

## Validierung UC 3 Feldtestsimulation - NEW 4.0 Feldtest

In der folgenden Validierung werden die Simulationsergebnisse (Feldtestwoche von Mo, 18.11.2019 bis So, 24.11.2019) mit den Daten der Netzbetreiber abgeglichen. Zugrundegelegt ist ein Simulationdurchlauf ohne Smart Balancing.

### Vergleich Simulation mit Netzdaten

Um den Vergleich durchzuführen, werden die Datensätze geladen und vorab mit Zeitstempeln versehen.

#### Import and load data

```
In [1]: import pandas as pd
import matplotlib.pyplot as plt
```

```
In [3]: # Lade Daten von Feldtest Simulation
Feldt = pd.read_csv("20200330Feldtest_sim_output_period_ARGE.csv",sep=';')
Feldt_sec = pd.read_csv("20200330_Feldtest_sim_output_all_ARGE.csv",sep=';')
```

```
In [2]: # Lade historische Daten Deutschland 2015 bis 2019
GER = pd.read_csv("GER.csv",parse_dates=True,index_col='Timestamp')
```

```
In [4]: #Ausgleichsenergiepreis (AEP) bzw. REBAP nicht in df GER enthalten, zusätzlich
        laden:
AEP1911 = pd.read_csv("REBAP_201911.csv",header=3,sep=';',thousands='.',decimal
='.',usecols=['QUAL.'])
AEP1911.index = pd.date_range(start='00:00 11.01.2019', end = '23:59 11.30.2019
',freq='15 min')
AEP1911_week = AEP1911['00:00 18.11.2019':'23:30 24.11.2019']
```

#### give time index and load historic data

```
In [5]: #data 2s to short
Feldt_sec.index = pd.date_range(start='00:00:00 18.11.2019', end = '23:59:58 2
4.11.2019',freq='S')
```

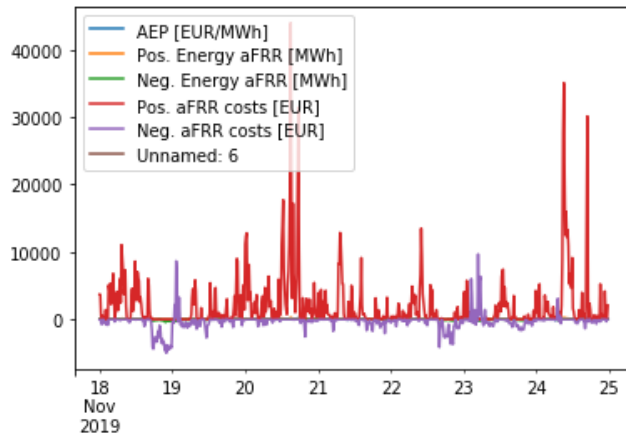
```
In [6]: #data 30 min to short
Feldt.index = pd.date_range(start='00:00 18.11.2019', end = '23:30 24.11.2019',
freq='15 min')
Historic = GER['00:00 18.11.2019':'23:30 24.11.2019']
```

#### visualisation of data "first impresion"

Die Plots veranschaulichen die output Daten der Simulation (Alle zu untersuchenden Daten stehen in der Legende).

```
In [97]: #Auflösung 15 Min
Feldt.drop("time [s]", axis=1).plot()
```

```
Out[97]: <matplotlib.axes._subplots.AxesSubplot at 0x1275af208>
```



## data analysis

Für die Validierung kommen nur die 15 Minuten Werte in Frage, da die Vergleichswerte diese Auflösung haben.

### Vergleich aktivierte Regelernergie

Die aktiverite SRL Leistung (über 15 Minuten gemittelt) korreliert sehr stark mit den Vergleichswerten (Korrelationsfaktor pow\_up 0.987 bzw. pow\_down 0.993)

```
In [77]: Vergleich_aFRR_pow = pd.DataFrame()
Vergleich_aFRR_pow["aFRR_pow_up_sim"] = Feldt["Pos. Energy aFRR [MWh]"]*4
Vergleich_aFRR_pow["aFRR_pow_up_hist"] = Historic["GER_aFRR_pow_up"]

Vergleich_aFRR_pow["aFRR_pow_down_sim"] = Feldt["Neg. Energy aFRR [MWh]"]*(-4)
Vergleich_aFRR_pow["aFRR_pow_down_hist"] = Historic["GER_aFRR_pow_down"]
```

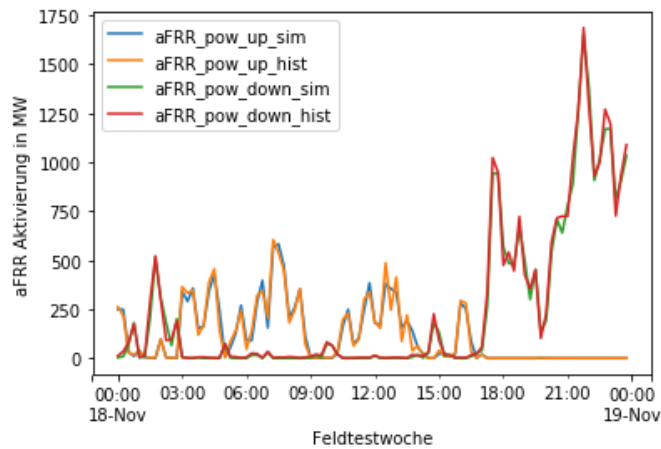
```
In [98]: Vergleich_aFRR_pow.corr()
```

```
Out[98]:
```

	aFRR_pow_up_sim	aFRR_pow_up_hist	aFRR_pow_down_sim	aFRR_pow_down_hist
aFRR_pow_up_sim	1.000000	0.987278	-0.295639	-0.298715
aFRR_pow_up_hist	0.987278	1.000000	-0.287682	-0.291869
aFRR_pow_down_sim	-0.295639	-0.287682	1.000000	0.993161
aFRR_pow_down_hist	-0.298715	-0.291869	0.993161	1.000000

```
In [100]: Vergleich_aFRR_pow["18.11.2019"].plot()
plt.ylabel('aFRR Aktivierung in MW')
plt.xlabel('Feldtestwoche')
```

```
Out[100]: Text(0.5, 0, 'Feldtestwoche')
```



