HEED

Coventry University

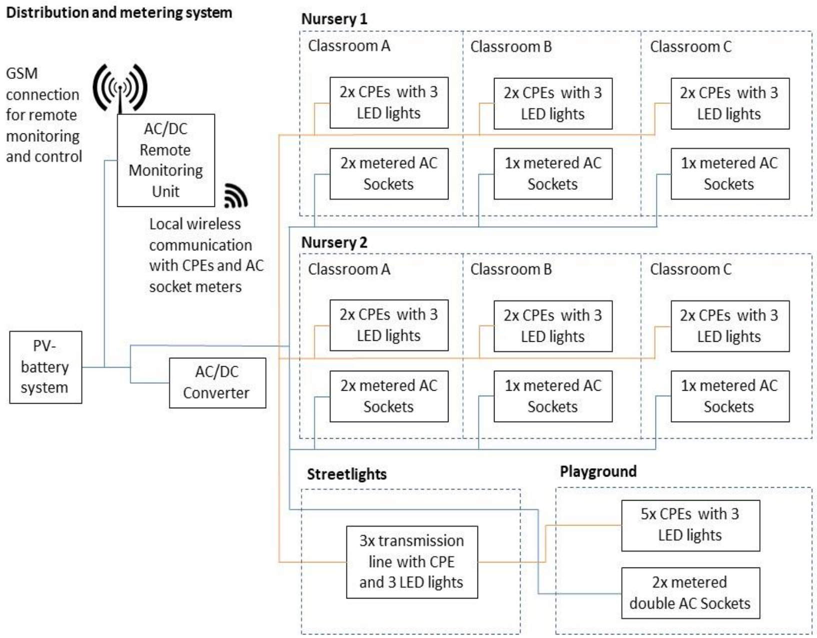
Microgrid Controller

Nandor Verba, Leonardo Alves Dias

08 Oct 2020

Meshpower System

- GSM connection with the Meshpower central Server
- Local Telnet interface can be used to attach extensions
- Local system uses LoRaWAN
- Meshpower Quota system can be used to impower power and time limits on sockets and lights.



Quota Info and
access on Meshpower Slides.

Motivation

The microgrid controller and interface aims to empower the local community to take ownership of their energy system by allowing them to monitor their power consumption and generation and allowing them to priorities their buildings.

Main Research Questions

- How can end-user negotiations inform optimal micro-grid system energy dispatch control strategies and reduce the cost of energy generated by the given infrastructure?
- How could a negotiation-based dispatch model (and other new dispatch algorithms tailored for remote and refugee camp settings) reduce the cost of energy and change/improve existing systems deployed in similar contexts or of similar size in other contexts?
- What energy consuming devices become critical over time and how does this influence system operation and negotiation trends?

HEED

Humanitarian Engineering and Energy for Displacement

Interface - Concept

"The interface aims to enable common pool resource management by the microgrid users, managers and building owners. Promoting understanding of how microgrids can function in humanitarian contexts and the implications of common pool and community-level management of energy systems."

	Principle	Rationale	Hypothesis
1	Defined System Boundaries	The installed microgrid system has defined boundaries, including max load, available energy (total and temporal), entitled users and number of connections.	Users must manage energy use dynamically to ensure system works within defined boundaries
2	Benefit-cost equivalence	Users must negotiate in a way which provides benefits for their contributions.	Users will interact with the UI / other users to ensure they can benefit from access to energy.
3	Collective Choice	Users must be able to set their own rules to ensure energy is delivered where it is needed and that conflicts do not arise, resulting in blackouts and user conflict.	Users will engage mutually to decide energy priorities, subsequently inputting decisions into the system via the UI.
4	Monitoring	The microgrid system and UI will provide a means of monitoring energy availability, SOC and user priorities.	Users will alter their behaviours in accordance with monitoring information provided by the UI.
5	Graduated Sanctions	Ineffective or conflicting use of the system will lead to unequal energy availability and system wide inefficiency (i.e. blackouts / reliability of power). Sanctions will be socially constructed and influenced through the monitoring data available via the UI	Users will develop a locally appropriate form of sanctioning users who do not interact with the system in a fair and equal way.
6	Conflict Resolution	Conflict resolution (energy and social) will be informed by system function and UI information	Users will develop means of conflict resolution to enable effective system functionality.
7	Local Autonomy	The system will function autonomously without users input. Users will be able to priorities specific loads within specific timeframes.	Users will govern their energy autonomously without the need for external inputs or governance.
8	Polycentric Governance	The system will allow for non-hierarchical governance across a number of users with their own 'energy communities' (i.e. building or service users).	Users will manage the microgrid system via the UI in a way which benefits their community whilst taking into account the needs of the other microgrid managers / users.

Interface – User Design

HEED
Humanitarian Engineering and Energy for Displacement
Coventry University

"User feedback was sought from potential microgrid system users; microgrid managers; building managers and wider relevant camp stakeholders at A workshop will be conducted in Kigeme camp, Rwanda in November 2019. The aim of the process will be to design a solution which is both locally applicable and understood."

Event Format

Introduction → Microgrid discussion forum → Shared resource games workshop → User interface presentation → Testing workshop → Feedback session → Wrap up / Questions


Microgrid discussion forum


Shared resource games workshop


Testing workshop

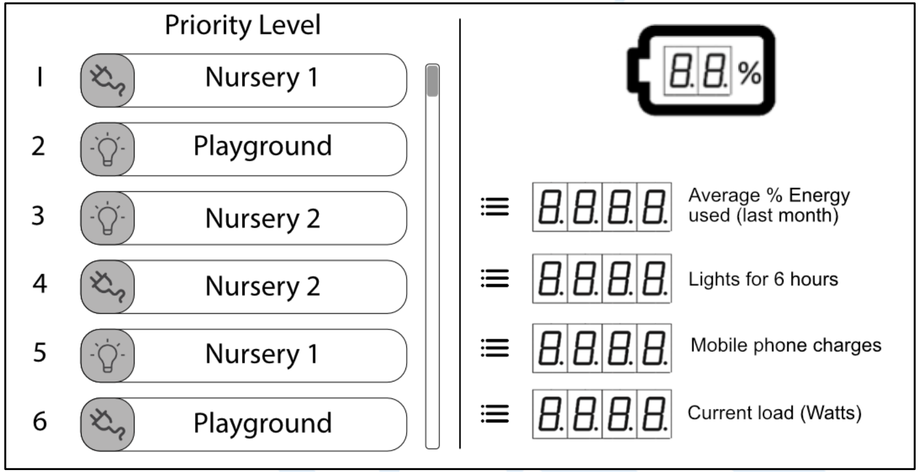
Interface – Design Requirements

	Feature	Location	Description
FR1	Energy charging	Interface	Needs to be able to display System information about the state of charging of the system (i.e. charging / not-charging).
FR2	State of charge	Interface	Needs to be able to display System information about the current state of charging and/or time to discharge.
FR3	Device related energy availability	Interface	Needs to provide information on the number of device charges (e.g. mobile phones) left in the system or number of hours of lighting left. ~10 metrics to be provided to users, with the ability to switch between variables via a small menu dropdown.
FR4	Prioritisation	Interface	Needs to provide users with a drag and drop style prioritisation system, similar to Trello boards. Users must be able to move each board and reposition in the queue. It is expected that there will be 6 – 9 separate boards.
FR5	Connectivity	Interface	Interface must have a suitable solution for dealing with connection drops. This may include providing static values based on last known figures, showing null values after a fixed point in time (i.e. when static figures are considered too inaccurate), removing prioritisation functionality when disconnected and displaying a warning of connectivity status to users attempting to use the prioritisation system.
NFR1	Language	-	All language prompts must be in English and Kinyarwanda via a language selection icon.
NFR2	Icons	-	Where possible, icons should be used in place of text.
NFR3	Simple display	-	The interface should have information displayed in an easily understandable and quickly readable manner.
NFR4	Device reset	-	Interface must be easily reset and troubleshooted by in-camp users or staff.

Interface – User Design

User Needs	A means to understand current energy availability
	Ability to decide who gets energy and when
	Ability to understand future short term energy availability
	Ability to reserve / provide energy for specific high energy demand events
Intended Use	Viewing of current energy generation status and battery state of charge
	Viewing of predicted future state of the system for the upcoming day
	Viewing current and future prioritisation of lighting / sockets in the microgrid
	Input of new priorities for lighting / sockets in the microgrid

Example User Interface Design



Simple user prioritisation of microgrid locations and energy use types (plugs Vs lighting)

Live system metrics

Interface – Development Status

COMPLETE

1. User-design process
2. System mock-ups
3. Functional & Non-functional requirements (next slide)
4. Live user interface with connection to microgrid data
   http://cogentee.coventry.ac.uk/heed_interface/
5. Open-source code repository for user interface design code

