

VPPLIB – EIN WERKZEUG FÜR DIE SIMULATION VIRTUELLER KRAFTWERKE AUF VERTEILNETZEBENE

Sascha BIRK^{*1,2}, Patrick LEHNEN², Lars PETERSEN², Gabriela WISE², Thorsten SCHNEIDERS^{1,2}

1 Virtuelles Institut Smart Energy (VISE) – Regionale Virtuelle Kraftwerke, www.smart-energy.nrw

2 Technische Hochschule Köln, Cologne Institute for Renewable Energy (CIRE), D-50679 Köln, +49 221 82752193, sascha.birk@th-koeln.de, www.th-koeln.de/anlagen-energie-und-maschinensysteme/cologne-institute-for-renewable-energy_13385.php

Kurzfassung: Für die künftigen Verteilnetze wird ein signifikanter Anstieg an dezentralen Energieanlagen wie Wärmepumpen, batterie-elektrischen Fahrzeugen und Photovoltaikanlagen erwartet. Diese bergen Herausforderungen für die Netze, da die installierte Leistung an den entsprechenden Anschlusspunkten stark ansteigt. Dieser Leistungsanstieg lässt sich durch elektrische und thermische Energiespeicher sowie geeignete Betriebsstrategien flexibilisieren. Diese Flexibilitäten wiederum lassen sich technisch, wie auch wirtschaftlich nutzen, um das lokale Energiesystem effizienter zu betreiben und die Energieanlagen in Netze und Märkte zu integrieren. Ein mögliches Geschäftsmodell stellen „Virtuelle Kraftwerke“ dar, die bereits seit gut zehn Jahren auf Übertragungsebene am Energiesystem partizipieren. Im Verteilnetz und insbesondere als Aggregator von Energieanlagen kleiner Leistung, haben sie sich bisher allerdings nicht etablieren können.

Für die Evaluierung der technischen und wirtschaftlichen Potentiale ist es nötig, das Verhalten der Energieanlagen auf Haushaltsebene sowie deren jeweiliger Einfluss auf die Verteilnetze zu analysieren. Neben den individuellen Einflüssen sind auch Implikationen von Anlagenkombinationen und aggregierten Anlagenparks zu untersuchen.

Um diese Untersuchungen durchführen zu können, wurde eine open source Modellbibliothek namens „VPPlib“ in der Programmiersprache Python entwickelt. Das vorliegende Paper beschreibt die Modellstruktur und Anwendungsbereiche der Bibliothek.

Keywords: Virtuelle Kraftwerke, Verteilnetze, Python, Open Source

1 Einleitung

Seit dem Beginn des Ausbaus der erneuerbaren Energien in Deutschland hat sich die Anzahl der Energieerzeugungseinheiten kontinuierlich erhöht. Das ursprünglich durch zentrale Kraftwerke mit Leistungen bis in den Gigawattbereich geprägte Energiesystem wird durchsetzt mit einer Vielzahl an dezentralen Erzeugungseinheiten im Kilowatt-Leistungsbereich. Diese Erzeugungseinheiten, bestehend aus Photovoltaikanlagen, Windenergieanlagen und Biomassekraftwerken sind zu einem großen Teil an die Mittel- und Niederspannungsnetze angeschlossen. In Nieder- und Mittelspannung waren 2014 zusammen ca. 22 GW Photovoltaikanlagen (PV), 14,5 GW Windenergieanlagen (WEA) und 25 GW Biomasseanlagen angeschlossen. Abhängig von dem zugrunde gelegten Ausbauszenario der

erneuerbaren Energien, werden sich diese Leistungen bis 2032 verdoppeln oder gar verdreifachen [1].

Neben der wachsenden Anzahl an elektrischen Erzeugungseinheiten ist auch ein Anstieg an elektrischen Verbrauchern höherer Leistung in den Verteilnetzen zu erwarten. Diese sind bedingt durch einen Wechsel von fossil befeuerten Wärmeerzeugern und Verbrennungsfahrzeugen die vornehmlich in privaten Haushalten zum Einsatz kommen, hin zu elektrisch betriebenen Alternativen wie Wärmepumpen und batterie-elektrischen Fahrzeugen. Diese finden sich ebenfalls im überwiegenden Maße in den Verteilnetzen wieder [1].

