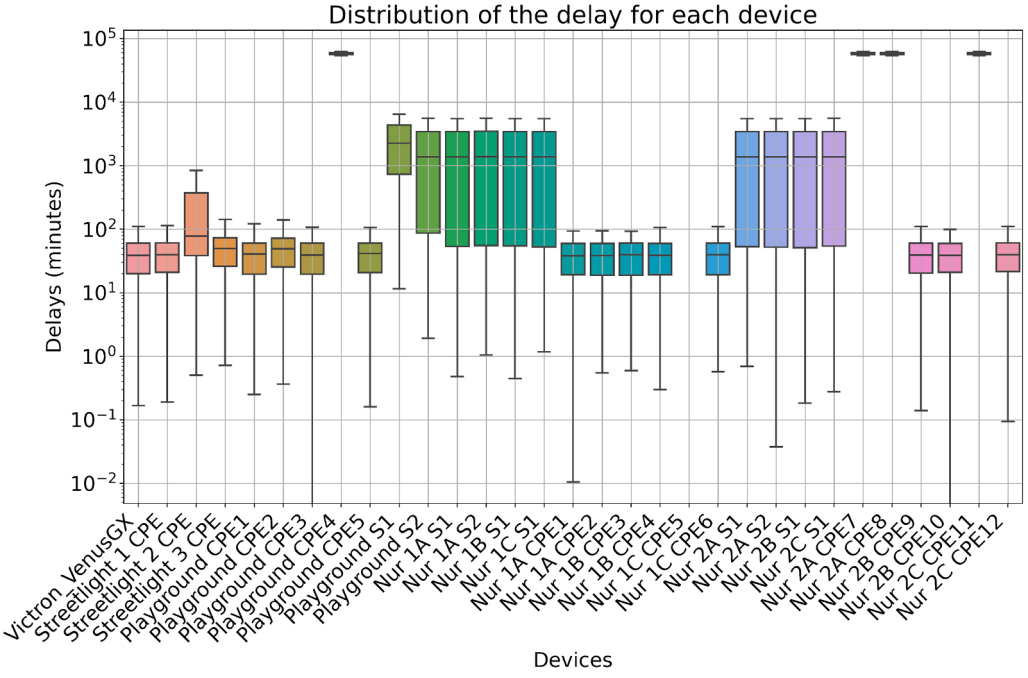


Fig. 5. System latencies in a remote deployment scenario.

while the worse ones had an average delay of 1924 minutes or 32 hours. Meantime, 3 devices failed to send any information. There is also the additional concern around how the system reacts more poorly to unexpected events (such as large consumers being plugged in), which means the forecasting of the system needs to be more accurate and local.

The measured latencies are a strong indicator for the need to deploy the control systems locally. In these cases the control method would always have direct access to the data and devices except for cases of failures. For a centralised system, or in cases where the control and optimisation methods are note deployed on site, the high latencies mean that the constraints and confidence in the forecasting methods need to be high enough to allow 1-2h delays in communication. Other control methods such as quote time-outs can also be put in place to ensure systems robustness that revert to default settings for the system. Creating quotas 24h ahead also means that the system has a certain degree of robustness to communication gaps of 24h given a typical forecasting accuracy of 8%. The issues with latencies arise with atypical consumption patterns or events that are difficult to forecast.

4.5. Data imputation and missing data

Power consumption and system state values are recorded periodically from the local or remote interface to the Micro-grid. Missing values in the live data can occur in two ways.

- MAR Missing at Random. MAR data can occur due to hardware failures and communication errors.
- MNAR Missing Not at Random, are cases where the cause of missing data can be deducted, such as State of Charge below lowest allowable Depth-of-Discharge (DoD).

In the Case of MNAR the battery protection system comes into effect and the controller is turned off together with the rest of the system, which means it can take no action or decision. Various components have different data cleaning and consistency requirements. Only the Controller and Forecaster have special requirements, every other components takes the latest available data. The data aggregator stores periodic means of the raw data and empty fields where data is not available. This is used for system logging and the API. The missing data is handled based on [35].

Model based imputation requires already available data which in the case of a recovering system or a freshly started one is not guaranteed (forecasting models can require 1-6 days previous data given a 24h seasonality). To solve this and decrease the computational requirements, two-stage imputation is used. In the first stage, the most recent hourly mean for the period is imputed. In stage 2, if the data is not available then the first available daily mean value is used. This method guarantees that data is available at any point and as the systems data quality gets better the quality of the forecasting data also improves.

4.6. Existing load and generation profiles

The overlay architecture is demonstrated using system data captured from 1st January 2020 until 12th March 2020 and before any restrictions were put in place in the camp due to COVID-19. The view of a typical days consumption and generation on the micro-grid can be seen in Fig. 6. As the system was oversized, two alternative system scale scenarios are proposed to examine how the energy management system would perform for a well-sized (medium scale) and undersized (small scale) system given the same energy demand of the community. Oversizing the system relates to the changing community dynamics in displaced community settings where people and governance around the building in the camp are subject to change, which often alters which buildings are in use and how much energy they draw.

4.7. Scaled systems

For the small-scale scenario the PV panels were reduced by 75% and the battery was reduced from a size of 21.2 kW to a size of 3.1 kW. For the medium-scale scenario the PV panels were reduced by 60% and the battery was reduced to a size of 4.2 kW. A typical day for the small and medium scaled systems can be seen in Fig. 7a and 7 b. Note that consumption levels remain the same across all system scale scenarios. In the original design, the battery SoC does not drop below 85%. For the medium scaled scenario the battery reaches an SoC level of 65%, while for a small-scale scenario the SoC reaches the 40% threshold, resulting in the PC regularly required to curtail inverter output to protect the battery.