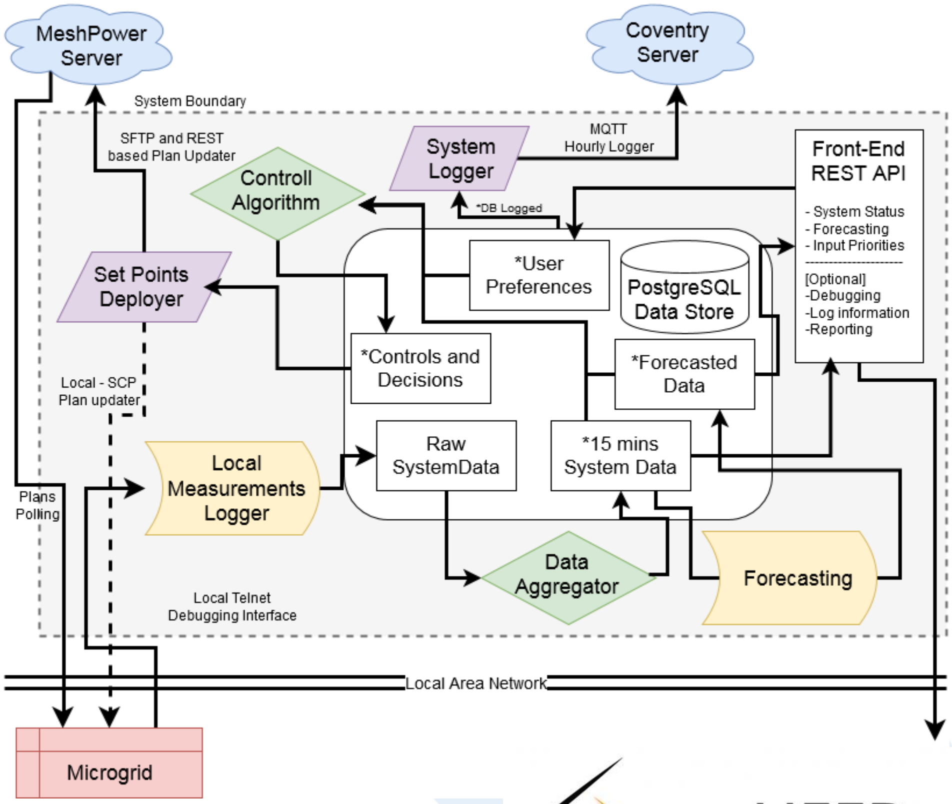
INCOMPLETE

1. Lab testing with microgrid controller and algorithm integration
2. In-situ testing at microgrid site (Kigeme Refugee Camp)
3. System monitoring and evaluation (including user feedback)
4. System design review and iteration 2.0

Video Presentation – Here.

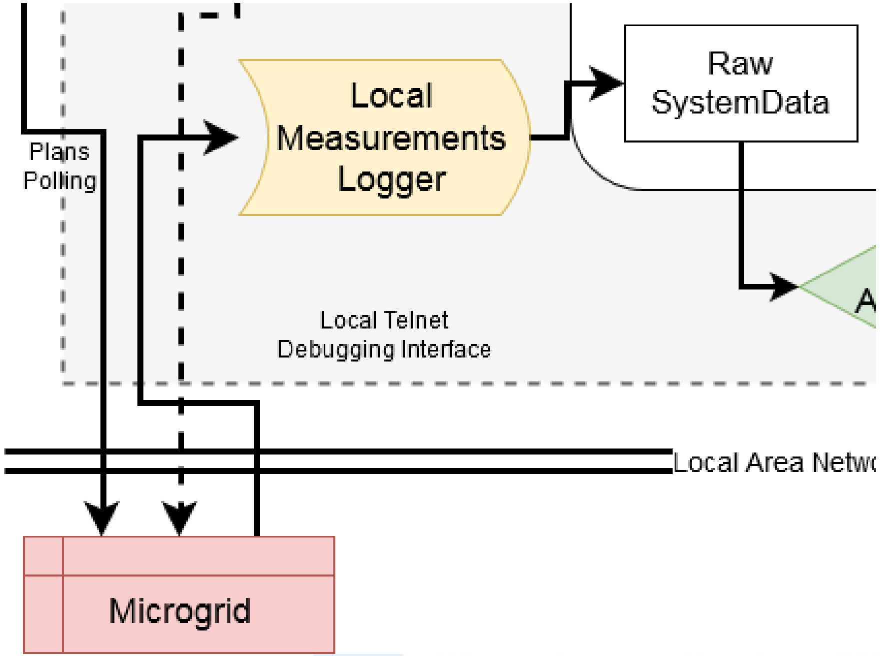
Architecture

- All components run independently with a database linking them together through shared data.
- Components deployed as docker services where appropriate to increase reliability and control over them.
- System state to be logged locally and sent to a central server through MQTT
- Local and Web based gathering of system data is possible
- Quotas are deployed using the MeshPower system.
- The code repository can be found [here.](#)



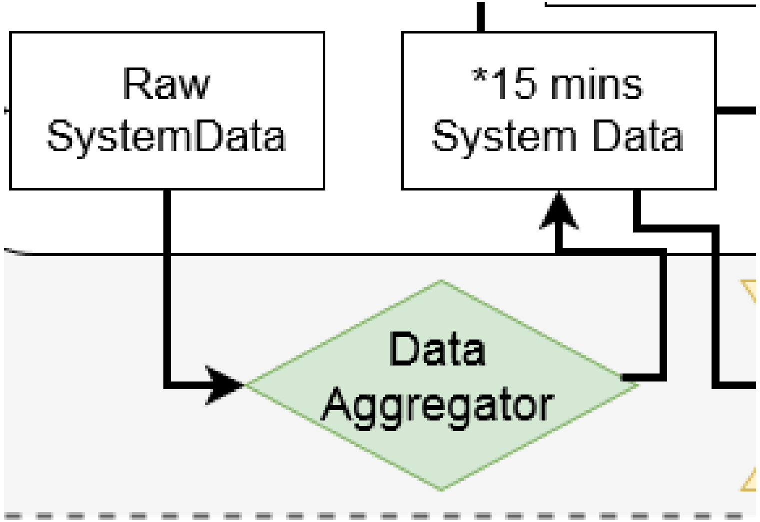
Architecture – Local Data Collector

- Collects data based on device mac addresses from the MeshPower Microgrid system every 15 seconds using their telnet debugging interface
- Collects System Load, Generation and State of Charge (SoC) data provided by the VenusGX systems from the same Meshpower Microgrid debugging interface that provides the Instantaneous Device Load and Session Load information.



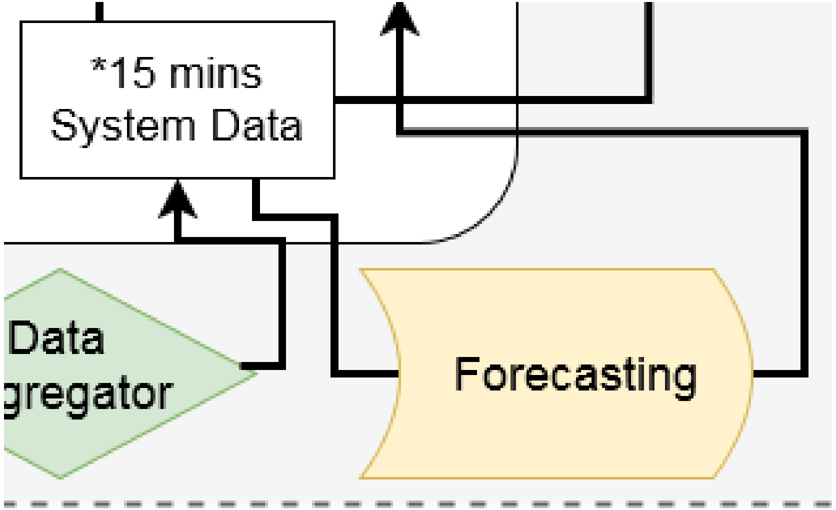
Architecture - Aggregator

- Read is the raw System Data values from time periods where there is no aggregation information available
- If data for a certain device on that hour is not available NaN values are insterted into the databse.
- Cleaning and imputation is performed in later staged.



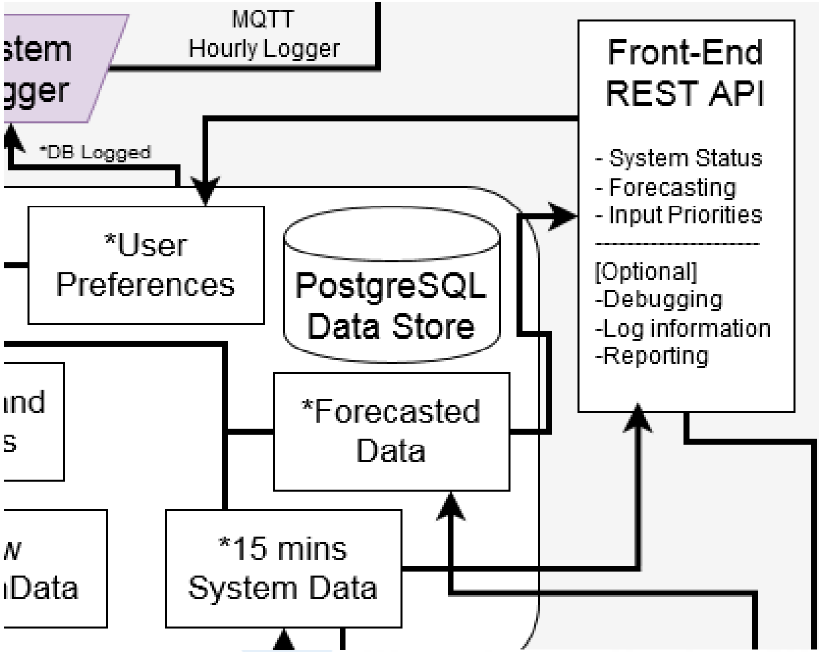
Architecture - Forecasting

- Forecasting reads values from the 15 minute aggregated data and generates a forecast for each consumer and producer of energy.
- Cleaning and aggregation steps detailed later in the slides.
- The forecasted data is hourly, is 24h hours ahead and is updated every 15 minutes.
- The algorithm is detailed in future slides