Im Gegensatz zu den Großkraftwerken sind die verteilten Erzeugungs- sowie die Verbrauchereinheiten nicht standardmäßig in eine, dem Energiesystem zuträgliche Betriebsweise eingebunden. Seit etwa zehn Jahren existieren zu diesem Zweck so genannte „Virtuelle Kraftwerke“ (VKW). Sie fungieren als Aggregator dezentraler Anlagen um diese einer Betriebsweise, ähnlich eines konventionellen Kraftwerksparks zu unterstellen. Bisherige VKW aggregieren überwiegend Erzeugungsanlagen ab einer Leistung von 100 kW und agieren auf Übertragungsnetzebene [2]. Dabei werden die Gegebenheiten in den Verteilnetzen der jeweiligen Energieanlagen nicht berücksichtigt.

Durch den Anstieg der dezentralen Energieanlagen in den Verteilnetzen werden vermehrte wetter- und nutzerabhängige Volatilitäten im elektrischen Energiebedarf erwartet. Dies geht einher mit einem steigenden Potential an Flexibilitäten, deren Nutzbarkeit für Systemdienstleistungen auf regionaler, sowie überregionaler Ebene zu untersuchen sind.

In dem vom europäischen Fonds für regionale Entwicklung (EFRE) und dem Land NRW geförderten Forschungsvorhaben „Virtuelles Institut Smart Energy – Regionale Virtuelle Kraftwerke (VISE-RVKW)“ wird diese Nutzbarmachung von Flexibilitäten auf Verteilnetzebene untersucht. Dabei werden sowohl energiemarktbedingte als auch technische Einflussfaktoren betrachtet.

Um die technischen Einflussfaktoren eines VKW auf Verteilnetzebene untersuchen zu können, wird ein geeignetes Werkzeug für die Untersuchungen benötigt. Zu den Eigenschaften dieses Werkzeuges zählen eine geeignete Abbildung der technischen Anlagen sowie des Netzes in dem die Anlagen lokalisiert sind.

Hinsichtlich der elektrischen Anlagen sollten die Erzeugungs- und Lastgänge möglichst realitätsgetreu nachgebildet werden. Eingriffe in den üblichen Betrieb und die damit einhergehenden Implikationen auf die Anlagen sollten ebenfalls berücksichtigt werden. Die aus den Anlagen resultierenden Lastflüsse müssen auf die Netze gespiegelt werden, um die dortigen Auswirkungen untersuchen zu können.

Die im Rahmen des VISE-RVKW Projektes entwickelte „VPPIib“ ist für diesen Anwendungsfall konzipiert und wird im Folgenden vorgestellt.

2 Methodik

Für die Erstellung der technischen Modelle wird die Programmiersprache „Python“ verwendet [3]. Diese zeichnet sich durch eine gute Handhabbarkeit und eine Vielzahl an offen zugänglichen und nutzbaren Bibliotheken aus (Open-Source-Software). Diese erleichtern unter anderem die Verarbeitung von anfallenden Daten und die Visualisierung der Ergebnisse.

Da sich Python im Bereich der Energiesystemmodellierung großer Beliebtheit erfreut, existieren bereits einige Open Source Bibliotheken von technischen Modellen und zur Netzanalyse. Die für diese Arbeit in Frage kommenden Bibliotheken werden zunächst vorgestellt und hinsichtlich ihrer Eignung für das Vorhaben untersucht.

Im Anschluss werden die neu entwickelten Modelle vorgestellt, die bisher noch nicht oder nicht im ausreichenden Maße in anderen Bibliotheken realisiert wurden. Zudem wird der Aufbau der VPPLib und der generelle Anwendungsbereich vorgestellt und diskutiert.

Um eine gute Reproduzierbarkeit der Ergebnisse zu gewährleisten wird die VPPLib ebenfalls als Open-Source-Software entwickelt und über die Plattform „GitHub“ zur Verfügung gestellt [4].

3 Identifizierung und Implementierung benötigter Modelle

In Folge eines Technologiescreenings wurden eine Vielzahl von Energieanlagen identifiziert [5]. Aus der erweiterten Auswahl an Technologien wurden, im Rahmen des VISE-RVKW Projektes, die Wärmepumpe, das batterieelektrische Fahrzeug, die Photovoltaik, BHKWs, sowie elektrische und thermische Energiespeicher als zweckmäßige Komponenten eines RVKW definiert. Um für Szenarien mit geringeren Anlagenzahlen die für den Einsatz am Markt benötigten Leistungskapazitäten zu erhalten, werden zudem Windenergieanlagen mit in die Betrachtung aufgenommen. Diese werden in der späteren Systembetrachtung im Mittelspannungsnetz verortet.