4.8. Applied forecasting models

For the energy consumption forecasting the trained models were evaluated based on two parameters: the Akaike Information Criterion (AIC) and the Normalised Mean Square Errors (NMSE). The evaluation of the Seasonal ARIMA Model on the data-set was performed using

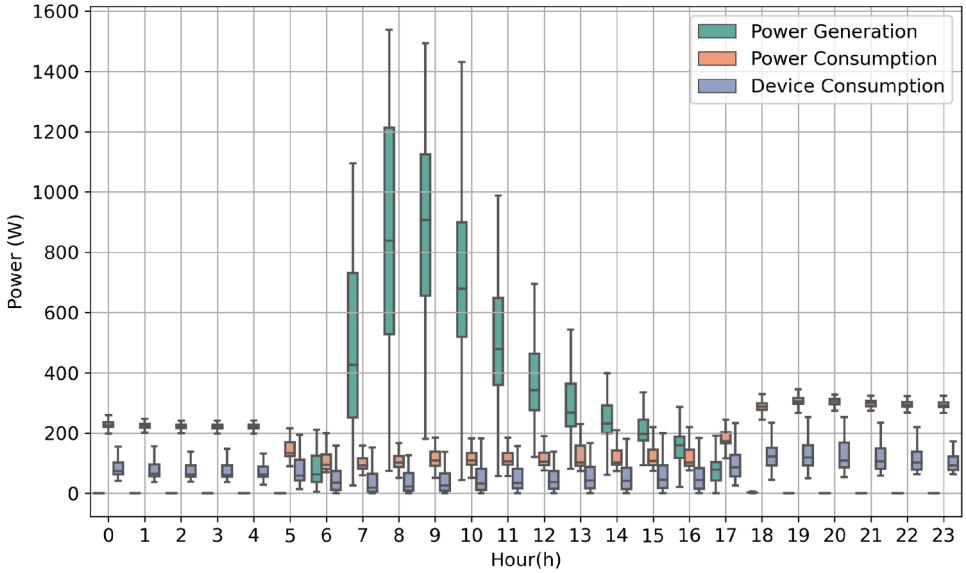
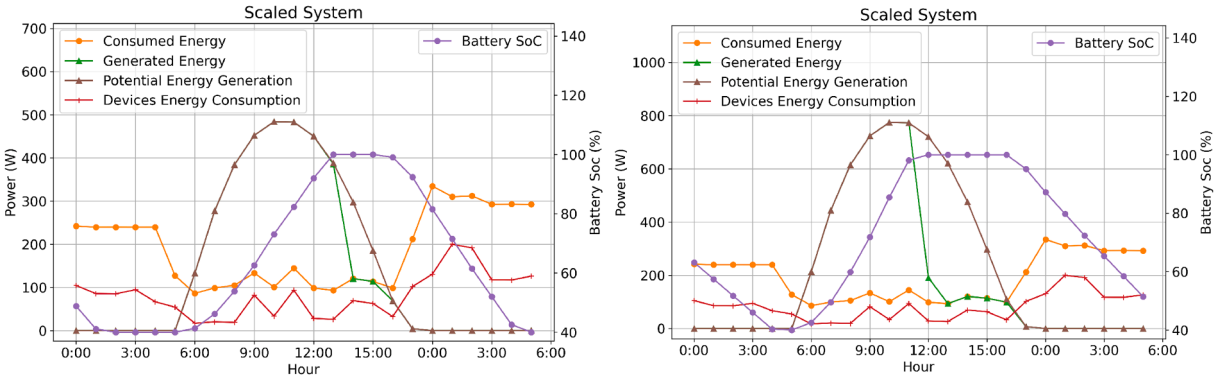


Fig. 6. Daily consumption and generation profile on the microgrid for the period 01 Jan 2020 - 12 Mar 2020.



(a) Small Scale Scenario (b) Medium Scale Scenario

Fig. 7. Scaling of the system for the testing and evaluation.

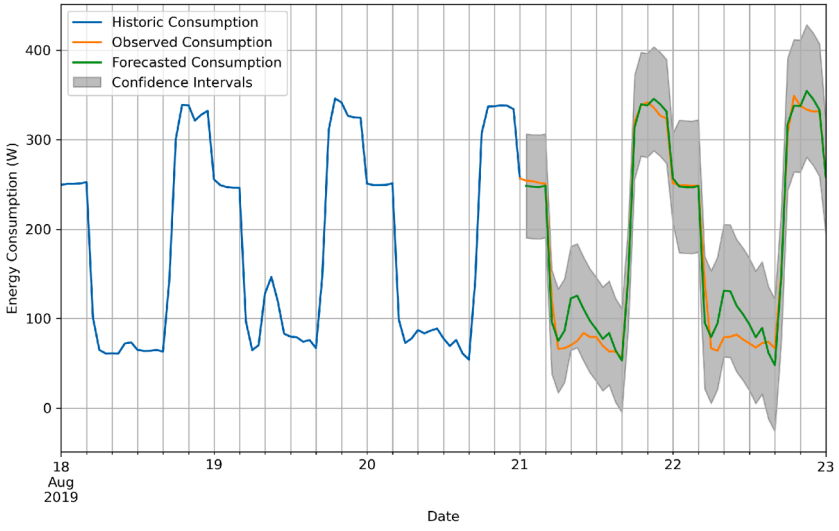
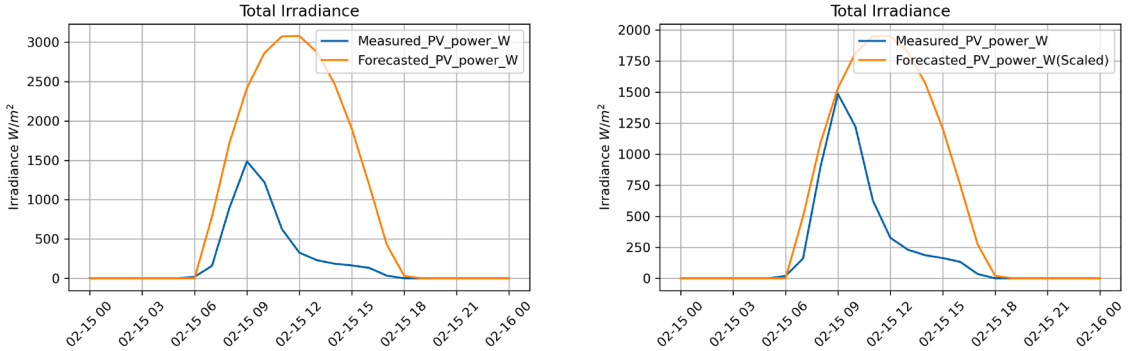