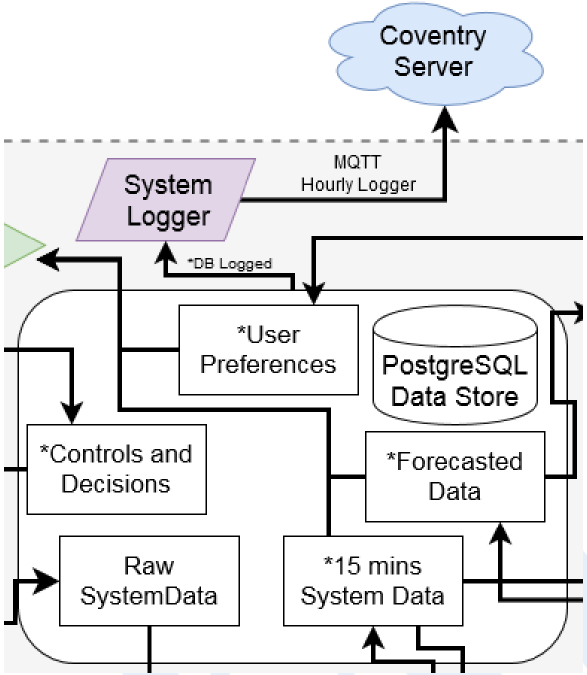
Architecture – Front End

- Provides and API for the Scene interface
- Uses forecasted and historic data to provide required information on :
  - battery energy available (kWh)
  - power consumption (kW)
  - solar pv power generation (kW)
  - solar pv energy generation 24h (kWh
  - solar pv energy generation 30days (kWh)
  - energy consumption 24h (kWh
  - energy consumption 30days (kWh)
- It also allows the interface to send back user priority preferences that is saves to the database



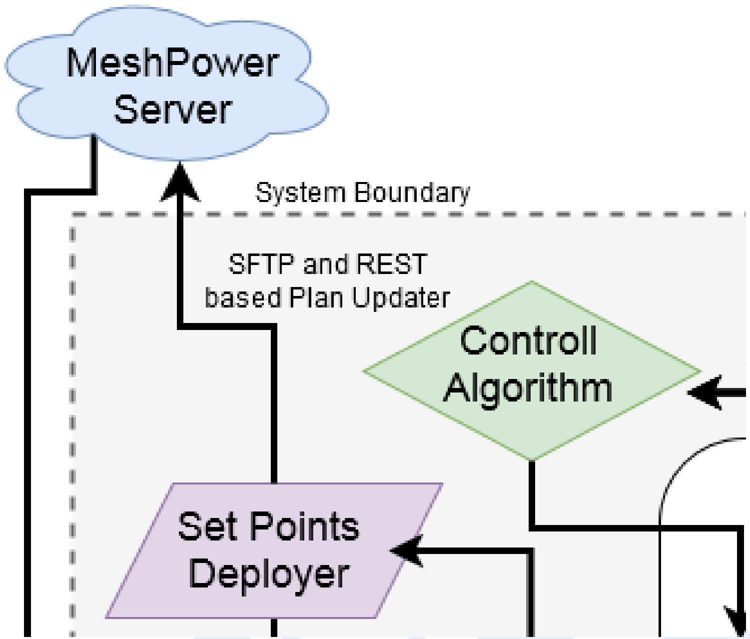
Architecture – System Logger

- The System Logger creates a snapshot of the systems every 15 minutes, saves it to the database and sends the log via MQTT to a Coventry university server.
- The parameters are detailed in the upcoming slides



Architecture – Controller and Enactor

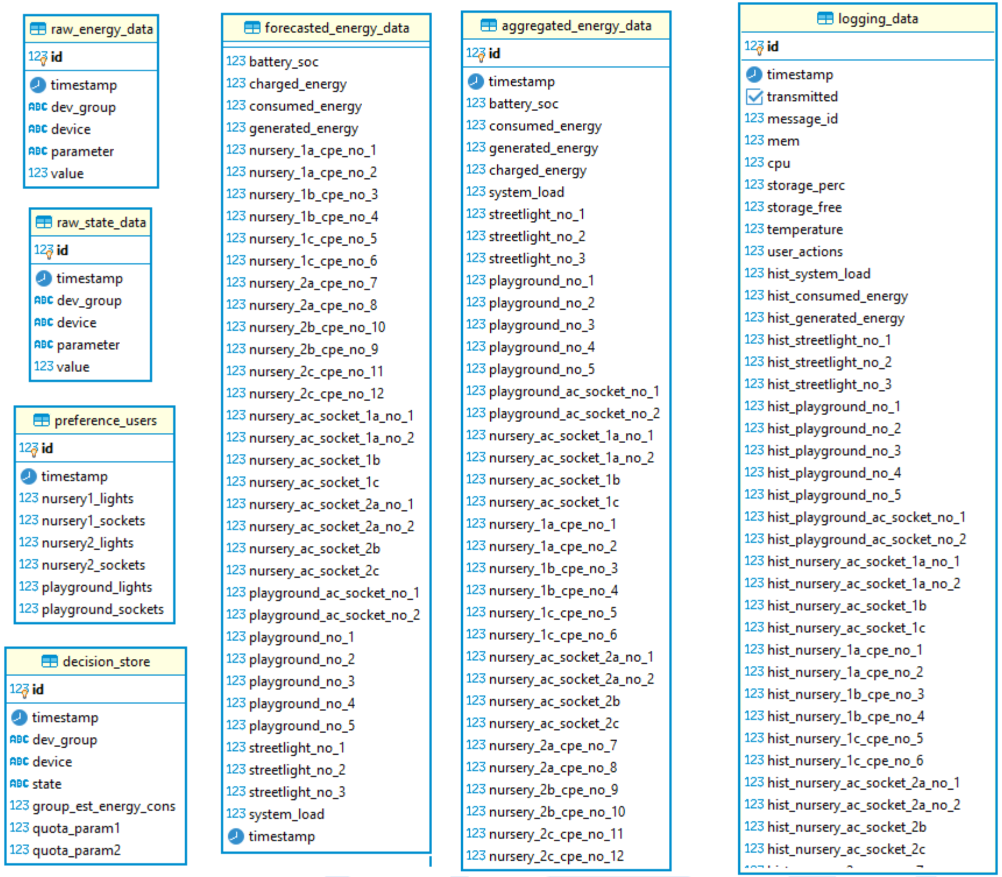
- The details on how the controller decides on quotas is below
- The enactor retrieves existing quotas from the MeshPower sftp server and checks whether these have changed. If they have it updates them.
- After a Quota is updated it needs to be enabled on their website through a WebSpoofer element.



HEED
Humanitarian Engineering and Energy for Displacement

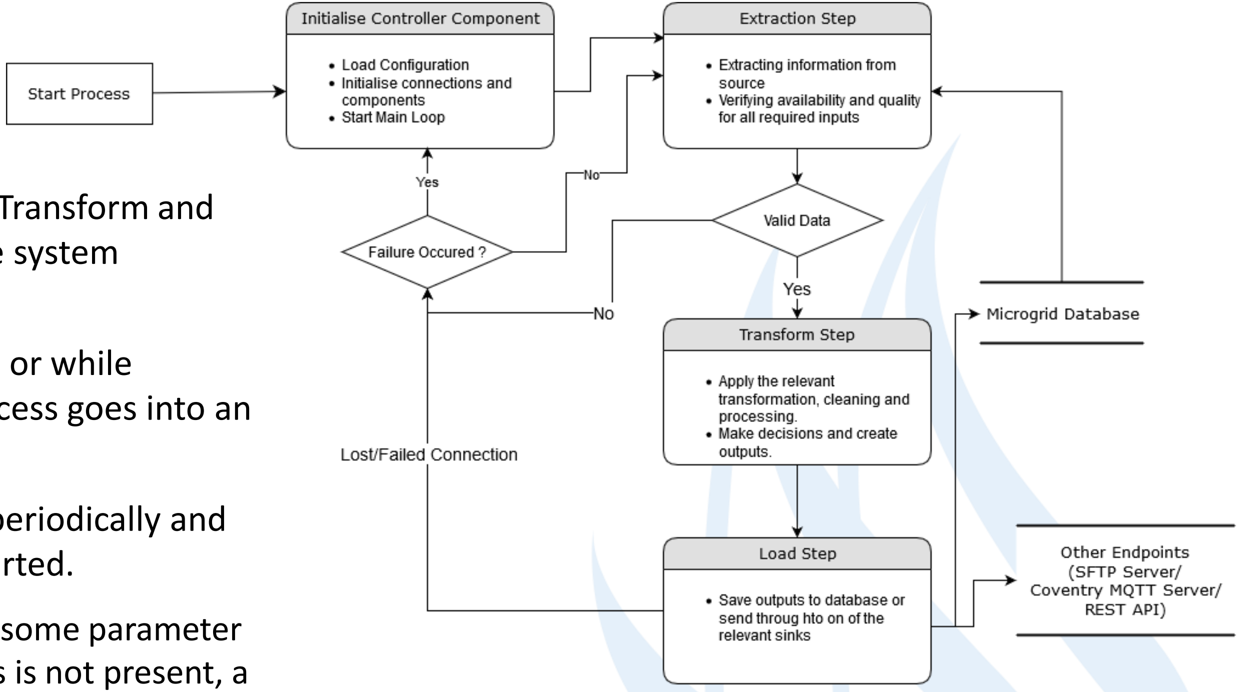
Database structure and diagram

- Independent databases with no foreign keys are created for each component so there are no issues with sparse population of resources
- The raw_energy_data and raw_state_data hold a single entry for each data-point.
- The use preferences hold the order number for each device group
- The decision store contains the stored quota information for each specific device and group, here the lights have two quotas NightLight and BrightLight while the sockets only have one and param2 is a None
- The structure of the forecasted energy_data is the same as the aggregated energy data. The forecasted database is re-built with each new forecast.
- The logging data contains a snapshot of the current system state at a given timestamp, together with message id and transmission information one sending the report to the central Coventry server.