3.1 Vorstellung bestehender Technologiemodelle

Wie bereits in Kapitel 2 erläutert erfreut sich die Programmiersprache Python einer wachsenden Beliebtheit für die Modellierung von Energiesystemen. Damit einhergehend ist eine wachsende Anzahl an frei zugänglichen Bibliotheken aufzufinden. Im Folgenden werden drei Bibliotheken für die Analyse von Photovoltaik-Systemen, Windenergieanlagen und elektrischen Netzen vorgestellt und hinsichtlich ihrer Eignung für den Einsatz in der VPPLib diskutiert.

3.1.1 Vorstellung der PVlib

Die „PVlib“ wurde an den Sandia National Laboratories, ursprünglich in der Programmiersprache „MATLAB“ entwickelt. Die Bibliothek wurde anschließend ebenfalls in Python umgesetzt und wird stetig weiterentwickelt [6]. Die Unterschiede zwischen der MATLAB und der Python Version werden in [7] diskutiert.

Die PVlib verfügt über eine Datenbank an PV-Modulen und PV-Wechselrichtern sowie deren technischen Parametern, die für die Simulationen der Systeme verwendet werden können. Sie erlaubt zudem die Einbindung von Einstrahlungsdaten im csv-Dateiformat und umfasst ortsabhängige Berechnungen von Einflüssen des Sonnenstands auf die Erträge. Für die Berechnung der U-I-Kennlinie wird das Ein-Dioden-Modell verwendet [7].

Die PVlib wurde in [8] mit „Photovoltaik systems (PVSyst)“ und „System Advisor Model (SAM)“ verglichen. Von den drei betrachteten Software-Lösungen eignet sich allerdings nur die PVlib für eine weitere Verwendung innerhalb der VPPLib, da diese in Python geschrieben und dementsprechend gut integrierbar ist. In [8] wurden die berechneten Erträge aus den drei

Programmen ebenfalls mit den Erträgen einer realen Anlage verglichen. Dabei wurde von allen drei Programmen ein geringerer Ertrag errechnet. Bei der PVlib liegt die Abweichung bei 5,07%, für SAM bei 3,86% und für PVSyst bei 3,37%.

3.1.2 Vorstellung der windpowerlib

Die windpowerlib wurde am Reiner Lemoine Institut entwickelt und dient zur Erzeugung von Einspeisezeitreihen aus Windenergieanlagen [9]. Sie greift zurück auf eine Datenbank der „Open Energy Platform“, die Daten zu Windenergieanlagen bereitstellt [10]. Als Datengrundlage werden des Weiteren Zeitreihen von Temperatur, Luftdruck und Windgeschwindigkeiten unter Angabe der Höhe des Messpunktes über dem Boden, sowie der Bodenbeschaffenheit über die Angabe der Rauheit benötigt. Die Wetterdaten werden innerhalb der windpowerlib auf die Höhe der Windturbine umgerechnet.

Die Bibliothek wurde in [11] validiert. Zu diesem Zweck wurden gemessene Einspeisezeitreihen mit den Ergebnissen der Simulation verglichen. Für die Simulation wurden unterschiedliche Quellen an Wetterdaten herangezogen. Im Ergebnis ist, abhängig von der Datenquelle und der Höhe des Messpunktes über dem Boden, mit einer überschätzten Erzeugung an Energie in den Modellen zu rechnen.

3.1.3 Vorstellung von pandapower

Die Bibliothek „pandapower“ wurde im Fachgebiet Energiemanagement und Betrieb elektrischer Netze an der Universität Kassel, in Kooperation mit dem Fraunhofer-Institut für Energiewirtschaft und Energiesystemtechnik entwickelt. Pandapower ist für die statische Analyse von symmetrischen Netzen konzipiert und basiert auf „PYPOWER“, einer Bibliothek für Energiesystemanalysen [12]. Die Lösung von Lastflussberechnungen in pandapower wurde gegenüber PYPOWER durch just-in-time Kompilierung nochmals beschleunigt. Die Nutzung von „pandas“ als tabellenbasierte Datenstruktur sorgt für eine einfache Ein- und Ausgabe, sowie Weiterverarbeitung der Daten [13]. Pandapower wurde gegenüber kommerziell verfügbarer Software validiert und bereits in mehreren Netzstudien eingesetzt [14].