Fig. 8. 48h forecasting results on the total energy consumption.

Historic Hourly Data from 21 January 2020. Model fitness is evaluated using the AIC, which is an estimator of in-sample prediction error and thereby relative quality of statistical models for a given set of data. Best Orders for 3rd degree requires at least 8 days of previous data with best orders of $(1,1,2)x(0,2,2)$ and a seasonality of 24h. This resulted in an AIC of 5806.55. Best Orders for 2nd degree requires at least 3 days of previous data with best orders of $(1,1,1)x(1,1,1)$ and a seasonality of 24hours. This resulted in an AIC of 6161.69. The results shown in Fig. 8, present a model with orders of $(1,1,1)x(1,1,1)$ and a seasonality of 24h, showing that for our context SARIMA can provide the system with a reasonably accurate forecast.

For the PV panel generation forecasting, the predicted value is scaled a clearness index ktto account for cloudy weather that the Clearsky model cannot account for. Based on the existing data an average



(a) Forecasting without scaling of values (b) Forecasting with the added scaling of values

Fig. 9. Scaled and un-scaled forecast of data.

clearness index kt of 0.63 was selected. There is a drop-off in PV energy generation around 9 a.m. due to the battery being fully charged; this drop-off makes use of methods such as SARIMA difficult as the potential energy available is higher than the energy captured. The scaled and unscaled data comparisons can be seen in Fig. 9.

5. Results and discussion

5.1. Evaluating the modifications of GA

Our second modification (V2) of GA considers a chromosome to be the hourly values for each device group for a 24h period. This would mean that an individual I_i can be defined as a set of chromosomes Q^h_{Dev} where *Dev* is the device name, *h* is the time slot and Q^h_{Dev} contains the quota or limitation for that specific device in time slot *h*. The device quota Q^h_{Dev} is then represented as percentage of the total forecasted energy consumption for the device, which creates genes with values between 0.0 and 1.0 with 0.1 steps (i.e. 11 possibilities). While this version offers the most control over the quotas, it has the downside of creating a very large search-space that makes finding even local optima points resource intensive, which often yields results that are worse than simple methods.

The third version (V3) of GA implemented considers a pre-processing step that divides the period into sets in a similar way as the SSEM method. The segmentation can be done using a simple sign based splitting of the data that creates a new segment every time the sign of E^i_{Gen} -$E^i_{Cons}[d1 \ldots d2, sys]$ changes. Another option is using the Binary Segmentation Algorithm [36]. This means that in the typical case, the number of time slots that require quotas is reduced from 24 (for a 24h period) to $4 - 5h$ periods, which reduces the total number of variations for 7 devices across 24h from 8.3×10^{22} possible solutions to 2.7×10^{15}. Further reduction can achieved by selecting quotas with 20% increments leaving 1.8×10^9 possible solutions, which becomes a much more manageable number of iterations. However, for an typical time of 0.025 seconds to test a single solution, this still results in a computation time of 531 days. This means that the only reasonable variation that has an exact solution is one where there are 5 sets with three possible values [0,0.5,1.0] that can be solved in 17.6 minutes. The combination chosen for GA is using the sets divider and 6 possible values for quotas [0.0,0.2, 0.4,0.6,0.8,1.0].

All implemented variants of GA use a 20% Random Select of the population for mutate and crossover; 5% elitism; 25% crossover; 25% mutate; 25% crossover and mutate, and 20% new random individuals in each generation. Mutation is at 5% with each gene having a chance of receiving a new value.

The benefits of the search space reduction methods can be seen in Fig. 10, where the different versions of GA are compared on a run with 600 individuals and 100 generations. The *GA V1* method shows the original version of GA without any hyper-parameter tuning. The results from *GA V2* show how tuning the hyperparameters to match the use-case improved the algorithm. The results for *GA V3* shows the third version of the algorithm The best consistent results were reached by the *GA V3 Binary-Seg 0.33* which uses a binary segmentation to divide the 24h period into slots and a set of possible quota values of [0,0.33,0.66,1]. From the results we can see that the smaller step increment had an adverse affect on the results and that a more refined segmentation policy improved the results. For the purpose of further evaluations in the paper, the *GA V3 Binary-Seg 0.33* version is used.