Flow Diagram and design principle

Research Centre Manufacturing and Materials Engineering | Coventry University



- Components follow the Extract, Transform and Load design principle to increase system resilience.
- If any errors occur to connection or while retrieving or saving data the process goes into an error state.
- All the processes are inspected periodically and the ones in error states are restarted.
- Invalid data can either mean that some parameter that is required to run the process is not present, a connection is lost or the data source has no available data.

Parameters and Calculations Deployed System

- System Parameters
  - Transmitted; message_id; mem; cpu; storage_perc; storage_free; temperature; user_actions
- Historic Values
  - hist_system_load; hist_consumed_energy; hist_generated_energy – Per_Device_Values
- Forecasted - Per_Device_Values
- Quota 1;Quota2; Priority per Device Groups
- 15 minute cleaned Average for Power consumption
  - Nursery1_Lights, Nursery1_Sockets, Nursery2_Lights, Nursery2_Sockets, Playground_Lights, Playground_Sockets, Streetlights
- Devices (30)
  - streetlight_no_1 streetlight_no_2 streetlight_no_3 playground_no_1 playground_no_2 playground_no_3 playground_no_4 playground_no_5 playground_ac_socket_no_1 playground_ac_socket_no_2 nursery_ac_socket_1a_no_1 nursery_ac_socket_1a_no_2 nursery_ac_socket_1b nursery_ac_socket_1c nursery_1a_cpe_no_1 nursery_1a_cpe_no_2 nursery_1b_cpe_no_3 nursery_1b_cpe_no_4 nursery_1c_cpe_no_5 nursery_1c_cpe_no_6 nursery_ac_socket_2a_no_1 nursery_ac_socket_2a_no_2 nursery_ac_socket_2b nursery_ac_socket_2c nursery_2a_cpe_no_7 nursery_2a_cpe_no_8 nursery_2b_cpe_no_9 nursery_2b_cpe_no_10 nursery_2c_cpe_no_11 nursery_2c_cpe_no_12
- Device Groups (7)
  - ursery1_lights; nursery1_sockets, nursery2_lights, nursery2_sockets, playground_lights, playground_sockets

Data Cleaning and Aggregation
Missing Data

- Power consumption and system state values are recorded at every 15s intervals from the Meshpower Telnet interface on the Microgrid.
- Missing values in the live data can happen 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 40%
- 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 component 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 15min means of the raw data and empty fields where data I not available. This is used for system logging and api 30 days display information.

Conclusions and Data handling done based on:

Kang H. The prevention and handling of the missing data. *Korean J Anesthesiol*. 2013;64(5):402-406. doi:10.4097/kjae.2013.64.5.402

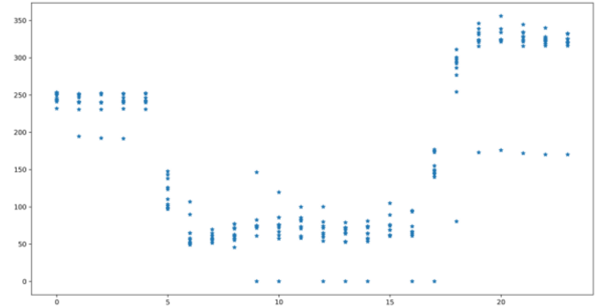
HEED

Humanitarian Engineering and Energy for Displacement

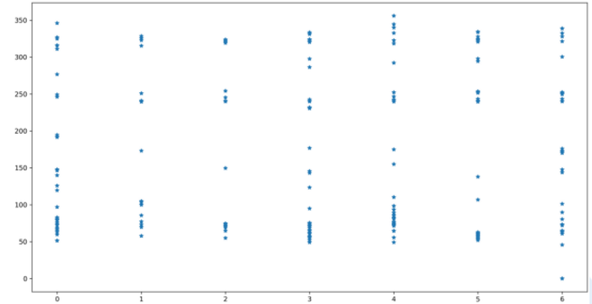
Data Cleaning and Aggregation

Consumption Forecaster:

- Requirement:
  - No missing data is allowed for the SARIMA model. Unusual peaks and outliers can also cause the forecasted data to be distorted.
  - Forecasting is done hourly so the values used are hourly means.
- Cleaning
  - For the Victron system there are no significant outliers so they require little cleaning.
  - The Device monitors sometimes have very high instantaneous power draws that are over 3kw and require some cleaning.
- Data Imputation:
  - For imputation we first identify the most significant seasonality of the data (this is also required for forecasting)



Daily Seasonality check with data plotted for each hour



Weekly Seasonality check with data plotted for each day

Data Cleaning and Aggregation

Consumption Forecaster:

- Seasonality:
  - For imputation we first identify the most significant seasonality of the data (this is also required for forecasting)
  - From the tests to the right we can see that the hourly seasonality is the most reasonable.
  - This seasonality can also be found in most the monitored devices exception being ones that are broken for the tested period or see 0 use. (ea. Nur 1C S1; Nur 1C; Nur 1C CPE6)
- Data Imputation:
  - Model based imputation requires already available data which in the case of a recovering system or a freshly started one is not guaranteed (SARIMA can require 1-6 days previous data with a 24h seasonality). To solve this and decrease the computational requirements, a two stage imputation is used.
    - Stage1: Most recent hourly mean for the period is imputed.
    - Stage2: 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 system's data quality gets better the quality of the forecasting data also improves.

Covariance

	AC_consumption_W	hour	day_wk
AC_consumption_W	11840.155820	195.216890	-10.756678
hour	195.216890	48.374279	0.089533
day_wk	-10.756678	0.089533	4.205017

Correlation

	AC_consumption_W	hour	day_wk
AC_consumption_W	1.000000	0.261687	-0.047536
hour	0.261687	1.000000	0.006278
day_wk	-0.047536	0.006278	1.000000

P_Values

	AC_consumption_W	hour	day_wk
AC_consumption_W	0	0	0.4645
hour	0	0	0.9425
day_wk	0.4645	0.9425	0

Covariance, Correlation and P_values for daily and weekly seasonality

HEED

Humanitarian Engineering and Energy for Displacement

Data Cleaning and Aggregation

Controller

- Requirements:
  - Current state of the system needs to be as accurate as possible.
  - Forecasting is required
  - If Victron state data or forecasting is not available the controller will not run
- The current state data for system consumption,generation and SoC is read from the VictronGX local telnet interface.
- For the device information, both the LVL or current Power values and the Session consumption values are used. The current power values are used for forecasting and the session values are used for control as these are the ones that allow limits to be set.
- No values are inputed for the controller. If not value are available or being read then the device is not controlled.
- The system load is used as a means of considering and forecasting any loads from which there is no info.

Parameters and Calculations Testing Environment

- 15 minute cleaned Average for Power consumption
  - Nursery1_Lights, Nursery1_Sockets, Nursery2_Lights, Nursery2_Sockets, Playground_Lights, Playground_Sockets, Streetlights
- System Parameters
  - Consumed_Energy, Generated_Energy, Battery_SoC, System_Load
- Quotas
  - Nursery1_Lights, Nursery1_Sockets, Nursery2_Lights, Nursery2_Sockets, Playground_Lights, Playground_Sockets, Streetlights
- Priorities
  - Nursery1_Lights, Nursery1_Sockets, Nursery2_Lights, Nursery2_Sockets, Playground_Lights, Playground_Sockets, Streetlights

HEED

Humanitarian Engineering and Energy for Displacement

Energy Forecasting - Consumers

- Basing it on this paper:
  - https://www.sciencedirect.com/science/article/pii/S1876610219310008
- LSTM forecasting has proven to be more accurate than ARIMA SARIMA and ARMAX
- LSTM requires more resources to run and might be an issue for Raspbery pi systems.
- Running multiple models has even higher requirements especially when it comes to re-training the data
- ARIMA (**Auto Regressive Integrated Moving Average(ARIMA)** was a close second and is more lightweight
  - Testing and evaluating it based on below:
    - https://medium.com/auquan/time-series-analysis-for-finance-arma-models-21695e14c999

Energy Forecasting – PV Panels

- Using the basis of this paper:
  - https://www.mdpi.com/1996-1073/12/9/1621
- Using just the Clear sky model for now
- SOLIS Clear sky model
  - https://www.sciencedirect.com/science/article/pii/S0038092X08000406
  - Using simplified version available in python
  - We calculate the GHI,DNI and DHI based on the *ineichen* model
- Using parameters to get the irradiance on the panel
  - Loutzenhiser P.G. et. al. "Empirical validation of models to compute solar irradiance on inclined surfaces for building energy simulation" 2007, Solar Energy vol. 81. pp. 254-267
  - Calculating the total irradiance based on GHI,DNI, DHI, solar_zenith, solar_azimuth surface_tilt and surface_azimuth. Based on [2] Perez, R., Seals, R., Ineichen, P., Stewart, R., Menicucci, D., 1987. A new simplified version of the Perez diffuse irradiance model for tilted surfaces. Solar Energy 39 (3), 221–232.
- Irradiance to power:
  - https://www.sku.ac.ir/Datafiles/BookLibrary/45/John%20A.%20Duffie,%20William%20A.%20Beckman(auth.)-Solar%20Engineering%20of%20Thermal%20Processes,%20Fourth%20Edition%20(2013).pdf
  - Power= irradiance *PanelArea*efficiency
    - Area = 21.3 ?
    - Efficiency = 15.5
- Using 60% of the value as a conservative estimate in case of clouds

Battery Model

- Using the basis of this paper:
  - https://ris.utwente.nl/ws/files/28201914/A_comprehensive_model_for_battery_State_of_Charge_prediction.pdf
- No considering battery deterioration
- Not considering battery fade as for the 24h time periods we use it is relatively small
- Estimating the state of charge as the available Stored Energy of the battery/ maximum stored energy of the battery
  - Charging efficiency set to 80% as based on the below article as a mean value, although at the 80-100% SOC's we are working at it should be 50%
    - https://ieeexplore.ieee.org/abstract/document/564417
  - $\Delta St = Ce * \Delta Ct - \Delta Dt$ – where St is Stored Energy; Ct is Charged Energy; Dt is Discharged energy and Ce is Charge Efficiency.