3.2 Aufstellen von noch nicht frei verfügbaren Modellen

Um die in Kapitel 3 gelisteten Technologien vollständig abzubilden, werden für jene Technologien, für die in Kapitel 3.1 keine existierenden Modelle identifiziert werden konnten eigene Modelle entwickelt. Diese basieren auf einem Lastprofilgenerator, der für die Arbeiten in [15] entwickelt wurde. Die dort verwendete Logik wird in eine objektorientierte Struktur übertragen und so in die VPPlib integriert. Die einzelnen Modelle werden an dieser Stelle daher nur kurz beschrieben.

3.2.1 Batterieelektrisches Fahrzeug

Für das batterieelektrische Fahrzeug werden die Ankunftszeiten der Fahrzeuge auf Grundlage von real gemessenem Nutzerverhalten aus [16] verwendet. Anhand des Wochentages wird aus einer Liste an Abfahrts- und Ankunftszeiten die Abwesenheit des Fahrzeugs, auf Basis von Beruflichen- und Freizeittätigkeiten oder Wochenendausflügen zufällig ermittelt. Für jede Stunde der Abwesenheit wird der Akku des Fahrzeugs um einen definierten Wert in kWh bis zu einem wahlweise festzulegenden Minimum entladen. Durch dieses Verfahren kann ein

Verbrauch pro Tag nachgestellt werden. Zudem lässt sich die Ladeleistung ab einem festzulegenden Speicherstand sukzessive reduzieren, um eine möglichst realistische Ladekurve zu erhalten.

3.2.2 Elektrischer Energiespeicher

Für den elektrischen Energiespeicher wird ein Modell mit den Parametern Speicherkapazität, Wechselrichterleistung und Wirkungsgrad verwendet. Das Laden bzw. Entladen des Speichers ist im jeweiligen Zeitschritt durch die Leistung des Wechselrichters beschränkt.

3.2.3 Thermischer Energiespeicher

Der thermische Energiespeicher wird über die Masse des Speichers, die spezifische Wärmekapazität des Speichermediums (c_p , bei Wasser 4,2 kJ/(kg*K)) und die Temperaturspreizung zwischen maximalem und minimalem Speicherstand definiert. Über die Formel

$$E = m \cdot c_p \cdot \Delta T \quad (3.1)$$

sind Energie und Temperatur des Speichers miteinander verknüpft. Umgerechnet in eine Temperatur des gesamten Speichers, lässt sich eine Hysterese bestimmen, die den Speicherbereich und somit den systembedingten Ein- und Ausschaltzeitpunkt des Wärmeerzeugers definiert.

3.2.4 Wärmepumpe

Für die Abbildung der Wärmepumpe erfolgt eine Berechnung des Coefficient of performance (COP) für wahlweise Luft- oder Erdwärmepumpen. Die Berechnungen basieren auf einer Analyse kommerziell erhältlicher Wärmepumpen und dem Vergleich mit Feldtests aus Deutschland und Großbritannien [17]. Aus dem COP und der elektrischen Leistung der Wärmepumpe wird in den Zeiten, in denen die Wärmepumpe betrieben wird, die thermische Erzeugung berechnet. Die Ein- und Ausschaltsignale werden in diesem Fall durch den thermischen Speicher bereitgestellt. Alternativ lässt sich die Zeitreihe der thermischen Leistung aus den UserProfiles durch die COP Zeitreihe teilen um das Betriebsverhalten bei kontinuierlichem Betrieb einer teillastfähigen Wärmepumpe darzustellen. Hierbei ist zu beachten, dass teillastbedingte Einflüsse auf den Wirkungsgrad der Wärmepumpe nicht berücksichtigt werden.

3.2.5 Blockheizkraftwerke

Für die Abbildung des Blockheizkraftwerkes (BHKW) werden elektrische und thermische Erzeugungsleistung definiert. Für den Betrieb werden Anfahr- und Abschaltzeiten sowie minimale Lauf- und Ruhezeiten der Anlagen festgelegt. In Verbindung mit einem thermischen Energiespeicher kann das BHKW wärmegeführt betrieben werden, in dem es auf die Hysterese bedingten Ein- und Ausschaltsignale des Speichers reagiert.

