Optimization 1: Super flexible halfhour optimization

*This document is based on Optimization\_vers5*

**Load information**

Uncontrollable loads are simply vectors containing 48 values that represent energy demand per halfhour

Controllable loads are stored as an Appliance object:

Appliance

X0 LB UB A b

On creation of each object, all those parameters are provided to the constructor

Appliance objects created: battery, pump and geyser.

Battery:

LB – 48 vector. Each element is the maximim amount of energy it can deliver to the system per halfhour

UB – LB – 48 vector. Each element is the maximum amount of energy it can charge per halfhour

X0 – 48 vector. Set as 500 per element (violates constraints from the start)

A, b – Generated by the BatteryInequalityGenerator function. Ensures battery does not charge or discharge beyond a specified level during operation. A contains 48 columns and 96 rows, while b contains 1 columns and 96 rows

Pump and Geyser:

LB – 48 vector. Lowest amount of energy it can use per halfhour is 0.

UB – 48 vector. Highest amount of energy it can use per element is demand\*halfhour

A, b – Ensures that the appliance uses a specified amount of energy per day

X0 – Simply set as 148 per element

**Constraints**

The LB, UB, A, b X0 is then combined in the CombineConfigAppl function, which accepts a variable amount of Appliance objects.

An Aeq and beq constraint is created to ensure that the battery has netto energy gain of 0 per day.

**Plotting**

For the uncontrollable appliances – constantplot

For the configurable appliances – ApplPlot (plots an extract from the optimset.x vector)

The sum of all configurable appliances – TotalControllable

The grid – gridplot

The TOU – constantplot

**Cost function**

The cost function minimizesd cost to the owner when gridusage is multiplied with TOU

Optimization 2: Scheduling fixed load cycles

*This document is based on Optimering\_vers6/Metode3*

**Load Information**

Uncontrollable appliances are stored as Uncontr\_Appl objects

Uncontr\_Appl

profile An 48 vector where each element contains the energy use per halfhour

Controllable appliances are stored as Contr\_Appl

Contr\_Appl

LB UB A b X0 d demand

LB elements are set as 0

UB is set as the demand\*halfhour

A and b ensures that the cycle times are between the 0 and 24 hours and that it doesn't overlap

To create an object, the X0, duration and demand is given.

Battery is created in the Battery class

Battery

LB UB A b X0

LB is the maximum energy the battery can provide per halfhour

UB is the maximum energy the battery can charge per halfhour

A, b is the integrating function to ensure battery doesnt over- or undercharge

**Constraints**

The CombineConfigAppl combines the battery and controllable appliances constraints

LB – 48 elements from battery plus foreach\_appliance(appliance\_amount\_of\_cycles)

UB – 48 elements from battery plus foreach\_appliance(appliance\_amount\_of\_cycles)

A – 48 columns and 96 rows for battery and adds for each appliance

b – rows correspond to A's rows

X0 – simply adds the X0's together

**Plotting**

Uncontrollable appliances – constantplot

Controllable appliances – cntrlApplPlot (index is provided)

Battery – profilePlot

Grid – gridplot

**Cost function**

The cost function minimizes cost to the owner when gridusage is multiplied with TOU

Optimization 3: (I) First find the optimal profile (II) Fit the appliances as close as possible

This document is based on Optimering\_vers6/Metode\_EersOptimale

**Load Information**

Uncontrollable appliances are stored as Uncontr\_Appl objects

Uncontr\_Appl

profile An 48 vector where each element contains the energy use per halfhour

Controllable appliances are stored as Contr\_Appl

Contr\_Appl

LB UB A b X0 d demand

LB elements are set as 0

UB is set as the demand\*halfhour

A and b ensures that the cycle times are between the 0 and 24 hours and that it doesn't overlap

To create an object, the X0, duration and demand is given.

Battery is created in the Battery class

Battery

LB UB A b X0

LB is the maximum energy the battery can provide per halfhour

UB is the maximum energy the battery can charge per halfhour

A, b is the integrating function to ensure battery doesnt over- or undercharge

**First optimization**

A 48 element vector must be found that optimizes the load distribution for the amount of energy that the loads requires to run. The battery allows for charging but as much as is charged must again be discharged aswell.

1. **Constraints**

LB – The limit for the amount that the battery can provide to the system in half an hour.

UB – The sum of the energy levels that the loads require + max of battery

Aeq, beq – The sum of the profile must add to the amount of energy that the profiles require to run

A, b – Empty

X0 – Initialized to zero

1. **Plotting**

Uncontrollable loads – Constantplot

Controllable vector -

1. **Cost function**

The cost function will optimize for minimum cost when grid usage is multiplied with TOU

**Second optimization**

1. **Constraints**
2. **Plotting**