HEED

Humanitarian Engineering and Energy for Displacement

Control Algorithm–Protection

- Currently on the Microgrid
- When the SoC has reached a value that is lower than a set point (in this case 40%) every device on the system is turned off to protect the battery. Once SoC values go above the 40% mark devices are turned back on.
- There are no priorities or orders, it’s all on or all off controller designed to protect the battery.
- The current version of this controller on the Microgrid system

Control Algorithm–Space Shared

- Algorithm based on Space shared Resource allocation
- At every 15 minutes the algorithm runs a set of calculations based on forecasted and measured data:
  - It Calculates the available Energy in the system:
    - E_Generated (Until 6Am reset) = SUM(PV_Generation(hourly)), where the values are after the current Timestamp
    - E_avail = (Battery_SoC-50.0)/100.0*21.1+E_Generated(Until 6Am reset)
  - Based on the Forecasting for each device summed by groups the total consumption of the system is calculated.
  - If the total consumption of the system is higher than the Energy Available Quotas need to be in imposed
    - Device groups are taken by their priority levels and quotas are imposed based on energy availability:
      - Example. If total *E_avail* is 7kw but the devices and system will use a total of 7.4 and the lowest priority device will use 1.4kw than it will receive a limit of 1kw consumption until reset.
    - This group_limit is then distributed by share to each device
    - 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 the Wh available
- Considering inaccuracies in measurement and forecasting a 15% over-estimation is considered on each device

HEED

Humanitarian Engineering and Energy for Displacement

Control Algorithm-Space Shared

- Considering the forecasted Data from the table below and an analysed period of two hours. With a 2.1kw battery

	timestamp	Nursery1_Lights	Streetlights	Consumed_Energy	Generated_Energy	Battery_SoC	System_Load	Nursery1_Lights_Priority	Streetlight_priority
Measured	19/08/2019 06:00	14	5	69	0	52.5	50	1	2
Forecasted	19/08/2019 07:00	7	14	72	21	50.08	51	1	2
	19/08/2019 08:00	9	9	66	48	50	48	1	2

Available Energy = (Battery_SoC-50.0)/100.0*2100+E_Generated = (52.5-50) /100*2100+(21+48)=121.5
Consumed Energy = 72+66 = 138Wh
Required Energy Difference = Available Energy - Consumed Energy = -16.5Wh
There is a deficiency so quotas need to be impose, starting from the device with the least priority:

Streetlight – Total Consumption: 23Wh; Deficit = 16.5Wh, so the total consumption needs to be capped to 6.5Wh or a quota imposed of 0.28.

This quota is the distributed equally to all the streetlights so ideally they turn off at the same time once hitting the quoat. Streetlight Remains the samev

timestamp	Nursery1_Lights	Streetlights	Consumed_Energy	Generated_Energy	Battery_SoC	System_Load	Nursery1_Lights_Priority	Streetlight_priority	Nursery1_Lights_Quota	Streetlight_Quota
19/08/2019 07:00	7	6.5	64.5	21	50.5	51	1	2	1	0.45
19/08/2019 08:00	9	0	59	48	50	48	1	2	1	0.0

The quotas are imposed as usage limits, so the streetlights will have 0.45 quota for the first hour and a quota of 0 for the second

HEED

Humanitarian Engineering and Energy for Displacement

Control Algorithm–Scheduling GA

- With the GA method the quota can be of 3 types
  - Binary – A device is either on or off in an hour
  - Percentage – Of the total energy available to be consumed only a certain percentage is allowed
  - Limits – kWh limits are imposed for each hour irrespective of the actual consumption
- The quotas are hourly per device group
- Each individual consists of 168 genes that are
  - Binary: either 0 or 1 (2 possible values)
  - Percentage: Values between 0.0 and 1.0 with 2 decimal step (20 possible)
  - Limits: in Wh with a step of 50wh

Version 2:

- Random Select of the top 20% of the population for mutate and crossover
- 5% elitism; 25% crossover; 25% mutate; 25% crossover and mutate; 20% new random individuals in each generation.
- Mutation is at 5% with each gene having a change of getting a new value; increment/decrement

Version 1:

- Roulette wheel selection of Individuals
- 5% Elitism;20% mutation of each generation
- Crossover and Mutation on all population

HEED

Humanitarian Engineering and Energy for Displacement

Control Algorithm–Scheduling GA

- Considering a optimisation scenario for 3 hours and two devices the gene looks as in the table to the right. It is applied to the original data to generate the one in the table below it.

	Nursery1_Lights	Streetlights
Hour 1	0.4	0.25
Hour 2	0.8	1.0

Original Table

timestamp	Nursery1_Lights	Streetlights	Consumed_Energy	Generated_Energy	Battery_SoC	System_Load	Nursery1_Lights_Priority	Streetlight_Priority	Nursery1_Lights_Quota	Streetlight_Quota
19/08/2019 07:00	7	14	72	21	51.21	51	1	2	1	0.45
19/08/2019 08:00	9	9	66	48	50	48	1	2	1	0.0

Resulting Table

timestamp	Nursery1_Lights	Streetlights	Consumed_Energy	Generated_Energy	Battery_SoC	System_Load	Nursery1_Lights_Priority	Streetlight_Priority	Nursery1_Lights_Quota	Streetlight_Quota
19/08/2019 07:00	2.8	3.5	57.3	21	51.78	51	1	2	0.4	0.25
19/08/2019 08:00	7.2	9	64.2	48	51.01	48	1	2	0.8	1.0

Fitness Function – Battery Health

The health of the battery requires consideration when optimizing microgrid systems. Maximizing the State of Charge and reducing the amount of draw throughout the runtime of the system is a practical surrogate to consider in cases where charge cycles are not available or are not practical for the timeframe. As we have two parameters they are in the range [0,2].

$$F_B(t) = \frac{SoC(t) - SoC_{min}}{SoC_{max} - SoC_{min}} + \frac{P_i(t)}{P_{rated}}$$

Where $F_B(t)$ is the Battery Factor for time(t), noramlised to the range [0,1] using the SoC_{max} value of 100.0% And SoC_{min} of 40%. The $P_i(t)$ is the total energy consumption – generation P_{rated} is set to 3,500W.

Fitness Function – Efficiency

Within the available parameters and system data there are two efficiencies that need to be considered. How much of the total energy use is useful load and not system loads and how much of the energy flow is consumption as a ratio. The second one is a way of ensuring that parameter is within the [0,1] bounds. As we have two parameters they are in the range [0,2].

$$F_E(t) = \frac{E_{Consumed}(t) - E_{SystemLoad}(t)}{E_{Consumed}(t)} + \frac{E_{Consumed}(t)}{E_{Consumed}(t) + E_{Generated}(t)}$$

Where $E_{Consumed}(t)$ is the total energy consumed by the system. $E_{SystemLoad}(t)$ is the Energy use of auxiliary systems that the Device montors don't capture. $E_{Generated}(t)$ is the amount of energy generated by the solar panels.

Fitness Function – Penalty factor

As there is a protection on the battery that cuts off any power consumers, reaching the cutoff values incurs a penalty on the system.

$$F_P(t) = \begin{cases} \sum_{t \in T} 0\,, SoC(t) \leq 40\% \\ \sum_{t \in T} 1\,, \qquad otherwise \end{cases}$$

HEED

Humanitarian Engineering and Energy for Displacement

Fitness Function – Dissatisfaction factor

A dissatisfaction degree is also introduced as at time slots of low SoC or not enough energy available, the user will not be able to use all desired appliances. Thus, this factor is modelled according to the priority of the appliance and its energy quota, as follows

$$F_D(t) = \frac{\sum_{a \in A} Quota_a(t) \times \frac{1}{P_a}}{\sum_{a \in A} 1}$$

where $F_D(t)$ represents the lack of energy available for the appliance a, $Quota$ is the amount of energy available to be used for the appliance and P_a is the priority of that appliance. This is divided by the total number of appliances to get the values in the region [0,1]

HEED

Humanitarian Engineering and Energy for Displacement

Fitness Function – Sustainability Factor

The sustainability factor evaluates how the SoC changes throughout the evaluated period. This concepts prevents any algorithm draining the battery for current time period gains that would be unwanted in real life situations.

$$F_S(t) = \frac{SoC(t) - SoC(t-1)}{SoC(t)}$$

where $SoC(t)$ is the current State of Charge for the battery and $SoC(t-1)$ is the final State of Charge for the previous time period.

Fitness Function – Complete

The utility function is a sum of all component functions applied for the time period in question.

$$\max f(x) = \sum_{t \in T} w_B \times F_B(t) + w_E \times F_E(t) + w_P \times F_P(t) + w_D \times F_D(t) + w_S \times F_S(t)$$

This allows for the standard function to be used regardless of the timeframe and each component to be allocated a weights. This also means that we can identify timeperiods where the utility is especially low and tweek any control algorithm for those scenarios.