4 Aufbau der VPPLib

4.1 Grundstruktur der Bibliothek

Die Grundstruktur der VPPLib ist in Abbildung 1 dargestellt. Die Klasse „Environment“ beinhaltet, für das jeweilig zu untersuchende Szenario, alle Informationen hinsichtlich der lokalen Wetterdaten und des Zeitbereichs bzw. der Zeitschritte in denen die Untersuchungen durchgeführt werden. Die Klasse „UserProfile“ umfasst alle Informationen, die spezifisch für einzelne Akteure innerhalb des Netzes sind. Dies umfasst unter anderem deren Wärmebedarf, das Nutzungsverhalten von Elektrofahrzeugen sowie deren Position im Netzgebiet. Ein UserProfile kann dabei ein Gebäude oder eine Wohneinheit abbilden. Instanzen von Environment und UserProfile werden an eine oder mehrere der „Component“ Klassen übergeben. Diese bilden beispielsweise Photovoltaikanlagen, Wärmepumpen oder Elektrofahrzeuge ab, die jeweils als eine abgeleitete Klasse von Component implementiert sind. Instanzen dieser Technologieklassen werden an die Klasse „VirtualPowerPlant“ übergeben. Diese agiert als Aggregator der Technologien, die im jeweiligen Einsatzbereich vorkommen. Zudem werden einem pandapower network Objekt die jeweiligen Namen der Komponenten übergeben, um eine Verknüpfung zu Erzeugern, Speichern und Lasten in pandapower herzustellen. Über diese Referenz können im weiteren Verlauf Betriebsanweisungen und Statusinformationen zwischen dem Virtuellen Kraftwerk und dem Netzmodell ausgetauscht werden. Sowohl die Instanz der VirtualPowerPlant als auch die Instanz des pandapower network werden an die Klasse „Operator“ übergeben. In dieser werden Funktionen und Betriebsstrategien für die Steuerung des Virtuellen Kraftwerks implementiert und Rückmeldungen aus Verteilnetz und Kraftwerk verarbeitet.