5.2. Behaviour evaluation of the control methods

The GA V3 is compared with the built-in protection controller (PC) and devised space-shared method (SSEM) for maximising battery health for the small-scale scenario (Utility - U1). With this utility function, we expect the best solution would attempt to set the quota for all devices to 0.0 to increase the Battery SoC and reduce the Power draw, except in cases where the devices would receive their energy from the PV panels and there would be no penalty for using energy (in which case they can be on or off).

The behaviour of the PC can be seen in Fig. 11, where the left side showcases the effects of its control on the systems and the right side shows the quotas it imposes. In this scenario, the PC allows all the devices to use energy up to the point where the minimum Battery State of Charge SoC^{Min}_{Bat} threshold is reached, at which point all the devices receive a quota of 0.0 regardless of their priority or consumption.

For the space-shared energy manager (Fig. 11), all devices are on whilst there is energy available from the PV array and battery. At hour 4 p.m., when the PV array output starts to decrease (see Fig. 7a), the controller limits the playground sockets for 1 hour and a further 4 devices from hour 5 p.m. onwards. This behaviour means that there is enough energy to power all the outdoor lights until the end of the period. Starting the space shared energy manager at 6 a.m. ensures that the analysis period ends coincides with Rwanda's durial cycle, meaning that the battery is not depleted before sunrise. Thus, the communitys highest priority of keeping outdoor lights on at the micro-grid is achieved, which is not the case with the simple battery protect model.

The GA method takes a different approach by assigning quotas in hour sets: [0, 1, 2, 3, 4], [5, 6, 7, 8,9], [10, 11, 12, 13, 14], [15, 16, 17, 18, 19], [20, 21, 22, 23] to each device (see Fig. 13). For Utility function 1, GA only wants to maximise the battery health, which would be achieved by reducing the device loads to zero, except for cases where the

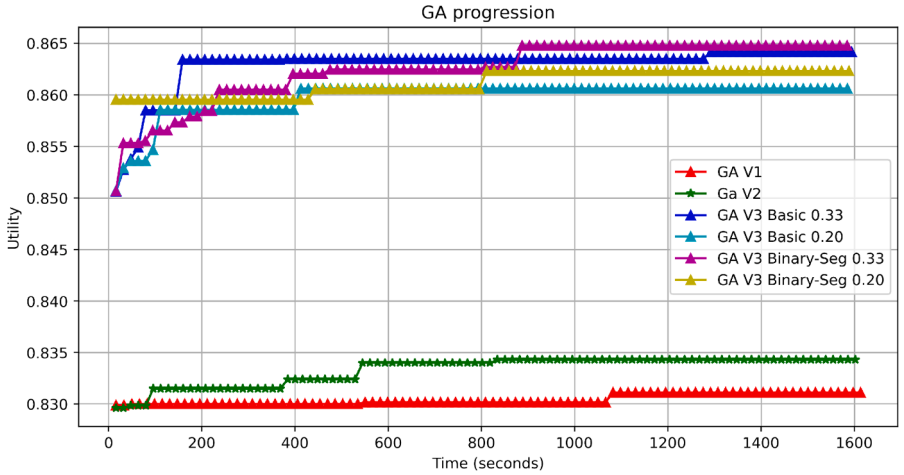


Fig. 10. Convergence of various GA methods on U1

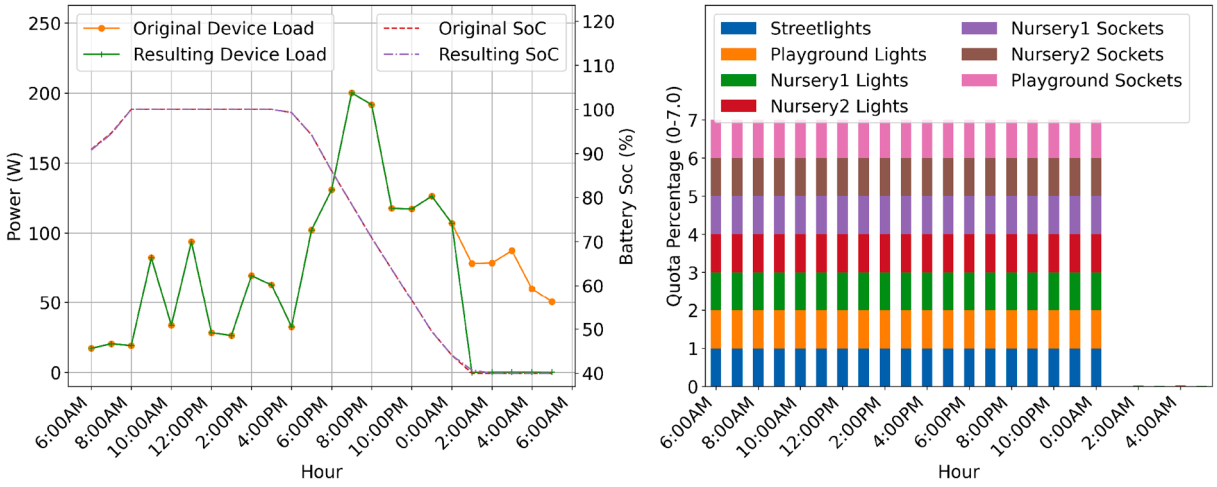


Fig. 11. Behaviour of the protection controller on a small scale scenario and the battery health considerations utility - U1 with a score of: 0.749