HEED

Humanitarian Engineering and Energy for Displacement

Fitness Function – Application

The fitness of a system is calculated at every hour. It is applied to the data below as shown.

The weights used for w1;w2 and w3 are 0.3.

timestamp	Nursery1_Lights	Streetlights	Consumed_Energy	Generated_Energy	Battery_SoC	System_Load	Nursery1_Lights_Priority	Streetlight_Priority	Nursery1_Lights_Quota	Streetlight_Quota	Utility
19/08/2019 07:00	2.8	3.5	57.3	21	51.78	51	1	2	0.4	0.25	0.0
19/08/2019 08:00	7.2	9	64.2	48	51.01	48	1	2	0.8	1.0	0.0

Utility for hour 7:00 – F_B(t) =(51.78-40)/60.0+(57.3-21)/3000 = 0.200
F_E(t) =(57.3-51)/57.3+57.3/(57.3+21) = 0.841
F_P(t) = 0
F_D(t) =(0.4*1/1+0.25*1/2)/2 = 0.256
$Util$(19/08/2019 07:00) $f(x)$ = 1.553

Utility for hour 8:00 – F_B(t) =(51.01-40)/60.0 +(64.2-48)/3000 = 0.189
F_E(t) =(64.2-48)/64.2+64.2/(64.2+48) = 0.824
F_P(t) = 0
F_D(t) =(0.8*1/1+1.0*1/2)/2 = 0.65
$Util$(19/08/2019 07:00) $f(x)$ = 1.663

Simulation Environment V1

- Recreates the Microgrid in a testing environment based on historic data to be used in GA based optimisation
- A python script served pre-processed historic data and enacted control parameters.
- Matlab was used to run a Genetic Algorithm based optimisation

Issues:

- Using system commands to run python scripts and sharing data through file meant that it took 0.81 seconds to evaluate a single control parameter and 8 hour to get the result in Fig. SE-V1

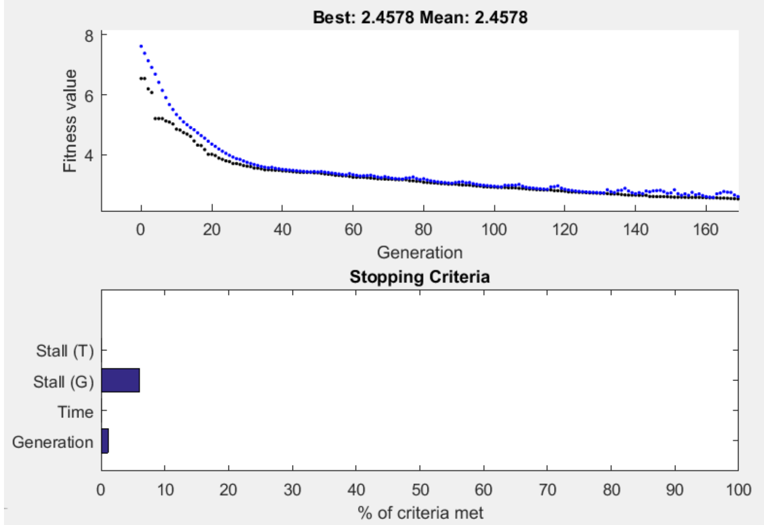


Figure SE-V1 – GA Run of optimising Depth of Discharge using the Simulation Environment V1

Simulation Environment V2

- Allows modifications to be made to the original microgrid environment by scaling the batteries, pv panels an possibly varying consumers.
- Fully coded in python to increase processing speed (0.12s for an evaluation as opposed to 0.84 (SSD) and 2.4(HDD))
- Modified GA (Version 2) to suit the peculiarities of the problem
- 4 controller implemented and tested
- Utility function can be modified for all using a single method

HEED

Humanitarian Engineering and Energy for Displacement

Testing and Evaluation

- Testing of functionality and business logic of components done based on the testing plan put together with Scene
- The Control Algorithm and GA as well as future components are evaluated using a defined set of plots that showcase how various utility function and system parameters changed/behaved during operation.
- Health and Status monitoring dashboard/Digital Twin in Coventry with Alarms, weekly status reports possible

Testing and Evaluation

Component State and Testing

Component	Implemented	Deployed	Results Validation	UnitTests	Graphical Evaluation
Local Data Logger	X	X[1]	X		X
Web Data Logger	X	X	X		X
Data Aggregator	X	X	X	X[2]	
Forecaster (Dev;PV)	X	X	X	X	X
Control Algorithm	X	X	X	X[3]	X
Control Enactor	X	X	X	X	
System Logger	X	X	X	X	
Front End	X	X	X		X
Full System Test	X	X[4]	X		X

[1] Deployed in testing environment as we had no access to the physical microgrid to conduct this test.
[2] UnitTests encapsulated in Data Retreiver and Forecasting tests
[3] Also includes tests for the Battery
[4] Full system was deployed for 3 weeks using the Web Data Logger to get close to real-time data from the meshpower portal.

HEED

Humanitarian Engineering and Energy for Displacement

Testing and Evaluation

Component State and Testing

- Example UnitTestRun Output:

```
HOME>make test
Running Tests...
python  -m unittest -v -b  tests/test_Forecaster_v2.py  tests/test_Data_Retreiver.py  tests/test_Data_Aggregator.py
tests/test_Controller.py  tests/test_Remote_Logger.py  tests/test_Control_Enactor.p
y  tests/test_Battery.py  tests/test_Forecaster.py
test_ForecastV2 (tests.test_Forecaster_v2.ForecastV2_Test) ... ok
test_componentResponse (tests.test_Data_Retreiver.DataRetreive_Test) ... ok
test_components_noResp (tests.test_Data_Retreiver.DataRetreive_Test) ... ok
test_Aggregator (tests.test_Data_Aggregator.Aggregator_Test) ... ok
test_Controller (tests.test_Controller.Controller_Test) ... ok
test_RemoteLogger (tests.test_Remote_Logger.RemoteLogger_Test) ... ok
test_check_diff_light (tests.test_Control_Enactor.Enactor_Test) ... ok
test_check_diff_socket (tests.test_Control_Enactor.Enactor_Test) ... ok
test_discharge (tests.test_Battery.BatteryTest) ... ok
test_Forecast (tests.test_Forecaster.Forecast_Test) ... ok

----------------------------------------------------------------------
Ran 10 tests in 181.355s

OK
```

HEED

Humanitarian Engineering and Energy for Displacement

Testing and Evaluation
Consumer and Load Forecasting

- Based on a Seasonal ARIMA Model
- Using Historic Hourly Data from 05-18 August 2019.
- Model fitness is evaluated using the AIC (The **Akaike information criterion** 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 w best orders of (1,1,2)x(0,2,2) and a seasonality of 24 hours. 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 24 hours. This resulted in an AIC of 6161.69
- The results from the figure show a model with orders of (1,1,1)x(1,1,1) and a seasonality of 24 hours

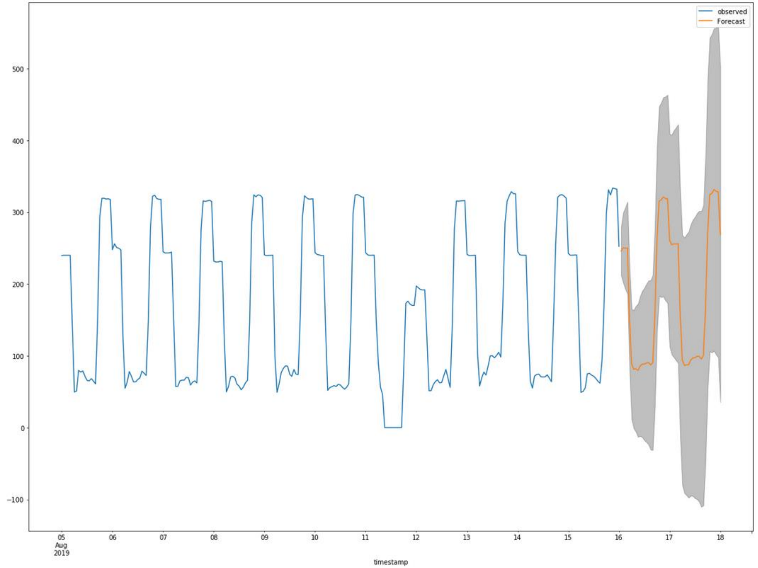


Figure - Forecasting for AC Consumption
Blue line is the measured AC consumption. The orange line is the mean predicted
And the gray is the upper and lower limits of what is predicted.

HEED
Humanitarian Engineering and Energy for Displacement

Testing and Evaluation
PV Generation Forecasting

- Based on a ClearSky Model and the pvlib python library.
- Conducted based on the previous theoretical slide.
- The use of ground and sky diffuse irradiance improved the accuracy of the prediction having the predicted consumption be in a similar shape as the generation
- There is a big difference between the predicted and actual as this is based of a model that doesn't consider weather factors.
- The Solcast values are also shown as a comparison to the Clearsky forecast.