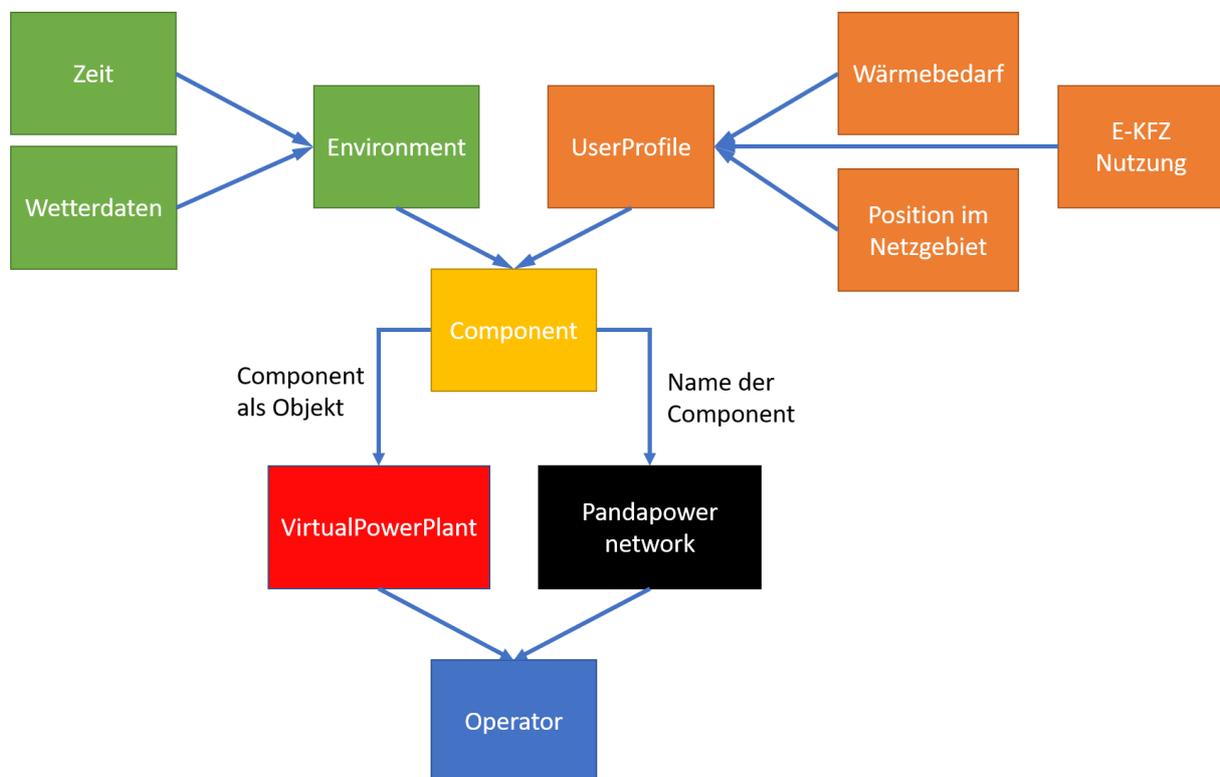


Abbildung 1: Grundstruktur der VPPLib

Diese Struktur erlaubt über die UserProfiles eine Darstellung der Nutzerpräferenzen in den Haushalten und die dadurch entstehenden Einflüsse auf den Betrieb. Auf der anderen Seite können im Operator wahlweise Betriebsstrategien umgesetzt werden, die das Netz berücksichtigen oder alleine das Virtuelle Kraftwerk betreffen. Hierdurch lassen sich entsprechend unterschiedlich granulare Szenarien und Geschäftsmodelle abbilden. Im Folgenden werden die Klassen der VPPlib eingehender betrachtet.

4.2 Das Environment

Für die Anwendung der Modelle auf spezielle Szenarien werden oftmals Daten verwendet, die global, für alle in dem Szenario verwendeten Modelle gleich sind. Hierzu zählen der Zeitbereich, in dem die Simulation oder Optimierung stattfindet, die Zeitschritte, in denen das Szenario untersucht wird sowie die Wetterdaten. Mit dem Environment wird eine zentrale Klasse geschaffen, die all diese Daten beinhaltet und über die jede Component auf sie zugreifen kann. Die jeweilige Instanz des Environments wird für das entsprechende Szenario jeder Component übergeben.

Die im Environment verwendeten Wetterdaten stammen aus dem „Climate Data Center (CDC)“ des Deutschen Wetterdienstes und sind über dessen ftp Server frei zugänglich [18]. Ein entsprechendes Skript zur Umwandlung der Rohdaten für die Verwendung mit der VPPlib ist ebenfalls in der VPPlib implementiert.

4.3 Das UserProfile

Auf regional aufgelöster Verteilnetzebene wird davon ausgegangen, dass das Verhalten einzelner Nutzer einen höheren Einfluss auf die zu erwartenden Lastflüsse im Netz hat. Bei Betrachtungen der Übertragungsnetze überlagern sich die Lasten mehrerer Nutzer an einem Netzknoten. Aus diesem Grund erlaubt die Klasse UserProfile eine genauere Abbildung dieser Nutzerpräferenzen. Hier können zuvor definierte Nutzergruppen abgebildet und eingebunden werden. In diesen Nutzergruppen finden sich Stromlastprofile basierend auf [20], sowie korrespondierende Wärmelasten. Die Wärmelasten werden aus dem Verhältnis der Jahreslast von Strom zu Wärme nach [20] errechnet. Das zeitlich aufgelöste Wärmelastprofil berechnet sich aus der jährlichen Wärmelast sowie dem Berechnungsverfahren „Standardlastprofil Gas“ nach [21].

Für die Abbildung von Unterschieden im Bereich der Elektromobilität, können die Daten unterschiedlicher, auf dem Markt erhältlicher batterie-elektrischer Fahrzeuge hinterlegt werden. Zusammen mit variablen, nutzerabhängigen An- und Abwesenheitszeiten ergeben sich individuelle Nachfragen für das Laden der Fahrzeuge. Dabei ist zu beachten, dass derzeit nur das Laden am eigenen Haus und nicht etwa an der Arbeitsstelle berücksichtigt wird.

4.4 Die Components

Die Grundversion der Component Klasse beinhaltet Vorgaben hinsichtlich Parametern und Funktionen, die jedes technische Modell der VPPlib umfassen sollte um eine bestmögliche Handhabbarkeit im Programmverlauf zu gewährleisten. Für die Erstellung von Zeitreihen sowie das Zurücksetzen und Auslesen dieser werden einheitliche Funktionsnamen definiert. Innerhalb der Modelle werden diese mit der entsprechenden Logik bestückt. Hinsichtlich der

Parameter wird definiert, dass jeweils eine Instanz des Environment und des UserProfile übergeben werden. Außerdem die Leistungs- oder Energieeinheit in der gerechnet wird sowie die Kosten, die für diese Einheit jeweils anfallen.