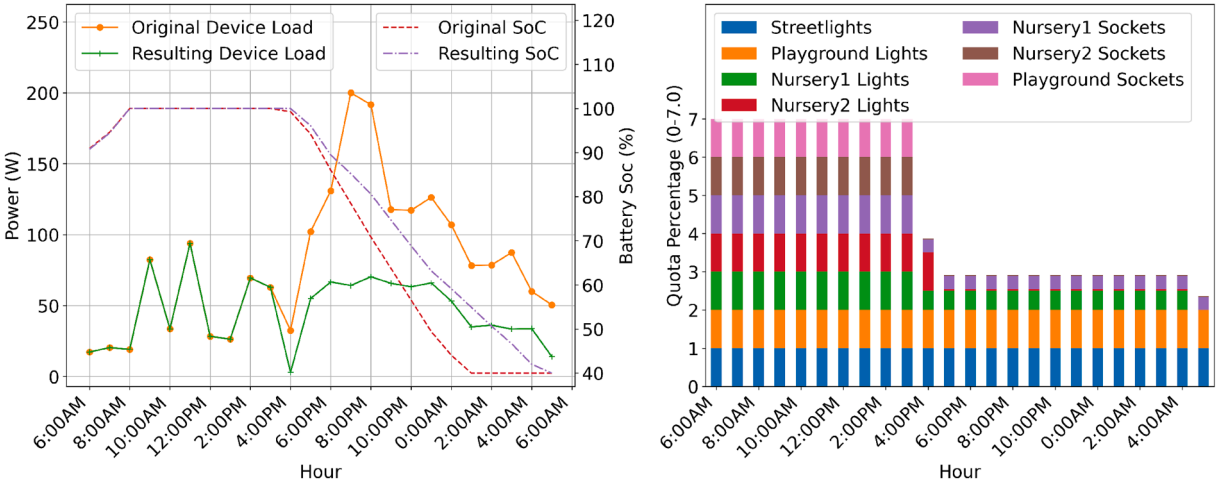


Fig. 12. Behaviour of the SSEM on a small scale scenario and the battery health considerations utility - U1 with a score of: 0.792

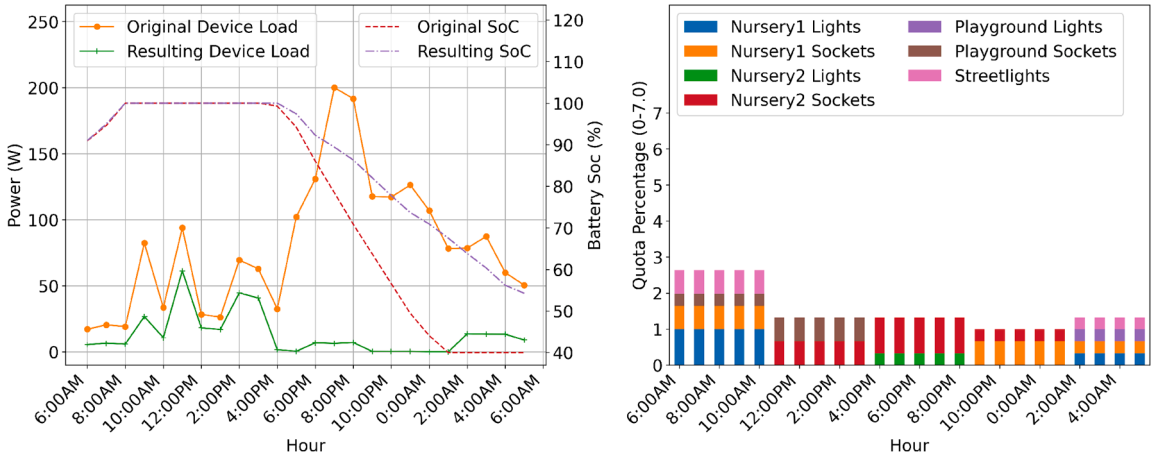


Fig. 13. Behaviour of GA V3 on a small scale scenario and the battery health considerations utility - U1 with a score of: 0.851

battery is charged, and there is more solar output than energy demand. GA is able to find a solution that is a local minima but was not able to find the absolute best solution. GA reached this solution in 1384 seconds, whereas the space shared method took 0.41 seconds.

5.3. *Method evaluation across the whole time-period*

To evaluate how well the methods perform throughout the whole period, the scenarios, utilities and methods were applied to the whole

Table 1
Resulting utilities and metrics of interest when the energy management methods are deployed for the whole time-period using different scenarios and utility function

Scenario	Utility Function	Method	↑ Utility Value	↑ SoCM	↑ nSM	↓ CS	↑ QM
small scale	U1	PC	0.20	64.47	**0.91**	**0.13**	**0.26**
		SSEM	0.26	71.50	0.91	0.29	0.18
		GA-V1	0.33	70.48	0.83	0.28	0.18
		GA-V3	**0.36**	**74.47**	0.77	0.43	0.13
	U2	PC	0.46	64.47	**0.91**	**0.13**	**0.26**
		SSEM	0.45	71.50	**0.91**	0.29	0.18
		GA-V1	0.54	67.64	0.87	0.19	0.21
		GA-V3	**0.62**	**74.19**	0.79	0.42	**0.26**
	U3	PC	0.72	64.47	**0.91**	**0.13**	0.26
		SSEM	0.64	71.50	**0.91**	0.29	0.18
		GA-V1	0.67	67.47	0.88	0.18	0.23
		GA-V3	**0.77**	**74.06**	0.80	0.41	**0.31**
medium scale	U1	PC	0.25	69.87	**0.69**	**0.07**	0.30
		SSEM	0.28	73.50	**0.69**	0.15	0.27
		GA-V1	0.37	77.23	0.62	0.28	0.18
		GA-V3	**0.39**	**82.71**	0.49	0.44	0.11
	U2	PC	0.55	69.87	**0.69**	**0.07**	**0.30**
		SSEM	0.55	73.50	**0.69**	0.15	0.27
		GA-V1	0.57	75.74	0.65	0.24	0.21
		GA-V3	**0.66**	**82.63**	0.50	0.43	0.28
	U3	PC	0.67	69.87	**0.69**	**0.07**	0.30
		SSEM	0.33	73.50	**0.69**	0.15	0.27
		GA-V1	0.71	71.16	0.91	0.19	0.23
		GA-V3	**0.78**	**81.34**	**0.69**	0.43	**0.31**