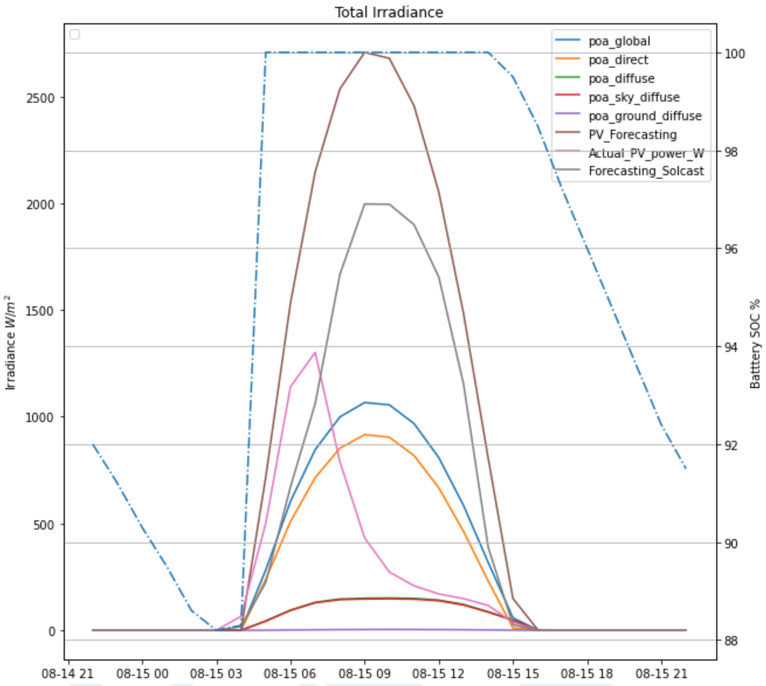


Figure - Forecasting for PV Generation
Contains all aspects of irradiance that affect he panel as well as historic predictions and the actual power generated in that time period

HEED
Humanitarian Engineering and Energy for Displacement

Testing and Evaluation

PV Generation Forecasting

- For the deployment, the predicted value is scaled by 0.545 to account for cloudy weather that the Clearsky model cannot account for.
- The drop-off in generation around 9am can be explained by the saturation of the battery (SoC=100%) that can be seen in the previous figure.
- Because the forecast doesn't account for this saturation, we get a more accurate forecast of the available energy as opposed to using a model like SARIMA.

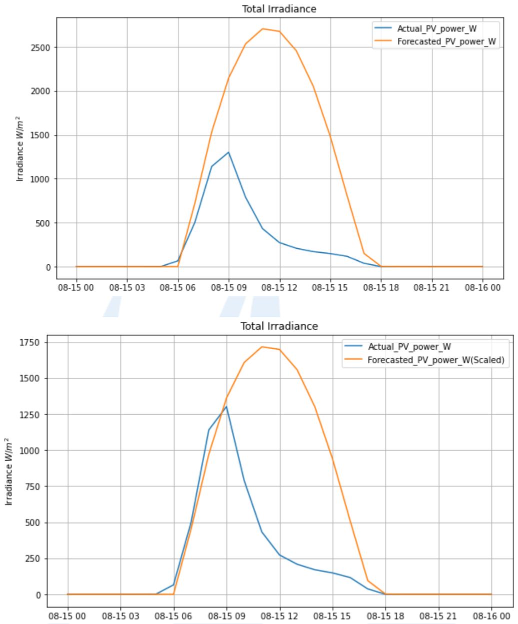


Figure PV2.
(a) Forecasting without clearness index
(b) Forecasting with added clearness index of 0.545

Testing and Evaluation
Space Shared Controller

- The Space Shared Controller and its difference compared to the protection controller is evaluated in a scenario where there is not enough energy to satisfy the needs of the community. The scenario was generated based on real data from august where we reduce the size of the PV panels and the battery and maintain consumption levels. The original day can be seen in Fig. 1 and the scaled system in Fig. 2.
- We can see that while the SoC doesn't drop below 85% in the original scenario, the scaled scenario sees it bottom out at 40% which means the protection controller would have kicked in. and system curtailment is needed to protect the battery.
- This was achieved by scaling the battery to 4.1kw instead of the 21.1kw it currently has and by scaling the pv system to 40% of it's capacity.
- It can also be noted that while it the first system the pv panels fully charge the battery they don't manage to do this in the second one.

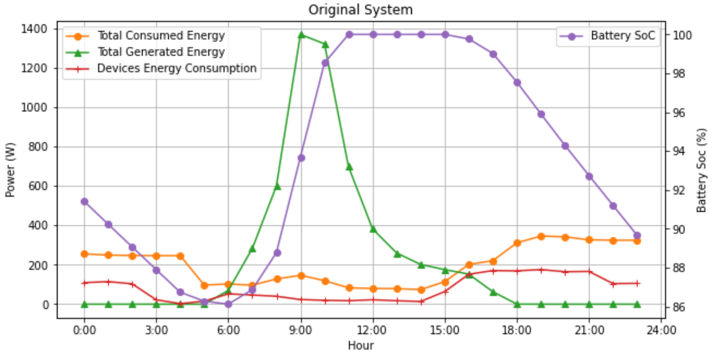


Figure 1

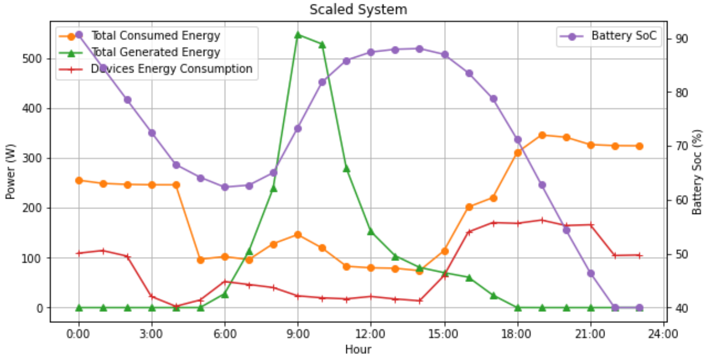


Figure 2

HEED
Humanitarian Engineering and Energy for Displacement

Testing and Evaluation
Space Shared Controller

- The Space Shared Controller utilizes the generation and consumption forecasting to estimate the energy deficit or surplus at the end of its period and create curtailment on devices based on this.
- The difference between this method and the currently deployed standard protection controller that turns off all devices when there is not enough power left can be seen in Fig1 and Fig2.
- The Space Shared Controller in Fig. 1 limits devices starting from the lowest priority one throughout the whole period. The protection controller in Fig2. only limits devices once there is not enough power left, hence all of the devices being turned off in the last two hours.

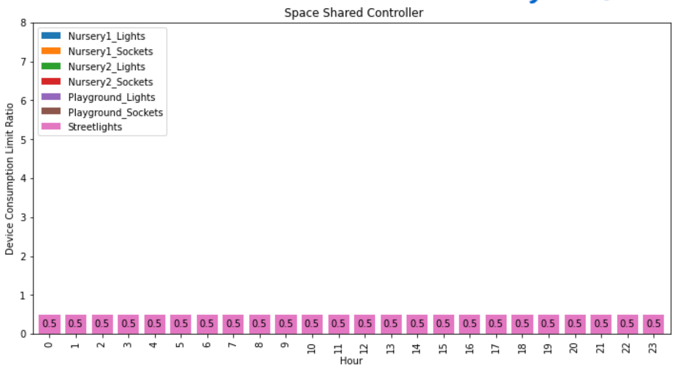


Figure 1

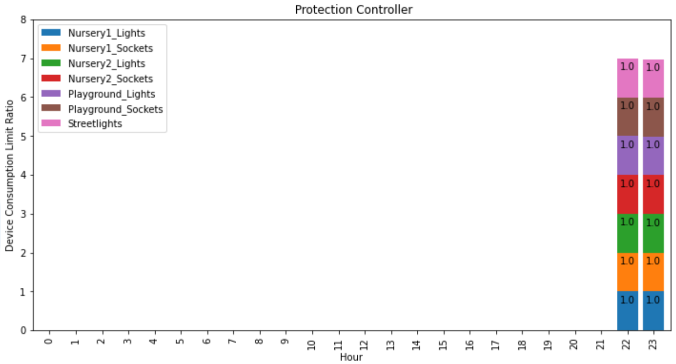


Figure 2

HEED
Humanitarian Engineering and Energy for Displacement

Testing and Evaluation
Space Shared Controller

- The influence of the algorithms on the system can be seen in Fig. 1 for the Space Shared controller and in Fig2. for the Protection Controller. The original (without control or protection) values for system and device consumption are plotted together with the battery SoC Changes.
- The change in the battery SoC can be seen between the two where in the Space shared on it only reaches the limit by the end of the period while for the protection one needs to curtail the devices for the final two hours.
- In the Space Shared one it is visible that the device power consumption is lower throughout the period as a single device is curtailed while for the protection on it only drops at the end by a significant value when there is no more energy available and the battery is depleted.

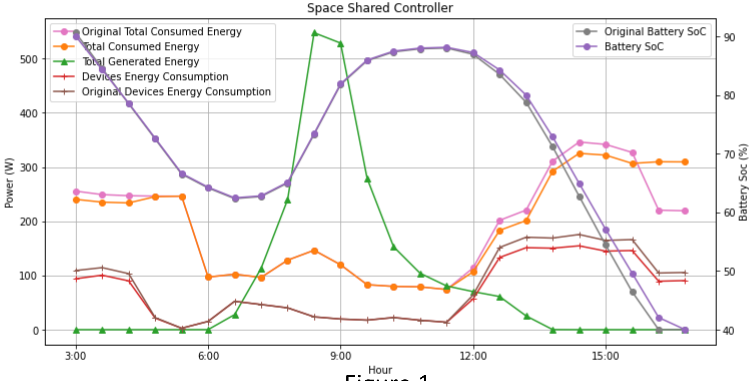


Figure 1

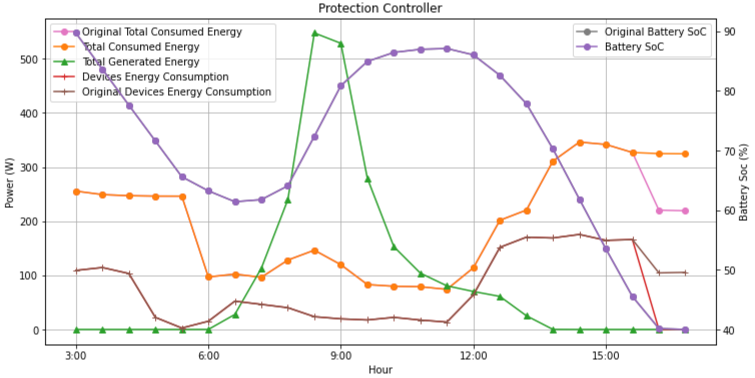


Figure 2

HEED
Humanitarian Engineering and Energy for Displacement

Testing and Evaluation

GA implementation on Testing Env

- The runtime slope in Figure 4 shows that improvements made to GA allowed it to converge faster and have a much more incremental slope rather than random hops.
- The example runs are for a simple SoC based utility to validate the behavior of the GA. In this case the best option would be to turn off all the devices where the consumption is larger than the generation. This can be seen in the Energy CoDiffrence (Figure 1) as regions with values larger than 0.
- The results in the graph show how the various controls affected the SoC of the battery (Figure 3). And total energy consumed by devices can be seen in(Figure 2).
- The difference is visible between the two GA solutions and the original deployment without any control enabled.