4.5 Die VirtualPowerPlant

Zur Aggregation der erzeugten Components wird die Klasse VirtualPowerPlant verwendet. Diese verfügt über Funktionen, um die Components in das Virtuelle Kraftwerk mit aufzunehmen und auch wieder zu entfernen. Zudem können die Zeitreihen der Components exportiert werden, sowohl in aggregierter Form als auch als separate Datenreihen.

4.6 Das pandapower network Objekt

Dieses Objekt entsteht aus der Erstellung eines elektrischen Netzmodells mithilfe der Bibliothek pandapower, welche in Kapitel 3.1.3 erläutert wird. Innerhalb dieses Netzelementes werden die Elemente, die zu den Components aus der VPPLib korrespondieren, dargestellt und mit den entsprechenden Namen und der Typ-Zuordnungen versehen. Somit ist ein Spiegelbild des VKW im Netzmodell vorhanden, um die Auswirkungen der Fahrweise des VKW auf das Netz zu untersuchen und in die Geschäftsmodelle und Betriebsstrategien mit einzubeziehen.

4.7 Der Operator

Der Operator Klasse werden sowohl die Instanz der VirtualPowerPlant inklusive der Components, als auch das pandapower network Objekt übergeben. Damit überblickt der Operator sowohl die Vorgänge im betrachteten VKW als auch im zugehörigen Netzgebiet. In dieser Klasse werden entsprechende Betriebsstrategien implementiert und die Rückkopplungen aus VKW und elektrischem Netz verarbeitet. Des Weiteren verfügt der Operator über Funktionen um die erzeugten Resultate weiter zu verarbeiten und zu visualisieren.

5 Fazit

Mithilfe der VPPLib ist es möglich, Geschäftsmodelle und Betriebsstrategien eines VKW auf regionaler Ebene und im Verteilnetz abzubilden. Die aufgrund von einzelnen Anlagen entstehenden Varianzen in den Energieflüssen der Verteilnetze, werden durch die Abbildung der technischen Modelle in VKW- und Netzmodell dargestellt. Der Eingriff des Operators in den Betrieb von Einzelanlagen, aufgrund von vorgegebenen Betriebsstrategien, lässt sich auf Seiten des technischen Modells, als auch auf Seiten des elektrischen Netzes nachvollziehen. Durch die Abbildung von Nutzerpräferenzen im UserProfile wird die Bibliothek um eine, über die rein technische Abbildung der Szenarien hinausgehende Fähigkeit erweitert.

Hinsichtlich der Erzeugungszeitreihen sind die Fehlerwahrscheinlichkeiten der technischen Modelle, beschrieben in Kapitel 3, bei der Evaluierung der Ergebnisse zu berücksichtigen. Da diese Fehler zwischen unterschiedlichen zu betrachtenden Szenarien, aufgrund der gleichbleibenden Datengrundlage nicht variieren, ist die Vergleichbarkeit der Szenarien trotzdem gegeben. Es ist allerdings zu überprüfen, ob Änderungen innerhalb des

Fehlerbereichs Auswirkungen auf die untersuchten Szenarien und damit auf die untersuchten Geschäftsmodelle haben.

6 Ausblick

Im weiteren Verlauf des „Virtuellen Institut Smart Energy – Regionale Virtuelle Kraftwerke“ werden Geschäftsmodelle und die damit verbundenen Aggregationskonzepte und Betriebsstrategien mit Hilfe der VPPLib untersucht. In diesem Verlauf wird die Operator Klasse um die entsprechenden Betriebsstrategien erweitert. Zudem werden unterschiedliche Flexibilitätsoptionen auf Haushaltsebene einander gegenübergestellt und hinsichtlich ihrer Wechselwirkungen untersucht. Des Weiteren wird die Bibliothek für den Einsatz in Optimierungsszenarien weiterentwickelt. In diesem Zusammenhang werden auch weitere Open Source Energiesystemmodelle analysiert um Synergien zu nutzen.

7 Danksagung

Wir bedanken uns beim Ministerium für Wirtschaft, Innovation, Digitalisierung und Energie (MWIDE) des Landes Nordrhein-Westfalen sowie beim Europäischen Fonds für Regionale Entwicklung (EFRE) für die Förderung dieses Projektes.