data-set. The resulting average utility function can be seen in Table 1 together with the metrics of interest outlined in equations: 3 for mean battery SoC; 5 for mean Capacity Shortage; 4 for mean system efficiency, and 6 for the total curtailment of devices adjusted by priority.

Each method was run for the whole period from the 3rd of January 2020 to the 28th of March 2020. The methods were run as a continuous deployment where the state of charge at the end of one day is the same as the start of the next one. Each method was run on the two scenarios and by utilising the 3 utility functions. For the methods with behaviour that is not influenced by the utility function, such as the SSEN and PC, the behaviour doesnt change across deployments; however, their utility value does as the quality of the result is evaluated with different metrics. The GA-V1 method represents the core GA available in most libraries with trained hyper-parameters while GA-V3 represents the modified GA method proposed in this paper.

The results for the $U1$ utility function help to identify how well the GA methods can optimise the quotas compared with two methods that are utility naive. From the results, we can see that in both scaled scenarios (see section 4.7), the best results are reached by GA-V3 that is on average 30% better than the PC, 36% better than the SSEM and 12% better than the original GA-V1.

From a processing time point of view, a comparison of the median value of 100 runs can be seen in Table 2. The tests were run as a single thread process on an Intel i7-2670QM CPU running at 2.2GHz. From the results, we can see that the GA based methods take considerably more time than the PC and SSEM methods, but with mean run-times under 5 minutes their use is still appropriate for our use-case. Given 7 device groups and the overall system consumption for which we need to forecast data, one PV forecast and single control run, for the PC and SSEM methods a full run-time will be under 3 minutes and for the GA version it will be under 6 minutes, even with the added overhead of the other components. When we consider that the forecast is only run hourly and the process is not required for a single run, this means that the PC and SSEM method would have responses under 5s, while the GA methods would be able to provide a new set of quotas in under 5 minutes.

The PC is one of the worst performers when it comes to battery health with the average Soc being the lowest of all the methods. This behaviour, however, allows this controller to reach the maximum possible system efficiency (nSM) and capacity shortage (C_s). Despite these values, as all of the devices are turned off in specific periods, the QM values are in most cases lower than for the $U2$ and $U3$ methods when the GA methods were applied.

In most cases, the SSEM method performs poorly in comparison to the PC. One of the SSEM method's main aims is to keep the system running, meaning that it will turn off devices to keep the main system components and loads online. This can be seen from it having a consistently and considerably higher mean SoC value than the PC.

The GA-V3 method outperforms GA-V1 and other methods consistently across the tests where we can see that for U1 it manages to increase the average SoC values from 0.33 and 0.37 to 0.36 and 0.39 as compared to the GA-V1. For $U2$ and $U3$ the Quota values are considerably higher for GA-V3 while also increasing the mean SoC value for U2 and only sacrificing 8%(for the small scale) and 6%(for the medium scale) of the system efficiency. The higher capacity shortages for GA-V3 but better scores are a result of the method identifying lower priority devices and adding higher quotas to these maximising battery health gains while minimising user dissatisfaction.

It is interesting to note that despite GA-V1 (or out of box methods available in Matlab and other platforms) being a popular option for micro-grid control optimisation, GA-V1 fails to provide any best-in-class solutions for any of the utilities and is even sometimes outperformed by the PC and SSEM model (see table 1). This shows that, for high complexity or search space problems where the utility is well defined, un-optimised GA may fail to find good solutions, and simple decision-making algorithms can be a better initial choice.

The GA-V3 outperforms all methods when considering the resulting utility function, which means that it can optimise the quotas to meet the proposed utility functions. Whilst GA-V3 performed best for balancing the alternative objectives for each utility function, this control method did result in the highest capacity shortage and lowest system performance across all scenarios. Therefore modified utility functions may be depending on the needs for different systems, applications, locations and communities.

6. Further work

The overlay design allows legacy micro-grid designs to upgrade their control capability with the use of low-cost computational resources. While this overlay was shown to be working and can satisfy the requirements of the community it does incur considerable latencies (with a mean latency of 53.6 minutes) when deployed. More work determining the latency requirements of the various first and second stage controllers is needed to decide which legacy components could be replaced to allow higher precision energy managers to be deployed. Due to the communication delays on the system, the optimisation method is needed to find optimal values for the upcoming 24h period. Given a lower latency system, the use of an exponentially increasing time step (0.5,1,2,4,8,12,16,24,36...) for the GA or PSO optimisation may allow higher precision to be reached for the current time periods while considering future periods as well.