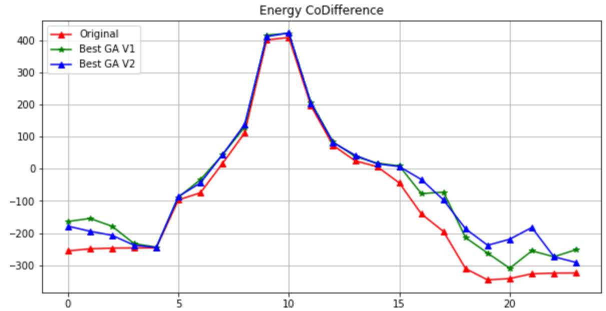


Figure 1

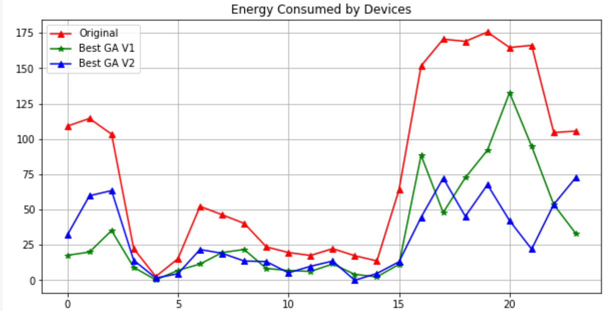


Figure 2

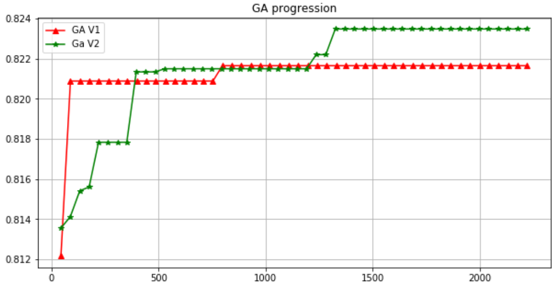


Figure 4

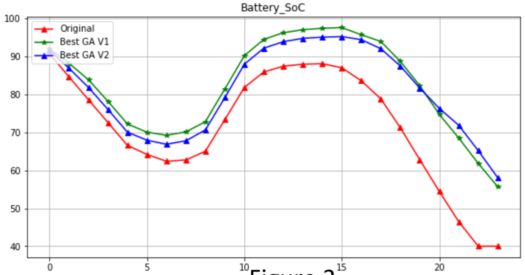
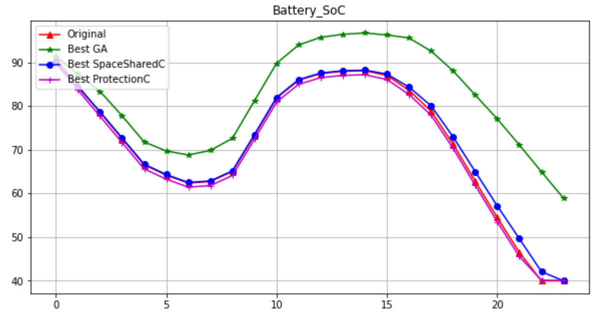


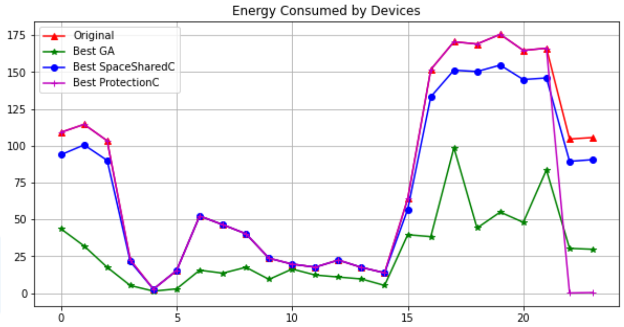
Figure 3

Testing and Evaluation

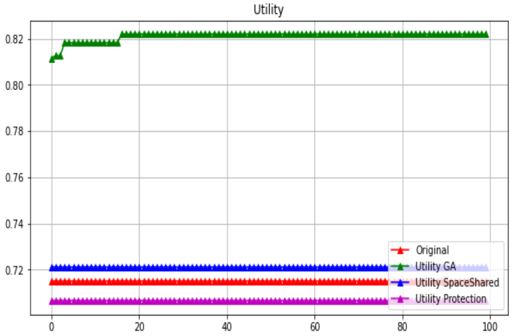
GA V2 comparison with Existing Controllers – SoC Scenario

- GA is compared with the built in protection and devised space-shared methods.
- The results show how they behave for the simple SOC scenario from above.
- The test is run with a modified scenario that has a max battery power of 4.1 kw and PV panels 40% the size of the current ones. This is done to trigger the safety and show the behavior of the methods.
- The GA has a lot better results but the previous controllers reach solutions in 0.15 seconds rather than hundreds or thousands of seconds.





- We can see that GA curtails devices across the range, while the protection turns all devices off for the final two hours. The Space shared method only limits on device to the amount needed to not trigger the protection for the time period.



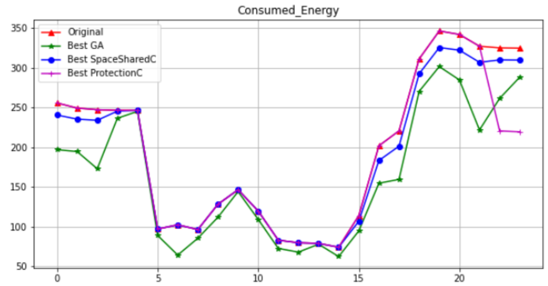
HEED

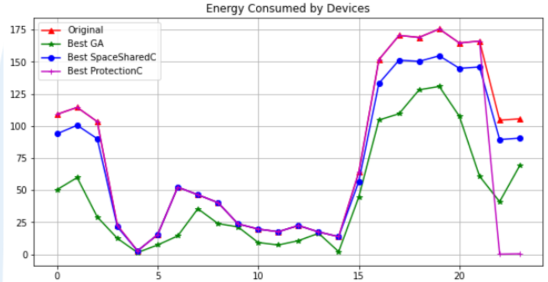
Humanitarian Engineering and Energy for Displacement

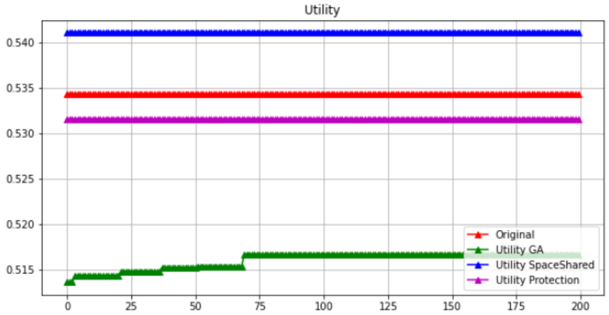
Testing and Evaluation

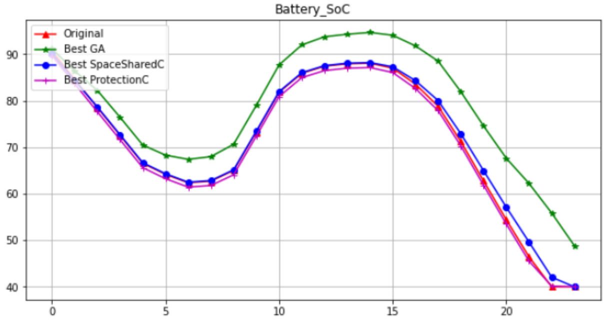
GA V2 comparison with Existing Controllers – Full Utility Scenario

- Uses the full spectrum of the utility function, considering all parameters.
- Gives equal weights of 0.25 to all parameters and 0.125 to parameters that are in range [0,2]
- Here we can see that the GA is Less likely to curtail devices as it has utility penalties if it does.









Conclusions and Outputs

Research Centre Manufacturing and Materials Engineering | Coventry University

- The physical deployment aims to demonstrate the functioning of each component and test how a stable and easy to understand controller would work.
- The work done by Leonardo aims to identify a utility function for the microcontroller
- The testing environment will give us the graphs and metrics to evaluate each deployment and controller
- Optimisation approaches are compared to show drawbacks and benefits from their implementation
- The GA based solutions might provide a good generic solver but due to the time and processing power required to find a solution deterministic solvers might be better solutions.
- Defining the utility function and creating a way of creating and evaluating individuals is key in creating any control method.

HEED
Humanitarian Engineering and Energy for Displacement