8 Literatur

- [1] Büchner, J., Flörcken, O., Dierkes, S., Verheggen, L., & Uslar, M: „Moderne Verteilernetze für Deutschland“ (Verteilernetzstudie), 12.09.2014
- [2] Next Kraftwerke, „Was ist ein Virtuelles Kraftwerk?“, aufgerufen am 29.01.2020, <https://www.next-kraftwerke.de/wissen/virtuelles-kraftwerk>
- [3] G. van Rossum, Python tutorial, Technical Report CS-R9526, Centrum voor Wiskunde en Informatica (CWI), Amsterdam, May 1995.
- [4] „VPPLib“, aufgerufen am 28.01.2020, <https://github.com/Pyosch/vpplib>
- [5] Samir Jeddi, Sascha Birk, Johannes Fleer, Max Schönfisch, Georg Holtz, Christian Tode, Dietmar Lindenberger, Thorsten Schneiders: "Virtuelles Institut Smart Energy - (Regionale) Virtuelle Kraftwerke: Definitorische Grundlagen und erste Erkenntnisse" (Langfassung), Köln, 10/2018
- [6] William F. Holmgren, Clifford W. Hansen, and Mark A. Mikofski: "pvlib python: a python package for modeling solar energy systems." Journal of Open Source Software, 3(29), 884, (2018). <https://doi.org/10.21105/joss.00884>
- [7] „pvlib-python“, <https://pvlib-python.readthedocs.io/en/stable/>, aufgerufen am 28.01.2020
- [8] Gurupira, T., Rix, A. J., Gurupira, T., & Rix, A. J.: Pv Simulation Software Comparisons: Pvsyst, Nrel Sam and Pvlib. 01/2017
- [9] Sabine Haas, Birgit Schachler, Uwe Krien, & Stephen Bosch. (2019, September 9). wind-python/windpowerlib: Revision release (Version v0.2.0). Zenodo. <http://doi.org/10.5281/zenodo.3403360>
- [10] OpenEnergy Plattform, aufgerufen am 28.01.2020 https://openenergy-platform.org/dataedit/view/supply/wind_turbine_library

- [11] Sabine Haas: "Implementation and validation of an open source model for generating wind feed-in time series", 23.01.2019, https://reiner-lemoine-institut.de/wp-content/uploads/2019/01/MasterThesis_SabineHaas.pdf
- [12] R. D. Zimmerman, C. E. Murillo-Sanchez, and R. J. Thomas, "MATPOWER: Steady-State Operations, Planning and Analysis Tools for Power Systems Research and Education," Power Systems, IEEE Transactions on, vol. 26, no. 1, pp. 12–19, Feb. 2011.
- [13] McKinney, W. (n.d.). pandas: A Foundational Python Library for Data Analysis and Statistics. <https://de.scribd.com/document/71048089/pandas-a-Foundational-Python-Library-for-Data-Analysis-and-Statistics>
- [14] L. Thurner, A. Scheidler, F. Schäfer et al, pandapower - an Open Source Python Tool for Convenient Modeling, Analysis and Optimization of Electric Power Systems, in IEEE Transactions on Power Systems, vol. 33, no. 6, pp. 6510-6521, Nov. 2018.
- [15] Sascha Birk, Christian Brosig, Eberhard Waffenschmidt, "Auswirkungen von Sektorkopplungsmaßnahmen im zukünftigen innerstädtischen Niederspannungsnetz", 15. Symposium Energieinnovation, Graz, 02/2018, DOI 10.3217/978-3-85125-586
- [16] Thorben Doum, „Notwendigkeit und Rahmenbedingungen eines Lastmanagements für Elektromobilität in Niederspannungsnetzen“, Köln, 12.08.2015
- [17] Staffel I., Brett D., Brandon N., Hawkes A., "A review of domestic heat pumps", Energy & Environmental Science, 2012
- [18] Deutscher Wetterdienst, "CDC-Portal", aufgerufen am 28.01.2020, ftp://opendata.dwd.de/climate_environment/CDC/
- [19] Tjarko Tjaden, Joseph Bergner, Johannes Weniger, Volker Quaschnig, "Repräsentative elektrische Lastprofile für Wohngebäude in Deutschland auf 1-sekundiger Datenbasis", Berlin, 05/2017
- [20] Umweltbundesamt, „Energieverbrauch privater Haushalte“, aufgerufen am 28.01.2020, <https://www.umweltbundesamt.de/daten/private-haushalte-konsum/energieverbrauch-privater-haushalte>
- [21] BDEW, VKU, GEODE, „Abwicklung von Standardlastprofilen Gas“, Berlin, 30.06.2015