The translation of hourly quotas into periodically changing daily

Table 2
Mean run-times is seconds for various methods given a medium-scale scenario and U2 utility function

Method	SARIMAX Training	SARIMAX Prediction				
	PV Forecast	PC	SSEM	GA-V1	GA-V3		
Processsing Time (s)	16.5	0.026	0.12	0.69	1.10	165.8	211.2

quotas by a control enactor can be imprecise if the forecasted consumptions vary greatly and can cause unwanted behaviour due to delays in the first layer controller receiving the quotas late. The robustness to changes in demand, as opposed to latency changes, is a topic that is worth investigating and can lead to new requirements and constraints when it comes to the design of controllers. The micro-grid system used as a case study in this research saw a low utilisation of sockets, and hourly, daily and monthly variability was also low. Therefore, control methods investigated in this paper need to be further evaluated for a wider range of system types, where there are greater changes and variations in energy demands.

Further work can be done on comparing the results for each method with typical energy dispatch models, cost service models and producer/supplier models. More social work can also be done to identify the various requirements and their respective utilities in differing communities to satisfy a wider range of requirements, e.g., refugees may want cheaper energy if only available at certain times, which could also increase system longevity. Identifying competing governance methods and their underlying technological requirements around these systems would also improve understanding and provide better utility for the communities they serve.

7. Conclusions

The deployment and use of community microgrids are playing an increasingly important role in electrification in remote and displaced settings. Yet, existing control solutions are often limited to focus primarily on costs rather than user needs and system longevity. Therefpre, the management of energy in these systems is crucial to maximising the benefits that these systems can offer and support communities in managing energy as a shared resource. The modified GA method employing search-space reduction techniques outlined in this paper was shown to provide significant advantages over other control methods. Whilst the proposed system architecture can be easily adapted to suit other community microgrids installations.

Each method's latencies and processing times can also be factors when considering whether a method is appropriate. If the microgrid use-case is subject to highly varying usage patterns, the 1h communication latency coupled with a 2h processing time for a GA method can result in unwanted behaviour, and the energy needs not being met by devices not being turned off in an effective and timely manor. These systems would benefit from a low processing time (SSEM) method that is deployed locally.

Our findings also show that the resulting search spaces can be larger than a resource-constrained computational unit can solve using conventional metaheuristic methods. Therefore, microgrid developers and operators need to be aware of some of the challenges of defining high-complexity objective functions and considerations when selecting a suitable method for different community requirements.

8. Funding

This research was funded under the Humanitarian Engineering and Energy for Displacement (HEED) project financed by Research Councils UK (EP/P029531/1).

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors would like to acknowledge the financial support of the Engineering and Physical Science Research Council (EPSRC) for funding the Humanitarian Engineering and Energy for Displacement (HEED) project as part of the Global Challenges Research Fund (EP/P029531/1). The authors would like to acknowledge and thank project delivery partners Practical Action and Scene Connect for their significant role in co-ordinating in-camp activities and providing technical inputs and tools. We would also like to recognise the support of MIDIMAR (Ministry of Disaster Management and Refugees) and UNHCR (United Nations High Commissioner for Refugees) and the contributions of the Global Plan of Action, Chatham House, and the RE4R (Renewable Energy for Refugees) Project (a partnership between Practical Action and UNHCR, supported by the IKEA Foundation).

References

[1] J. Lehne, W. Blyth, G. Lahn, M. Bazilian, O. Grafham, Energy services for refugees and displaced people, Energy Strategy Reviews 13 (2016) 134–146.
[2] U. H. C. for Refugees, Global strategy for sustainable energy; a unhcr strategy 2019-2024, 2019, (https://www.unhcr.org/uk/partners/projects/5db16a4a4/global-strategy-for-sustainable-energy.html).[Online; posted 22-February-2021].
[3] V.C. Johnson, S. Hall, Community energy and equity: the distributional implications of a transition to a decentralised electricity system, People, Place & Policy Online 8 (3) (2014).
[4] S. Rosenberg-Jansen, Leaving no one behind: An overview of governance of the humanitarian energy sector, Energy Access and Forced Migration (2019) 15–33.
[5] G. Elena, B. James, C. Heaven, J. Brandi, V. Nandor, EPSRC HEED data repository: Surveys, 2021, 10.5281/zenodo.4454580.
[6] B. Soudan, A.M. Darya, Modular hybrid energy system for refugee camps. 2021 6th International Conference on Renewable Energy: Generation and Applications (ICREGA), IEEE, 2021, pp. 33–38.
[7] N.N.A. Bakar, M.Y. Hassan, M.F. Sulaima, M.N. Mohd Nasir, A. Khamis, Microgrid and load shedding scheme during islanded mode: A review, Renewable and Sustainable Energy Reviews 71 (2017) 161–169, https://doi.org/10.1016/j.rser.2016.12.049.https://www.sciencedirect.com/science/article/pii/S1364032116311030
[8] D. Schnitzer, D. Lounbury, J. Carvallo, R. Deshmukh, D. Kammen, Microgrids for rural electrification: a critical review of best practices based on seven case studies usa: United nations foundation, línea]. Disponible en: https://rael. berkeley. edu/publication/microgrids-for-rural-electrification-a-critical-review-of-best-practices-based-on-seven-case-studies (2014) 120.
[9] S. Wang, D. Guo, X. Han, L. Lu, K. Sun, W. Li, D.U. Sauer, M. Ouyang, Impact of battery degradation models on energy management of a grid-connected dc microgrid, Energy 207 (2020) 118228, https://doi.org/10.1016/j.energy.2020.118228.https://www.sciencedirect.com/science/article/pii/S0360544220313359
[10] C. Bordin, H.O. Anuta, A. Crossland, I.L. Gutierrez, C.J. Dent, D. Vigo, A linear programming approach for battery degradation analysis and optimization in offgrid power systems with solar energy integration, Renewable Energy 101 (2017) 417–430, https://doi.org/10.1016/j.renene.2016.08.066.https://www.sciencedirect.com/science/article/pii/S0960148116307765
[11] C. Ju, P. Wang, L. Goel, Y. Xu, A two-layer energy management system for microgrids with hybrid energy storage considering degradation costs, IEEE Transactions on Smart Grid 9 (6) (2018) 6047–6057, https://doi.org/10.1109/TSG.2017.2703126.
[12] S.M.S. Kalajahi, H. Seyedi, K. Zare, Under-frequency load shedding in isolated multi-microgrids, Sustainable Energy, Grids and Networks 27 (2021) 100494, https://doi.org/10.1016/j.segan.2021.100494.https://www.sciencedirect.com/science/article/pii/S2352467721000655
[13] L. Luo, S.S. Abdulkareem, A. Rezvani, M.R. Miveh, S. Samad, N. Aljojo, M. Pazhoohesh, Optimal scheduling of a renewable based microgrid considering photovoltaic system and battery energy storage under uncertainty, Journal of Energy Storage 28 (2020) 101306, https://doi.org/10.1016/j.est.2020.101306. https://www.sciencedirect.com/science/article/pii/S2352152X19309466
[14] D.R. Jones, M. Schonlau, W.J. Welch, Efficient global optimization of expensive black-box functions, Journal of Global optimization 13 (4) (1998) 455–492.
[15] M.S. Kadu, R. Gupta, P.R. Bhave, Optimal design of water networks using a modified genetic algorithm with reduction in search space, Journal of Water Resources Planning and Management 134 (2) (2008) 147–160.
[16] E. Goncalves, A.R. Balbo, D.N. da Silva, L. Nepomuceno, E.C. Baptista, E.M. Soler, Deterministic approach for solving multi-objective non-smooth environmental and economic dispatch problem, International Journal of Electrical Power & Energy Systems 104 (2019) 880–897.
[17] Y.D. Sergeyev, D. Kvasov, M. Mukhametzhanov, On the efficiency of nature-inspired metaheuristics in expensive global optimization with limited budget, Scientific reports 8 (1) (2018) 1–9.
[18] B. Homan, G.J. Smit, R.P. van Leeuwen, V. Marnix, B. Ten, A comprehensive model for battery state of charge prediction. 2017 IEEE Manchester PowerTech, IEEE, 2017, pp. 1–6.
[19] S. Muzaffar, A. Afshari, Short-term load forecasts using lstm networks, Energy Procedia 158 (2019) 2922–2927.
[20] G.E. Box, G.M. Jenkins, Time series analysis: forecasting and control San Francisco, Calif: Holden-Day (1976).