### Vergleich Kosten für Regellenergie

Die Kosten für aktiverite SRL Leistung (über 15 Minuten gemittelt) korreliert sehr stark mit den Vergleichswerten (Korrelationsfaktor cost\_up 0.982 bzw. cost\_down 0.926)

```
In [ ]: Vergleich_aFRR_cost = pd.DataFrame()
Vergleich_aFRR_cost["aFRR_cost_up_sim"] = Felddt["Pos. aFRR costs [EUR]"]
Vergleich_aFRR_cost["aFRR_cost_up_hist"] = Historic["GER_aFRR_cost_up"]

Vergleich_aFRR_cost["aFRR_cost_down_sim"] = Felddt["Neg. aFRR costs [EUR]"]*(-1)
Vergleich_aFRR_cost["aFRR_cost_down_hist"] = Historic["GER_aFRR_cost_down"]
```

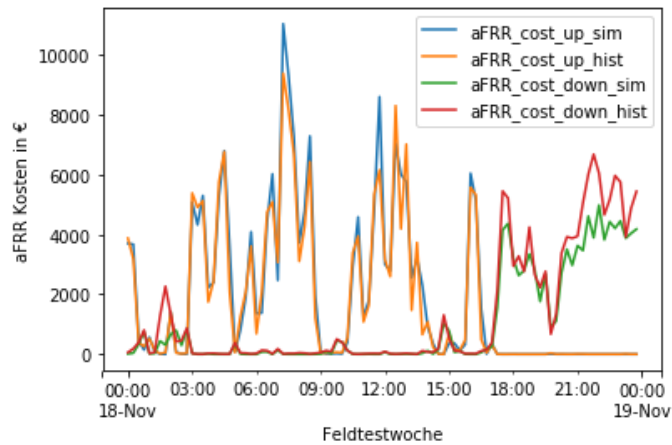
```
In [101]: Vergleich_aFRR_cost.corr()
```

```
Out[101]:
```

	aFRR_cost_up_sim	aFRR_cost_up_hist	aFRR_cost_down_sim	aFRR_cost_down_hist
aFRR_cost_up_sim	1.000000	0.981976	-0.123527	-0.176785
aFRR_cost_up_hist	0.981976	1.000000	-0.122475	-0.176235
aFRR_cost_down_sim	-0.123527	-0.122475	1.000000	0.925513
aFRR_cost_down_hist	-0.176785	-0.176235	0.925513	1.000000

```
In [88]: Vergleich_aFRR_cost["18.11.2019"].plot()
plt.ylabel('aFRR Kosten in €')
plt.xlabel('Feldtestwoche')
```

```
Out[88]: Text(0.5, 0, 'Feldtestwoche')
```



### Vergleich AEP

Der berechnete AEP aus dem Modell hat einen mittelstarken positiven linearen Zusammenhang mit dem realen AEP (Korrelationsfaktor 0.525).

```
In [121]: Vergleich_AEP = pd.DataFrame()
Vergleich_AEP["sim_output"] = Feldt["AEP [EUR/MWh]"]
Vergleich_AEP["historic"] = AEP1911_week["QUAL."]
#Vergleich_AEP["Diff_abs"] = Vergleich_AEP["historic"]-Vergleich_AEP["sim_output"]

#Vergleich_AEP["Diff_rel"] = Vergleich_AEP["Diff_abs"]/abs(Vergleich_AEP["historic"])*100
```

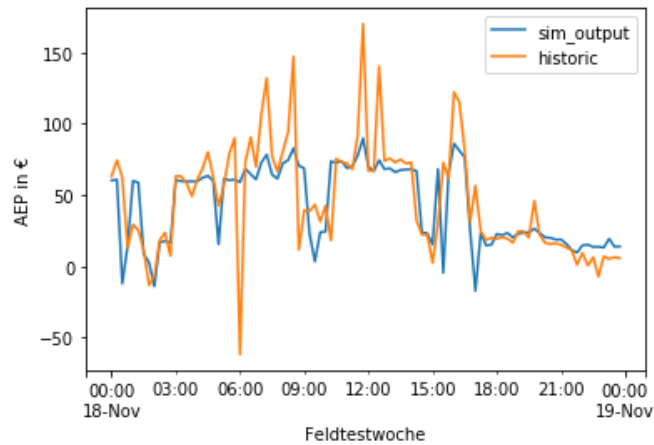
```
In [115]: Vergleich_AEP.corr()
```

```
Out[115]:
```

	sim_output	historic
sim_output	1.000000	0.524927
historic	0.524927	1.000000

```
In [120]: Vergleich_AEP["18.11.2019"].plot()  
plt.ylabel('AEP in €')  
plt.xlabel('Feldtestwoche')
```

```
Out[120]: Text(0.5, 0, 'Feldtestwoche')
```



## Conclusion (Simulation vs. Realität)

Der Einsatz und die Kosten von Regelenergie wird vom Modell sehr gut berechnet, was durch den sehr starken positiven linearen Zusammenhang zwischen den Vergleichswerten zeigt (Korrelationsfaktoren über 0.9).

Der berechnete AEP aus dem Modell hat einen mittelstarken positiven linearen Zusammenhang mit dem realen AEP (Korrelationsfaktor 0.525).

```
In [ ]:
```