[21] A. Nespoli, E. Ogliari, S. Leva, A. Massi Pavan, A. Mellit, V. Lughi, A. Dolara, Day-ahead photovoltaic forecasting: A comparison of the most effective techniques, Energies 12 (9) (2019) 1621.
[22] P. Ineichen, A broadband simplified version of the solis clear sky model, Solar Energy 82 (8) (2008) 758–762.
[23] J.A. Duffie, W.A. Beckman, N. Blair, Solar Engineering of Thermal Processes, Photovoltaics and Wind, John Wiley & Sons, 2020.
[24] R. Deo, P. Samui, S.S. Roy, Predictive Modelling for Energy Management and Power Systems Engineering, Elsevier, 2020.
[25] R. Xiong, L. Li, J. Tian, Towards a smarter battery management system: a critical review on battery state of health monitoring methods, Journal of Power Sources 405 (2018) 18–29.
[26] S. Khurana, K. Marwah, Performance evaluation of Virtual Machine (vm) scheduling policies in cloud computing (spaceshared & timeshared). 2013 Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT), IEEE, 2013, pp. 1–5.
[27] U.H.C. for Refugees, Operational data portal (ODP) - refugee situations, 2021a, (https://data2.unhcr.org/en/country/rwa). [Online; visited 23-September-2021].
[28] U.H.C. for Refugees, UNHCR Rwanda Kigeme Camp profile april 2021, 2021b, (https://data2.unhcr.org/en/documents/details/86480, note = "[Online; visited 01-October-2021]",b).
[29] P. Sandwell, T. Tunge, A. Okello, L. Muhorakeye, F. Sangwa, L. Waters, T. Kayumba, S. Rosenberg-Jansen, Ensuring refugee camps in rwanda have access to sustainable energy, Practical Action (2020).
[30] P. Palensky, D. Dietrich, Demand side management: Demand response, intelligent energy systems, and smart loads, IEEE transactions on industrial informatics 7 (3) (2011) 381–388.
[31] E. Ostrom, Governing the commons: the evolution of institutions for collective action, Cambridge University Press, 1990.
[32] Z. Zhang, J. Wang, T. Ding, X. Wang, A two-layer model for microgrid real-time dispatch based on energy storage system charging/discharging hidden costs, IEEE Transactions on Sustainable Energy 8 (1) (2016) 33–42.
[33] P. Vassiliadis, A. Simitsis, S. Skiadopoulos, Conceptual modeling for ETL processes. Proceedings of the 5th ACM International Workshop on Data Warehousing and OLAP, 2002, pp. 14–21.
[34] V. Nandor, Heed-microcontroller, 2019, (https://github.com/cogent-computing/Heed-Microcontroller). 10.5281/zenodo.4020685.
[35] H. Kang, The prevention and handling of the missing data, Korean journal of anesthesiology 64 (5) (2013) 402.
[36] P. Fryzlewicz, et al., Wild binary segmentation for multiple change-point detection, Annals of Statistics 42 (6) (2014) 2243–2281.